

UNIVERSIDAD DE EL SALVADOR
FACULTAD DE INGENIERIA Y ARQUITECTURA
ESCUELA DE INGENIERÍA DE SISTEMAS INFORMÁTICOS



Sistema Informático para planeación y gestión de los procesos de servicio de mantenimiento del Área de Mantenimiento General, Nivel Regional y Local del Ministerio de Salud. (SIM)

PRESENTADO POR:

EMERSON JOAQUÍN MINERO SÁNCHEZ

CALEB RUBÉN RODRÍGUEZ OROZCO

CARLOS AARÓN ROMERO DELGADO

ALBER JOSÉ ROMERO SALGADO

PARA OPTAR AL TÍTULO DE:

INGENIERO DE SISTEMAS INFORMÁTICOS

CIUDAD UNIVERSITARIA, MARZO DE 2014

UNIVERSIDAD DE EL SALVADOR

RECTOR :

ING. MARIO ROBERTO NIETO LOVO

SECRETARIA GENERAL :

DRA. ANA LETICIA ZAVALETA DE AMAYA

FACULTAD DE INGENIERIA Y ARQUITECTURA

DECANO :

ING. FRANCISCO ANTONIO ALARCÓN SANDOVAL

SECRETARIO :

ING. JULIO ALBERTO PORTILLO

ESCUELA DE INGENIERIA DE SISTEMAS INFORMÁTICOS

DIRECTOR :

ING. JOSÉ MARÍA SÁNCHEZ CORNEJO

UNIVERSIDAD DE EL SALVADOR
FACULTAD DE INGENIERIA Y ARQUITECTURA
ESCUELA DE INGENIERIA DE SISTEMAS INFORMÁTICOS

Trabajo de Graduación previo a la opción al Grado de:

INGENIERO DE SISTEMAS INFORMÁTICOS

Título :

Sistema Informático para planeación y gestión de los procesos de servicio de mantenimiento del Área de Mantenimiento General, Nivel Regional y Local del Ministerio de Salud. (SIM)

Presentado por :

EMERSON JOAQUÍN MINERO SÁNCHEZ

CALEB RUBÉN RODRÍGUEZ OROZCO

CARLOS AARÓN ROMERO DELGADO

ALBER JOSÉ ROMERO SALGADO

Trabajo de Graduación Aprobado por:

Docente Director :

ELMER ARTURO CARBALLO RUIZ

San Salvador, marzo de 2014

Trabajo de Graduación Aprobado por:

Docente Director :

ELMER ARTURO CARBALLO RUIZ

Dedicatoria

A mi madre, la mujer que siempre ha estado para mí incondicionalmente.

Agradecimientos

Ha llegado el momento culmine de una etapa, la cual marca el inicio de otras y antes de extenderme con las personas que han marcado huella en este período, doy gracias a Dios por permitirme estar con salud y haber terminado con éxito nuestro Trabajo de Graduación al lado de mi equipo de trabajo, en cada una de nuestras defensas depositamos la confianza en él y hasta este momento la mantenemos.

Él ha sido la fuerza motora que me ha impulsado en esos momentos donde nadie más pudo hacerlo, haber puesto la esperanza, confianza y convicción en él, es lo que ha dado como resultado el hecho de estar escribiendo en estos momentos estos párrafos.

Probablemente pareciera que esto es el resultado de todo un año duro de trabajo, pero en realidad esto inició desde el primer día de clases en la Universidad de El Salvador, espero poder dar un espacio aunque sea pequeño a todas esas personas que formaron parte de mi vida desde ese entonces.

Agradezco a mi familia por ser mi motivación para salir adelante, por creer en mí y darme esas palabras de aliento cuando las cosas no iban bien, gracias a mi madre preciosa, ella es una persona única e invaluable, ella en todo este tiempo se negó a sí misma cualquier necesidad que tuviera para poderme dar lo que yo necesitaba en la Universidad, siempre recordaré eso y más allá de una ayuda económica, su forma de vivir y la actitud que tiene ante la vida es algo que me llena interiormente y siempre me hace seguir luchando, madre bella gracias por todo, es por ella por quien yo soy una persona que intenta hacer las cosas de la mejor manera.

Gracias a mi hermana Karen, ella siempre ha mostrado respeto y admiración por mis proyectos, gracias por creer en mí y por su apoyo incondicional, a mi padre y hermanos. No puedo dejar de mencionar a mis tíos y primos, los que están lejos o cerca, siempre me ayudaron de una u otra manera, gracias por mostrarme el valor que tiene el mantenerse unidos. Quiero mencionar de forma especial a mi tío Rigoberto, Mario y Teresa, para poder culminar esta etapa ellos me ofrecieron su ayuda incondicional y sin importar como lo haga eso no se los puedo pagar porque no tiene precio.

Agradezco de forma especial a la familia Iraheta quienes son mis amigos sinceros, cada vez que llegue a su casa a prestar su computadora, no tengo ningún recuerdo que me la hubieran negado, siempre dejaban de usarla para cedérmela, lo tengo presente y se los agradezco enormemente, gracias por su amistad y darme ese apoyo.

En todo este trayecto tuve ayuda de tantas personas y entre ellas quiero mencionar a tres que pertenecen a la Unidad de Estudios Socioeconómicos: Licda. Karen Campos, Licda. Yesenia de Guzmán e Ing. Ubaldo Cruz, el tiempo que trabaje ahí fue de mucho aprendizaje, ellos saben de la forma que me apoyaron y eso permitió que pudiera seguir estudiando de mi cuarto año hasta la fecha actual, gracias por su amistad tan valiosa.

Este proyecto no se pudo haber realizado sin la dirección y coordinación de una gran persona y profesional a la vez, nuestro asesor de tesis Ing. Elmer Arturo Carballo, mis respetos hacia él, es una persona con un cúmulo de conocimientos obtenidos gracias a su experiencia, y lo demostró en cada asesoría que nos dio, gracias por todos esas directrices que nos dio desde un inicio, por motivarnos

hacer las cosas de la mejor manera, por formar ese lazo de amistad con nosotros, por haber aceptado asesorar a nuestro equipo.

Otra persona importante que marco huella en mi carrera fue mi amigo y maestro Ing. Rubén Ascencio, el me dio las bases para que yo pudiera seguir avanzando, lo admiro mucho por su pedagogía y los principios que tiene como persona, gracias por sus conocimientos y sincera amistad.

Agradezco al Área de Mantenimiento General y la Dirección de Tecnologías de Información y Comunicaciones del Ministerio de Salud junto a todo el personal de ambas instituciones que nos brindaron su ayuda para poder obtener el título de Ingeniero de Sistemas Informáticos, ellos siempre estuvieron abiertos a las peticiones que como equipo les hicimos, deseo hacer mención especial al Ing. Noel Ayala, Lic. Jorge Alas, Inga. Sonia Viana e Inga. Karen Peñate.

Una familia que siempre estuvo pendiente de nuestro trabajo y a quienes estoy inmensamente agradecido es a la familia Romero Delgado, ellos estuvieron todo el año con nosotros, apoyándonos con el alojamiento, alimentación, nuestra salud, les tengo un especial aprecio y espero algún día poder retribuirles aunque sea un poco de lo mucho que ellos nos dieron, gracias por todo Sra. Raquel Delgado, usted se portó como una madre para nosotros, gracias a Nelson Delgado y Miriam Romero por hacernos sentir como de la familia durante el tiempo que estuvimos ahí, gracias a las hermanas Romero, su hospitalidad fue invaluable.

Llegando casi al final de este escrito, deseo agradecer de forma especial a mi amiga Tatiana Rivera, porque durante el tiempo de desarrollo del Trabajo de Graduación siempre estuvo con la disponibilidad de compartir ideas y hacer que pudiera tomar pausas, para poder seguir trabajando tomando aspectos que por mi cuenta hubieran sido difíciles de notar.

Finalmente esto fue un trabajo de equipo, no únicamente como compañeros sino como amigos, agradezco a Caleb Rodríguez, Aarón Romero y Alber Romero por haber superado este obstáculo juntos, por haber logrado esa cohesión y buena comunicación, lo cual fue clave para poder desarrollar lo mejor que se pudo, todo lo que el proyecto conllevaba. Gracias por haber sido tolerantes con mis defectos y errores, por ayudarme cuando no sabía cómo resolver un problema, por esa amistad que se logra con el tiempo, los admiro como persona, al igual que la capacidad que poseen, son grandes mis amigos y futuros ingenieros, sólo falta recibir ese ansiado título, les deseo lo mejor en su vida laboral. Con estima.

Emerson Joaquín Minero Sánchez.

Dedicatoria

Con todo mi cariño a mi padre y madre, quienes me ayudaron a hacer posible la culminación de este gran sueño, motivándome e impulsándome, brindándome siempre su apoyo incondicional, a ellos, mis más sinceros agradecimientos: José Osmín Rodríguez y Ana Isabel Orozco.

Agradecimientos

La vida es como un lienzo en blanco, en el que se va escribiendo o dibujando poco a poco, y en donde sólo los valientes cuentan grandes historias.

Agradezco a Dios primeramente por haberme permitido llegar a cumplir este sueño y poner a las personas adecuadas dentro del camino, que me permitieron llegar a la culminación de esta etapa de mi vida, a mis padres por siempre brindarme su apoyo en los momentos más difíciles, por sus sabios consejos, por el cuidado que ha tenido de mí a pesar de la distancia, por su comprensión, por su cariño, por su sacrificio y quienes desde temprana edad inculcaron valores en mi vida y me enseñaron el éxito solamente se logra a través del sacrificio y la perseverancia, a mis hermanos que siempre me apoyaron, a Nehemías Otoniel Ordoñez que siempre me apoyó incondicionalmente, a Carlos Antonio Ponce Rodríguez y Norma Lilian Reyes Rodríguez, quienes me extendieron su mano, me abrieron las puertas de su casa, me apoyaron de manera desinteresada y quienes estuvieron ahí en los momentos más difíciles, a mis compañeros de Tesis y en especial a Carlos Aarón Romero Delgado, a su tío Nelson Delgado, a su madre Raquel Delgado y a su familia por siempre recibirnos con los brazos abiertos, por su hospitalidad y cuidado, por abrirnos las puertas de su casa durante todo el trabajo de la tesis, haciéndolo de manera desinteresada y con toda su voluntad.

Las palabras no alcanzan para explicar la gratitud que tengo con estas personas quienes siempre estuvieron ahí, y de quienes he aprendido mucho a lo largo del tiempo que he pasado con ellos, con quienes he establecido un gran vínculo, una amistad sincera y quienes han dejado una marca dentro de mi vida, y a quienes siempre les estaré infinitamente agradecidos.

En lo académico y profesional al Ing. Elmer Arturo Carballo, por su excelente visión como profesional, como asesor, por su vasta experiencia y por habernos guiado de excelente manera en esta última etapa de nuestra vida académica, al Ministerio de Salud dentro de los que se encuentran el Ing. Carlos Martín, la Ing. Sonia Viana, la Ing. Karen Peñate, el Ing. Noel Ayala y PhD. Jorge Alas quienes siempre estuvieron prestos para recibirnos y atender todas las solicitudes que les hacíamos, por abrirnos las puertas del Ministerio y Ayudarnos en el trabajo realizado y quienes fueron parte importante para la culminación exitosa de este proyecto.

“Nadie está a salvo de las derrotas. Pero es mejor perder algunos combates en la lucha por nuestros sueños, que ser derrotado sin saber siquiera por qué se está luchando.”

Pablo Coelho

Caleb Rubén Rodríguez Orozco.

Dedicatoria

A Dios, quien me ha dado la vida y la salvación por medio de su Hijo Jesús. Él me ha dado la sabiduría, la inteligencia y ha provisto todo lo necesario para llegar hasta este punto de mi vida.

A mis padres, Raquel Delgado y Francisco Romero, quienes han forjado en mi disciplina, orden y buenos principios, y sobre todo me dieron amor e inculcaron la fe en Dios. Su esfuerzo y trabajo por sus hijos es invaluable y ahora se ve reflejado en mi persona.

Agradecimientos

A Dios por darme sabiduría, por guiarme a lo largo de la carrera y todos mis estudios. Él ha permitido que llegue a tan importante etapa de mi vida y carrera profesional. Gracias por la salud que he tenido, pues no hubo motivos para faltar a clase, para perder alguna evaluación o algún accidente que lamentar. Gracias por las bendiciones y por la provisión que nunca ha faltado. Porque muchas puertas se han abierto gracias a Él y a puesto a las personas indicadas en el momento adecuado. La gloria y honra son para Él.

A mis padres, por su apoyo incondicional, por su trabajo y esfuerzo por lograr que el futuro de sus hijos sea de bien. Por haber inculcado en mí el deseo de superación, buenos principios y valores cristianos. A mi madre, por darme su amor cada día, su comprensión, sus palabras de aliento y ánimo en los momentos de tensión y tristeza, por las oraciones que salieron de su corazón en todo momento, por sus buenos consejos, por su cuidado y cariño no solo en este proceso sino desde mi niñez. A mi padre, por haber exigido, porque eso hizo esforzarme más, y por forjar en mi la disciplina y el esfuerzo por alcanzar lo que deseamos. Los amo.

A mi tía Miriam Romero, por su apoyo incondicional en todo momento, porque su amor desde la distancia, por habernos tomado, a mis hermanas y a mí, como parte especial de su vida y una razón por la cual esforzarse. Por querer y esperar lo mejor de mí.

A mi abuelita María Cristina Romero, por darme su amor y cariño como a un hijo, por esas palabras de ánimo, y su ejemplo de esfuerzo y superación. Enseñando que trabajando horadamente es como se sale adelante y por sus buenos consejos.

A mis hermanas, Cristina Romero y Miriam Romero, por ser especiales en mi vida, porque me dan su cariño y amor, por sus atenciones y cuidados. Porque a veces parece que soy el menor y ellas me cuidan. Si he sido ejemplo para ustedes, recuerden que es gracias a Dios y seguir su camino. No deben ser como yo, deben ser mejores que yo.

A Tania Alvarenga, por darme su amor, por sus cuidados, su cariño y atención. Por esas palabras de ánimo en el momento que las necesite, por su apoyo y comprensión. Por el tiempo que hemos estado juntos, porque formas parte de mi vida. Porque a veces no estuve ahí, por mis estudios, pero aun así me das tu amor.

A la familia Alvarenga, de forma especial a Francisco Alvarenga, mi Pastor, por instruirme y enseñarme de la palabra de Dios y su esposa Juanita De Alvarenga, por exigirme responsabilidad y sus atenciones.

A mi tío Nelson delgado, por su apoyo y ayuda sin esperar nada a cambio, por estar pendiente de nuestras necesidades y su cariño. Gracias por prestar su hogar al equipo de trabajo a lo largo del proyecto, que cómicamente lo llamamos como Centro de Operaciones.

A nuestro asesor del trabajo de graduación, Ing. Elmer Carballo, por orientarnos con su gran experiencia profesional, por compartir su conocimiento para lograr un proyecto exitoso. Por el tiempo, la amistad, los buenos consejos y por habernos exigido más de lo que podíamos dar, que fue para nuestro bien y hoy vemos la recompensa de este gran trabajo.

A mis compañeros de tesis, Caleb, Emerson y Alber, que a la vez son mis amigos. Porque Dios los puso en mi camino y hemos formado un gran equipo. Por su esfuerzo, por su preocupación, por trabajar para que todos seamos hoy Ingenieros de Sistema Informáticos, no solo en el trabajo de gradación, sino cada vez que formamos grupos de trabajo durante la carrera, porque su esmero es invaluable. Por compartir conmigo sus conocimientos, su dedicación, por enseñarme que antes la adversidad siempre hay que dar lo mejor. Disfruten ahora este esfuerzo.

A todos mis compañeros e ingenieros de la universidad, que de un u otra forma han sido apoyo para culminar mi carrera, que hemos compartido esfuerzos, conocimiento y formado grandes lazos de amistad: Gustavo Romero, Douglas Tobar, Rodrigo Bazán, y muchos más a quienes tengo muy presente.

Gracias...

Carlos Aarón Romero Delgado.

Dedicatoria

A Dios y mi Salvador El Señor Jesucristo, "Porque de él, y por él, y para él, son todas las cosas. A él sea la gloria por los siglos. Amén." Romanos 11:36.

A mis padres y mis hermanos y hermanas.

Agradecimientos

Agradezco a Dios y a su Hijo el Señor Jesucristo mi Salvador (1 Pedro 2:24, Juan 3:16) a mi padre José Noé Romero y a mi madre Alba Flora Salgado de Romero por su apoyo incondicional en todo el transcurso de la carrera, agradezco a Hazel Brownlie por sus consejos y por estar en esos momentos más difíciles para brindarme su apoyo, agradezco a mis hermanos y hermanas por su apoyo.

Alber José Romero Salgado.



Índice

1	INTRODUCCIÓN.....	18
2	OBJETIVOS	20
2.1	Objetivo General.....	20
2.2	Objetivos Específicos	20
3	JUSTIFICACIÓN.....	21
4	IMPORTANCIA.....	22
5	ALCANCES.....	24
5.1	Resultados del Sistema Informático.	24
5.2	Características del Sistema Informático.	25
5.3	Productos Entregables	26
5.3.1	Institución.....	26
5.3.2	Universidad.	26
6	LIMITACIONES.....	26
7	CAPÍTULO I - GENERALIDADES	27
7.1	Acrónimos, Abreviaturas y Definiciones Generales.....	28
7.1.1	Acrónimos y Abreviaturas Generales	28
7.1.2	Definiciones Generales	28
7.2	Marco Teórico.	30
7.2.1	Establecimientos de Salud y Redes que conforman el MINSAL.....	30
7.2.1.1	Ministerio de Salud.....	30
7.2.1.2	Dependencias.	30
7.2.1.3	Región.....	30
7.2.1.4	Redes Integrales e Integradas de Servicio de Salud.....	30
7.2.2	Documentos Requeridos en los Servicios de Mantenimiento.	32
7.2.2.1	Solicitud de Servicio.....	32
7.2.2.2	Orden de trabajo.....	32
7.2.2.3	Orden de Requisición.....	32
7.2.2.4	Vale de Salida.	32
7.2.2.5	Inventario Técnico de Equipo.....	32
7.2.2.6	Programa Anual de Mantenimiento Preventivo.....	33
7.2.2.7	Rutinas de Mantenimiento Preventivo.....	33
7.2.2.8	Ficha de Vida.....	33



7.2.2.9	Informe de Actividades de Mantenimiento.....	33
7.3	Marco Legal.....	34
7.3.1	Manual de Organización y Funciones de la Gerencia General de Operaciones.....	34
7.3.2	Manual de Procedimientos de la Unidad de Conservación y Mantenimiento del Ministerio de Salud. 34	
7.3.3	Manual de Organización y Funciones de la Dirección de Tecnologías de Información y Comunicaciones del Ministerio de Salud.	34
7.3.4	Lineamientos Técnicos para la Administración de Activo Fijo del Ministerio de Salud.	35
7.3.5	Manual de Organización y Funciones de las RIISS.	35
7.3.6	Lineamientos Técnicos para el Mantenimiento, Administración y Desarrollo de Tecnologías de Información y Comunicaciones.	36
7.3.7	Otros Instrumentos Legales.....	36
7.4	Antecedentes.....	37
8	CAPÍTULO II - METODOLOGÍA.....	39
8.1	Metodología de Recolección de Datos.....	40
8.1.1	Entrevista.....	40
8.1.2	Encuesta.	41
8.2	Técnicas de Diagnóstico y Formulación del Problema.....	44
8.2.1	Diagrama Causa-Efecto.....	44
8.2.2	Análisis FODA.....	46
8.2.3	Enfoque de Sistemas.....	47
8.2.4	Diagrama de Caja Negra.....	48
8.3	Gestión de Proyectos.....	49
8.3.1	Acta de Constitución del Proyecto.....	53
8.3.2	Administración de la Comunicación del Proyecto.	53
8.3.2.1	Identificar a los interesados (stakeholders).....	53
8.3.2.2	Planificación de la Comunicación con los Interesados.....	55
8.3.3	Gestión de los Riesgos del Proyecto.....	58
8.3.3.1	Plan de Gestión de Riesgos.....	58
8.3.3.2	Identificación de los Riesgos.....	61
8.3.3.3	Planificación de Respuestas a los Riesgos.....	61
8.3.3.4	Monitoreo de Riesgos.....	62
8.4	Metodología de Desarrollo de Software.....	63
8.4.1	Ciclo de Vida de Desarrollo de Software Incremental.	64
8.5	Metodología de Análisis de Requerimientos.....	67
8.6	Técnicas de Análisis y Diseño.....	71



8.6.1	Enfoque Orientado a Objetos.....	71
8.6.2	UML 2.0.....	72
8.6.3	Modelado de Procesos de Negocio	78
8.7	Metodología de Diseño de Base de Datos	80
8.8	Técnicas de Programación.....	85
8.9	Metodología de Pruebas del Software.....	88
8.9.1	El Plan de Pruebas y el Ciclo de Vida de Desarrollo de Software.....	88
8.9.1.1	Etapas del Proceso de Pruebas.....	89
8.9.1.2	Modelo del Proceso de Pruebas	90
8.9.2	Diseño y Ejecución de Casos de Prueba.	91
8.9.2.1	Lineamientos de Diseño y Ejecución:	91
8.9.2.2	Software para Implementación y Ejecución de Casos de Prueba.	93
8.9.3	Sistema de Seguimiento de Errores (Bug Tracking System).	94
8.9.4	Técnicas de Pruebas de Software.....	95
8.10	Estándares de Documentación.....	98
8.11	Control de Versiones.....	99
8.11.1	Control de Versiones de Software.....	99
8.11.1.1	Metodología de Desarrollo con GIT.....	100
8.11.2	Control de Versiones de Documentación.....	104
8.12	Almacenamiento en la Nube	106
8.13	Descripción de Herramientas.	108
8.14	Cronograma de Actividades.....	110
8.15	Planeación de recursos y costo del proyecto.....	115
8.15.1	Presupuesto y costo del Proyecto.	115
9	CAPÍTULO III - ANÁLISIS.....	116
9.1	Análisis de la Situación Actual	117
9.1.1	Descripción de la Organización.....	117
9.1.1.1	Organización del Mantenimiento en el Nivel Superior	117
9.1.1.2	Organización del Mantenimiento en el Nivel Regional.....	120
9.1.1.3	Organización del Mantenimiento en el Nivel Local.....	121
9.1.2	Proceso Principal de los Servicios de Mantenimiento	122
9.1.3	Enfoque de Sistemas de la Situación Actual.....	125
9.2	Planteamiento del Problema.....	128
9.2.1	Diagnostico	128
9.2.1.1	Análisis Causa-Efecto.....	128



9.2.1.2	Diagrama Causa-Efecto	130
9.2.1.3	Análisis FODA	131
9.2.2	Definición del Problema.....	133
9.2.3	Formulación del Problema.....	134
9.3	Solución Propuesta	135
9.3.1	Propuesta del Proceso de Servicio de Mantenimiento para el SIM.	137
9.3.2	Enfoque de Sistemas de la Solución Propuesta	138
10	CAPÍTULO IV - ESPECIFICACIÓN DE REQUERIMIENTOS (SRS)	141
10.1	Introducción.....	142
10.1.1	Propósito	142
10.1.2	Definiciones, Acrónimos y Abreviaturas.	142
10.1.2.1	Definiciones de Elementos de Casos de Uso	142
10.2	Resumen del Documento.	143
10.3	Descripción General	144
10.3.1	Perspectiva del Producto.	144
10.3.1.1	Funciones del Producto.	146
10.3.2	Características de los Usuarios.	147
10.3.2.1	Clasificación de Usuarios.	147
10.3.2.2	Perfil de los Usuarios.	148
10.3.3	Restricciones.....	152
10.3.3.1	Políticas Reguladoras.....	152
10.3.3.2	Contenido Mínimo de Manuales Adjuntos al Sistema Informático.	153
10.3.3.3	Restricciones de Diseño.....	154
10.3.3.4	Consideraciones de Seguridad.....	157
10.3.3.5	Suposiciones y Dependencias.	157
10.4	Especificación de Requerimientos.....	160
10.4.1	Interfaces externas.....	160
10.4.1.1	Interfaces del Sistema.....	160
10.4.1.2	Interfaces de Usuario.	160
10.4.1.3	Interfaces de Software.....	161
10.4.1.4	Interfaces de Comunicación.	161
10.4.1.5	Requerimientos de Adaptación del Sitio.....	162
10.4.2	Requerimientos funcionales.	163
10.4.2.1	Requerimientos Funcionales Generales.....	163
10.4.2.2	Requerimientos Funcionales Específicos.	167



10.4.2.3	Diagramas de Secuencia del Sistema	171
10.4.3	Otros atributos y requerimientos.	174
10.4.3.1	Requerimientos de Hardware.	174
11	CAPÍTULO V - DISEÑO	176
11.1	Diseño de Estándares.	177
11.1.1	Estándar de Estructura de Página	177
11.1.2	Estándares de Presentación de Contenido.	179
11.1.3	Estándares de Salida.	182
11.1.4	Estándares de Documentación Interna.	187
11.1.4.1	Estándares de Documentación Interna del Código Fuente	187
11.1.4.2	Estándares de Documentación Interna de los Objetos de la Base de Datos.....	189
11.1.5	Estándares de Documentación Externa.....	189
11.1.6	Estándares de Base de Datos.	191
11.1.7	Estándares de Mensajes.	194
11.1.8	Estándares de Código.	197
11.1.9	Estándares de Estructura de Navegación.	201
11.1.10	Estándares de Estructura de Archivos.	202
11.2	Contexto del Sistema.....	205
11.2.1	Modelo de Contexto del Sistema.....	205
11.2.2	Modelo de Interacción del Sistema.....	206
11.2.3	Diagrama de Contexto Arquitectónico.....	208
11.3	Diseño Arquitectónico.	210
11.3.1	Diseño Arquitectónico de Alto Nivel	210
11.3.2	Modelo de Tres Capas.	212
11.3.3	Patrón MVC (Modelo-Vista-Controlador).....	213
11.3.4	Funcionamiento Interno del Framework Symfony.....	215
11.3.5	Diseño a Nivel de Componentes.....	218
11.3.5.1	Descripción del Diagrama de Componentes del Sistema.....	219
11.3.6	Diseño de la Arquitectura Física (Despliegue)	222
11.3.7	Diseño de la Base de Datos	228
11.3.7.1	Diagrama Entidad – Relación (DER).....	228
11.3.7.2	Modelo Lógico.	229
11.3.7.3	Modelo Físico.	230
11.3.8	Conclusión Sobre Diseño Arquitectónico	231
11.4	Diseño de Seguridad.....	233



11.4.1	Registro de Usuarios y Activación mediante Correo Electrónico.....	233
11.4.2	Privilegios Mediante Grupos y Roles.	235
11.4.3	Generación de Contraseñas Seguras.	235
11.4.4	Cifrado de Contraseñas en la Base de Datos.....	237
11.4.5	Inicio de Sesión de Usuario e Inactividad.	237
11.4.6	Proceso de Seguridad Lógica del Sistema.....	238
12	CAPÍTULO VI - INCREMENTOS DE SOFTWARE.....	240
12.1	Incrementos de Software	241
12.2	Primer Incremento	242
12.2.1	Descripción Extendida de Casos de Uso.	242
12.2.1.1	Casos de Uso de Gestión de Tipos de Establecimiento.....	242
12.2.2	Diseño de Interfaces.....	247
12.2.2.1	Interfaces de Gestión de Tipos de Establecimiento.....	247
12.2.3	Diagrama de Secuencia.	249
12.2.3.1	Diagrama de Secuencia de Gestión de Tipos de Establecimiento.....	249
12.2.4	Diagramas de Actividad con la Notación BPMN.	252
12.2.4.1	Diagramas de Actividad de Gestión de Tipos de Establecimiento.....	252
12.2.5	Diagramas de Clases del Primer Incremento (Establecimientos, Dependencias y Empleados).	254
12.2.6	Diseño de Casos de Prueba.....	255
12.2.6.1	Diseño de Casos de Prueba de Integración de Subsistemas.	255
12.2.7	Ejecución de Pruebas.	257
12.2.7.1	Ejecución de Casos de Prueba de Integración de Subsistemas.....	257
12.2.8	Resumen de Resultados de Pruebas.....	259
12.2.8.1	Suite de Pruebas de Tipo de Establecimiento.....	259
12.3	Segundo Incremento	260
12.4	Tercer Incremento.....	260
12.5	Cuarto Incremento	261
12.6	Quinto Incremento	261
12.7	Sexto Incremento.....	262
12.8	Séptimo Incremento.....	263
13	CAPÍTULO VII - PLAN DE IMPLEMENTACIÓN	264
13.1	Alcance de la Implementación.....	265
13.2	Requerimientos de la Implementación.	266
13.2.1	Recursos Tecnológicos	266
13.2.1.1	Requerimientos de Hardware.	266



13.2.1.2	Requerimientos de Software.....	268
13.2.2	Recurso Humano	272
13.2.2.1	Perfiles del Recurso Humano	272
13.2.3	Arquitectura.....	276
13.3	Estrategia de Implementación.	277
13.4	Actividades de Implementación.	278
13.4.1	Planeación de la Implementación.....	278
13.4.1.1	Recurso Humano a Utilizar	279
13.4.2	Organización de la Implementación.....	279
13.4.2.1	Recurso Humano a Utilizar	280
13.4.3	Ejecución de la Implementación.	280
13.4.3.1	Recurso a Utilizar.....	283
13.4.3.2	Costos Basados en Horas Hombres.....	285
13.4.3.3	Resumen de Costos basados en Horas Hombres.....	287
13.4.3.4	Costo por Materiales.....	287
13.4.3.5	Total de Costos.....	287
13.4.4	Control de la Implementación.....	288
13.4.4.1	Recurso Humano Involucrado.....	288
13.5	Cronograma de Implementación.....	289
13.6	Control de la Implementación	291
13.6.1	Metodología de Control de la Implementación	291
13.6.1.1	Formularios de Control	291
13.6.1.2	Índices de Control de Avance.....	297
14	CONCLUSIONES.....	300
15	RECOMENDACIONES	302
16	GLOSARIO TÉCNICO.....	303
17	BIBLIOGRAFÍA	307
17.1	Libros.....	307
17.2	Páginas Web.....	307
17.3	Otros Documentos.....	309



1 Introducción

El Ministerio de Salud (MINSAL) está conformado por un conjunto de dependencias entre las cuales se encuentra la red de Hospitales y Unidades Comunitarias de Salud Familiar (UCSF) a Nivel Nacional que brindan Servicios de Salud a gran parte de la población salvadoreña.

Los Servicios de Salud son el conjunto de actividades que buscan responder a las necesidades en materia de Salud de la población, que involucra promover, proteger, recuperar y rehabilitar la Salud. Para que los Servicios de Salud sean eficaces, todos los departamentos y servicios de apoyo en un establecimiento de Salud, deben brindar oportunamente las condiciones y requerimientos que de ellos se demandan, para ofrecer una atención de calidad a los pacientes.

En este sentido, el Área de Mantenimiento General y los Departamentos de Mantenimiento Regional y Local tienen el rol y la responsabilidad de conservar los equipos e instalaciones médico-hospitalarias en condiciones apropiadas de uso para el cuidado de la Salud de los pacientes.

El Área de Mantenimiento General, así como los departamentos de mantenimiento del Nivel Regional y Local realizan un gran esfuerzo en el control y gestión de los servicios de mantenimiento, los cuales son: Preventivo y Correctivo. Un servicio de mantenimiento inicia por medio de una solicitud de servicio de mantenimiento (la cuál puede ser generada por una Unidad Comunitaria de Salud Familiar, un Hospital o una Región), un plan de mantenimiento preventivo o un contrato con empresas externas. Estos elementos dan origen a una orden de trabajo que es asignada a un técnico, quién será el encargado de resolver la orden de trabajo, que finaliza con el cierre de la misma y el ingreso de diversos costos involucrados.

Los servicios de mantenimiento, juegan un papel importante en la prestación de los servicios de salud a lo largo de todos los Establecimientos de Salud a Nivel Nacional, e incluso a los Establecimientos de índole administrativa, debido a que estos permiten el correcto funcionamiento del equipo médico-hospitalario, mobiliario, instalaciones e infraestructura.

Es así como nace la necesidad de contar con un Sistema Informático que apoye la planificación y control de los procesos de servicio de mantenimiento, para el nivel Superior, Regional y Local; y que a la vez se integre con otros Sistemas pertenecientes al MINSAL para una correcta coordinación. Esto ha dado origen a uno de los más grandes proyectos de automatización, debido a su complejidad y la gran cantidad de consideraciones que involucra en cuanto a conocimiento y seguimiento de los procesos de negocio, flexibilidad, seguimiento de estándares, seguridad e integración con otros sistemas.

El presente documento, comprende el desarrollo el Sistema Informático para planeación y gestión de los procesos de servicio de mantenimiento del Área de Mantenimiento General, Nivel Regional y Local del Ministerio de Salud, desde su concepción, presentando las antecedentes que han dado origen al proyecto, la planeación y metodología a utilizar durante el proyecto; así como el desarrollo del mismo, que cubre las etapas de Análisis y Determinación de Requerimientos, Diseño y Construcción. El resultado final es un Sistema Informático que cumple con los requerimientos y necesidades especificadas, funcional y con la aceptación de los usuarios, junto con la documentación necesaria.



La automatización de los procesos de servicio de mantenimiento, brindará apoyo al Área de Mantenimiento General y los diferentes departamentos de mantenimiento (Regional y Local), permitiendo la planificación y gestión de los servicios de mantenimiento de los establecimientos de la Red Nacional de Salud del Ministerio de Salud de El Salvador, brindando mayor eficiencia, control de costos, información útil y sobre todo contribuyendo a la sociedad salvadoreña, permitiendo que los Servicios de Salud sean brindados de forma oportuna, segura y eficiente.

A lo largo de proyecto, se utiliza una gran variedad de conocimientos y técnicas que son desarrolladas en la carrera de Ingeniería de Sistemas Informáticos, estas son integradas y ampliadas de tal forma, que se presenta un documento de alta calidad académica, profesional y técnica, constituyendo incluso, una guía y fuente de información para posteriores trabajos académicos y profesionales. Algunos de los conocimientos y técnicas aplicados son: Técnicas de Diagnóstico y Formulación de Problemas, Análisis y Diseño con Enfoque Orientado a Objetos, Modelado del Sistema con UML 2.0 y Modelado de Procesos de Negocio con la notación BPMN, Metodología de Desarrollo de Software (estudio y selección de la misma), Diseño Arquitectónico, Programación Orientada a Objetos y utilización de estándares en todos los aspectos del proyecto.

El presente trabajo, esta enriquecido con diversas investigaciones y estudios, que conlleva la utilización de estándares y técnicas innovadoras: Estándar *IEEE 830-1998* para la Especificación de Requerimientos de Software, que establece una guía ordenada y organizada para especificación de las características y funcionalidades del Sistema a desarrollar; la adopción de la Guía del PMBOOK 4, un estándar internacional que brinda los lineamientos a seguir para que el proyecto se realice con éxito en términos de calidad, tiempo y requerimientos del usuario, dentro del cual se destaca el "Acta de Constitución del Proyecto" que brinda formalidad y compromiso entre las partes; la aplicación de un Ciclo de Vida de Desarrollo de Software Incremental, seleccionado a través del estudio de las características propias del proyecto, que al final del mismo se observaron los claros beneficios y aportes, demostrando que no necesariamente se debe seguir el Ciclo de Vida Tradicional en Cascada; la aplicación del Control de Versiones de Software que facilita la comunicación entre los desarrolladores y la integración de código y utilización de un Sistema de Seguimiento de Errores (Bug Tracking System) junto a las Pruebas de Software para garantizar la calidad del software, por medio del seguimiento y solución de los defectos o errores encontrados.



2 Objetivos

2.1 Objetivo General

Desarrollar un Sistema Informático para planeación y gestión de los procesos de servicio de mantenimiento del Área de Mantenimiento General, Nivel Regional y Local del Ministerio de Salud, para brindar un servicio eficiente, permitiendo así una pronta respuesta, dando como resultado el correcto funcionamiento y mantenimiento del equipo médico, planta física y mobiliario, mecánica general y electromecánica de las diferentes dependencias.

2.2 Objetivos Específicos

- Analizar los procesos de solicitud de servicio, órdenes de trabajo y planificación de servicios de mantenimiento del Nivel Superior, Regional y Local.
- Determinar los requerimientos informáticos, operativos y de desarrollo que deberá satisfacer el sistema informático.
- Diseñar y construir el Sistema Informático que cumpla con los requerimientos establecidos.
- Realizar las pruebas necesarias al Sistema Informático para garantizar que este cumpla con los requerimientos establecidos, sea funcional y esté libre de errores.
- Elaborar los manuales de usuario, administración, de instalación y técnico, para facilitar el uso, instalación, actualización y mantenimiento del Sistema Informático.
- Diseñar un plan de implementación para la puesta en marcha del Sistema Informático.



3 Justificación

El Área de Mantenimiento General de la Unidad de Conservación y Mantenimiento del MINSAL realiza diferentes tipos de actividades, su principal función como su nombre lo indica, es la de brindar servicios de mantenimientos a las diferentes dependencias que conforman el MINSAL mediante las secciones que la conforman (Planta Física y Mobiliario, Electromecánica, Equipo Médico y Mecánica General), atendiendo un total de 30 Hospitales, 5 regiones (central, metropolitana, paracentral, occidental y oriental) y 402 Unidades Comunitarias de Salud Familiar (UCSF)¹; en ellas se desea tener un control detallado de cada uno de los procesos que se realizan.

El Sistema Informático SIM ayudará a la planificación y control de los servicios de mantenimiento, el seguimiento de las órdenes de trabajo, el manejo dinámico de un inventario técnico útil, con una estructura que permita la flexibilidad y extensibilidad a nuevas características y nuevos equipos, herramientas y repuestos, así también, de la administración de rutinas de trabajo para los mantenimientos preventivos (mensual, bimensual, trimestral, semestral, anual, etc.) y correctivos que se realizan en las diferentes dependencias del MINSAL, con el fin de fortalecer y facilitar la gestión y provisión de servicios de conservación y mantenimiento, en forma oportuna y eficiente, y por medio de ello, proveer el adecuado funcionamiento de la infraestructura, instalaciones, equipos y demás recursos materiales y tecnológicos, que se requieren para la eficiente prestación de servicios en las diferentes dependencias de Nivel Superior, Regional y Local, ofreciendo así servicios de Salud con calidad a la población.

Actualmente sólo el Área de Mantenimiento General atiende un promedio de 3600 solicitudes de servicio anuales, y tomando en cuenta que El Ministerio de Salud atiende aproximadamente el 74% de la población en su red hospitalaria², con una tendencia a la alza, se estima que el número de solicitudes de servicio de mantenimiento crecerá proporcionalmente, por lo que es de gran importancia hacer un buen uso de los recursos que el Área o Departamento de Mantenimiento tiene a su disposición, para poder cubrir así la demanda de solicitudes que este tiene. Para lograr este objetivo, es necesario la implementación del Sistema Informático SIM, que permita realizar una mejor planificación y gestión de los servicios de mantenimiento que se brindan.

¹ Ministerio de Salud (octubre 2012). *Redes Integrales e Integradas de los Servicios de Salud*. Recuperado de https://new.paho.org/els/index2.php?option=com_docman&task=doc_view&gid=684&Itemid=99999999

² Para más información ver Anexo 13 – Impacto Social del Documento de Anteproyecto que se encuentra dentro del directorio /Documentos/1 – Anteproyecto/ del DVD anexo.



4 Importancia

El servicio de Salud es uno de los servicios con mayor demanda en El Salvador. Este servicio es brindado por diferentes instituciones públicas y privadas dentro de las que se encuentran: Ministerio de Salud (MINSAL), Instituto Salvadoreño del Seguro Social (ISSS), Instituto Salvadoreño de Bienestar Magisterial (ISBM), Salud Militar y Seguros de Salud Privados (individuales y/o colectivos). El Ministerio de Salud le da cobertura a un total del 74% de la población salvadoreña³, tomando como base el porcentaje de cobertura para el año 2010 y según el estimado de Proyecciones Nacionales del Ministerio de Economía y Censos Nacionales⁴, se estima que para el año 2013 el total de población al que el MINSAL le dará cobertura será de 4,653,786 millones de salvadoreños.

Para poder atender la gran demanda de servicios hospitalarios, el Ministerio de Salud ha establecido las Redes Integrales e Integradas de Servicios de Salud (RIISS), que se componen de 3 niveles para atender a la población salvadoreña; en el primer nivel se encuentran 402 Unidades Comunitarias de Salud Familiar (UCSF), en el segundo nivel 11 Hospitales básicos, 14 departamentales y 2 regionales, y en el tercer nivel 3 Hospitales nacionales de tercer nivel⁵.

Uno de los problemas que actualmente afectan el servicio de Salud en el MINSAL es la falta de recursos en equipo médico necesario para poder atender a la población salvadoreña demandante de dicho servicio, en su Red Nacional de Unidades Comunitarias de Salud Familiar (UCSF) y Hospitales, por lo cual es de vital importancia que el equipo que se encuentra disponible, esté en condiciones óptimas para apoyar los diferentes servicios de Salud, por lo que se requieren servicios de mantenimiento con cortos tiempos de respuestas.

La falta de un servicio de mantenimiento o tiempos de respuestas inadecuados conllevan a no poder brindar servicios de Salud requeridos por la población salvadoreña, lo que se puede observar en el caso reciente que se dio en el Hospital Zacamil⁶, que debido a problemas con equipos de lavandería se dejó de brindar servicios hospitalarios de cirugía, oftalmología, etc., en un Hospital que atiende 1.8 millones de pacientes anuales. Debido a esto, los servicios de mantenimiento juegan un papel importante en la prestación de servicios de Salud, ya que la no disponibilidad del equipo médico, eléctrico y mecánico a nivel nacional conlleva enormes problemas al MINSAL y a la población salvadoreña que se atiende.

El desarrollo e implementación de un Sistema Informático que apoye la planeación y gestión de los diferentes servicios de mantenimiento será de gran ayuda a una Institución que actualmente no cuenta con un control y gestión adecuado de los servicios de mantenimiento que se brindan en la

³ Para más información ver Anexo 1.- Estructura de las Redes Integrales e Integradas de Servicios de Salud, y Cuadro 21-13 del Anexo 13 – Impacto Social del Documento de Anteproyecto que se encuentra dentro del directorio /Documentos/1 – Anteproyecto/ del DVD anexo.

⁴ Ministerio de Economía y Censos Nacionales. (Mayo 2010). *Estimaciones y Proyecciones Nacionales de Población 1950 – 2050*. Obtenido de: <http://www.censos.gob.sv/util/datos/proyecciones.pdf>. Pag. 82. San Salvador.

⁵ Para más información ver Cuadro 21-12 del Anexo 13 – Impacto Social del Documento de Anteproyecto que se encuentra dentro del directorio /Documentos/1 – Anteproyecto/ del DVD anexo.

⁶ Para más información ver Caso del Hospital Zacamil del Anexo 13 – Impacto Social del Documento de Anteproyecto que se encuentra dentro del directorio /Documentos/1 – Anteproyecto/ del DVD anexo.



Institución a Nivel Nacional. Por tanto es de gran importancia para la Institución, la existencia de dicho Sistema Informático que permita consolidar, integrar y estandarizar sus servicios de mantenimiento, teniendo así un mejor control de los costos de los diferentes servicios prestados, a la vez que permitirá una correcta gestión de los servicios antes mencionados.

El Ministerio de salud al realizar la implementación de dicho sistema, se ahorra un gasto de \$ **478,360.43 USD** que son requeridos para el desarrollo del Sistema Informático, ver detalles del presupuesto y costo del proyecto en la sección 8.15 del presente documento.



5 Alcances

5.1 Resultados del Sistema Informático.

Se entregará el Sistema Informático para Planeación y Gestión de los Procesos de Servicio de Mantenimiento del Área de Mantenimiento General del Ministerio de Salud, que cumpla con los requerimientos y necesidades especificadas, que sea funcional, libre de errores, y con la aceptación de los usuarios, junto con la documentación necesaria.

El Sistema Informático estará conformado por los siguientes subsistemas:

- Administración de Roles y Usuarios
 - Gestión de Roles
 - Gestión de Usuarios
 - Gestión de Grupos
- Bitácora de Acciones en el Sistema
- Administración de Elementos del Sistema
 - Gestión de Empresas
 - Gestión de Equipos
 - Gestión de Personal
 - Gestión de Servicios
 - Gestión de Contratos
 - Gestión de Establecimientos de Salud
 - Gestión de Departamentos de Mantenimiento
- Gestión Solicitudes de Servicio
- Planificación Preventiva
 - Gestión de Planes
 - Gestión de Períodos de Rutinas
- Ordenes de Trabajo
 - Gestión de Órdenes de Trabajo
 - Gestión de Informes

Generación de Reportes.

- Bitácoras de Usuarios
- Solicitud de Servicios
- Órdenes de Trabajo
- Hoja de vida del equipo.



Sistema a los cuales se interconectará el SIM (Sistemas Externos).

- SINAB (Sistema Nacional de Abastecimiento), sistema el cual contiene información relacionada al almacén de materiales y repuestos.
- SIAF (Sistema de Activo Fijo), sistema que contiene información relacionada con el inventario de equipos que posee cada una de las dependencias del MINSAL.

Subsistemas que no se consideraran:

- Reportes gerenciales o de información necesaria para la toma de decisiones.

5.2 Características del Sistema Informático.

Con el desarrollo del Sistema Informático se brindará apoyo a los procesos de mantenimiento de las diferentes dependencias de Nivel Superior, Regional y Local del Ministerio de Salud, agilizando los procesos de mantenimiento, mediante la gestión de las solicitudes de servicio y ordenes de trabajo, que permita un mejor control de los procesos de mantenimiento correctivo, y una mejor planificación de los procesos de mantenimiento preventivo.

A continuación se detallan las características más relevantes del sistema:

- a. Permitirá la administración de roles y usuarios, con lo que se podrá tener un mejor control de los privilegios con los que cuentan los diferentes usuarios del sistema, y el nivel de información a los que estos tienen acceso.
- b. Contará con una bitácora, en la cual se registrará las acciones que los usuarios del sistema realicen.
- c. Permitirá la administración de diferentes elementos del sistema tales como: empresas, equipos, personal, servicios, contratos con terceros, dependencias, departamentos de mantenimientos.
- d. Proporcionará a determinados usuarios la realización de solicitudes de servicios a las instancias correspondientes, según el tipo de usuario.
- e. Le permitirá a los usuarios darle seguimiento al estado del proceso de los procesos de mantenimiento solicitados.
- f. Proporcionará a los usuarios la capacidad de planificación de los procesos de mantenimientos preventivos en las diferentes instancias del Ministerio de Salud.
- g. Proporcionará a determinados usuarios la gestión de las diferentes órdenes de trabajo, que permita atender las diferentes solicitudes de servicios.
- h. Permitirá a los técnicos ingresar los informes pertenecientes a los servicios de mantenimiento, los cuales formarán parte de la hoja de vida del equipo.



5.3 Productos Entregables

5.3.1 Institución.

1. Documento de trabajo de graduación impreso y empastado.
2. CD o DVD que incluye el Código Fuente del Software, Manual de Usuario, Manual de Administración, Manual de Instalación, Manual Técnico, Plan de Implementación, Script de la Base de Datos y las aplicaciones necesarias para el correcto funcionamiento del sistema.

5.3.2 Universidad.

1. Documento de trabajo de graduación impreso y empastado.
2. CD o DVD que incluye el documento de trabajo de graduación, Código Fuente del Software, Manual de Usuario, Manual de Administración, Manual de Instalación, Manual Técnico, Plan de Implementación, Script de la Base de Datos y las aplicaciones necesarias para el correcto funcionamiento del sistema.

6 Limitaciones

Se ha analizado que el Área de Mantenimiento General del Ministerio de Salud tiene la disposición de trabajar y colaborar en la realización del proyecto, por lo tanto no se tienen limitaciones.



7 Capítulo I

Generalidades

Con el fin de brindar al lector un conocimiento previo del entorno de negocios sobre el cual se desarrolla el proyecto, se presenta una serie de temas, términos y elementos que conforman las generalidades.

Es importante para un buen seguimiento y comprensión del documento, la definición de los acrónimos, abreviaturas y términos utilizados en el mismo. De igual manera, antes de entrar en materia, el Marco Teórico y el Marco Legal son una importante y buena fuente de información para comprender la situación y el contexto del proyecto.

El Marco Teórico, sustenta el proyecto a realizar, exponiendo aquellos conceptos, teorías e investigaciones, que de forma general, brindan un marco de referencia del estudio del proyecto. El Marco Legal presenta una base legal sobre la cual se desarrolla el proyecto, tales como, manuales y lineamientos.

Con estos elementos y los Antecedentes de la organización en cuestión, el lector tendrá un preámbulo completo para continuar con las siguientes secciones del documento.



7.1 Acrónimos, Abreviaturas y Definiciones Generales.

Las definiciones, acrónimos y abreviaturas son descripciones que permiten exponer de manera clara y exacta el significado de un término, con el fin de brindar al lector un conocimiento previo de los mismos, para una mejor comprensión del documento. Esta sección se ha dividido en dos apartados: Acrónimos y Abreviaturas Generales, y Definiciones Generales. Las definiciones técnicas se presentan en el Glosario Técnico (Sección 16).

7.1.1 Acrónimos y Abreviaturas Generales

Acrónimo Abreviatura	Definición
DTIC	Dirección de Tecnologías de Información y Comunicación del Ministerio de Salud.
MINSAL	Ministerio de Salud de El Salvador.
OMG	Por sus siglas en inglés Object Management Group, que en español es Grupo de Gestión de Objetos. Es un consorcio dedicado al cuidado y el establecimiento de diversos estándares de tecnologías orientadas a objetos.
SIAF	Sistema Informático de Activo Fijo. Es un sistema de control administrativo, para cuantificar y clasificar los bienes que posee el MINSAL.
SIM	Sistema Informático para Planeación y Gestión de los Procesos de Servicio de Mantenimiento del Área de Mantenimiento General, Nivel Regional y Local del Ministerio de Salud.
SINAB	Sistema Nacional de Abastecimientos. Es el sistema para el registro y manejo de los bienes en los almacenes del MINSAL.
SRS	Por sus siglas en inglés Software Requirements Specification, que en español es Especificación de Requerimientos de Software.
SUIS	Sistema Único de Información en Salud. Conformado por la integración de los diferentes Sistemas de Información del MINSAL.

Tabla 7-1. Acrónimos y Abreviaturas Generales

7.1.2 Definiciones Generales

Termino	Definición
Bien	Definición generalizada que hace referencia o engloba a: Equipo Eléctrico, Mecánico, Biomédico, Mobiliario, Planta Física e Infraestructura perteneciente a un determinado Establecimiento del MINSAL.
Bien Inventariable	Dícese de todo aquel Bien que puede ser registrado dentro del Inventario Técnico, como lo son: equipo médico, maquinaria, mobiliario, entre otros.
Bien No Inventariable	Todo aquel Bien que no puede ser registrado dentro de un inventario técnico, dentro de ellos se encuentran, por ejemplo: Planta Física, Instalación, entre otros.
Diseño	Trazo o bosquejo, descripción detallada de la solución propuesta. Diseño es el trabajo de aplicar cada uno de los requerimientos que se necesitan para que la solución sea aceptada por el usuario.



Termino	Definición
Framework	Palabra en ingles que significa marco de referencia. Es un conjunto estandarizado de conceptos, prácticas, criterios, librerías, programas y/o tecnologías para enfocar un tipo de problemática particular, que sirve como referencia para enfrentar y resolver nuevos problemas de índole similar.
Mantenimiento	Desde la perspectiva de Procesos de Servicio de Mantenimiento, son todas las acciones que tienen como objetivo mantener un objeto o restaurarlo a un estado en el cual pueda llevar a cabo alguna función requerida. Estas acciones incluyen la combinación de acciones técnicas y administrativas correspondientes. Esto aplica a cada uno de los equipos instalados en las diferentes dependencias del MINSAL, así como infraestructura, planta física y mobiliario.
Metodología	Es un conjunto de procedimientos ordenados, racionales y sistemáticos que son utilizados para alcanzar los objetivos que rigen el desarrollo de un proyecto.
Modelo	En caso de referirse a un Modelo de Equipo: Es la representación de un conjunto de características o especificaciones técnicas de un Bien. En caso de referirse a un Modelo en el área de informática: Se utiliza para tratar con la complejidad inherente en los sistemas. Ayuda al ingeniero a visualizar el sistema a construir, desde una determinada perspectiva o enfoque.
Nivel Local	Enfocado a las Áreas y Departamentos de Mantenimiento, está conformado por los Departamentos de Mantenimiento Internos de cada Hospital, tanto los de Segundo y Tercer Nivel (Ver definición de Segundo y Tercer Nivel de Atención en esta sección).
Nivel Regional	Enfocado a las Áreas y Departamentos de Mantenimiento, está conformado por los Departamentos de Mantenimiento Regional brindan servicio a las Unidades Comunitarias de Salud Familiar.
Nivel Superior	Enfocado a las Áreas y Departamentos de Mantenimiento, está conformado por el Área de Mantenimiento General del Ministerio de Salud.
Objetivo del Negocio	Representan requerimientos de alto nivel de la organización o del cliente que requiere el sistema.
Primer Nivel de Atención	Conformado por las Unidades de Salud, Casas de la Salud, y los Centros Rurales de Salud y Nutrición.
Reglas del Negocio	Describe las políticas, normas, operaciones, definiciones y restricciones presentes en una organización y que son de vital importancia.
Requerimiento	Es una necesidad documentada sobre el contenido, forma o funcionalidad de un servicio.
Segundo Nivel de Atención	Conformado por los Hospitales Nacionales Generales y los Hospitales Nacionales Regionales.
Sistema	Conjunto de elementos que interactúan entre sí para lograr un determinado objetivo o fin.
Software	Se conoce como software al equipamiento lógico o soporte lógico de un sistema informático, que comprende el conjunto de los componentes lógicos necesarios que hacen posible la realización de tareas específicas
Tercer Nivel de Atención	Conformado por los Hospitales Nacionales Especializados.
Usuario	Individuo que utilizará el Sistema Informático a desarrollar y que tiene roles asignados que le permite realizar determinadas acciones en el sistema.

Cuadro 7-1 Definiciones Generales.



7.2 Marco Teórico.

El marco teórico consiste en sustentar o fundamentar teóricamente el proyecto a realizar. Implica analizar y exponer aquellos conceptos, teorías e investigaciones, de forma general, que se consideren válidos para el correcto marco de referencia del estudio del proyecto⁷.

7.2.1 Establecimientos de Salud y Redes que conforman el MINSAL

7.2.1.1 Ministerio de Salud.

De acuerdo con el Art. 65 de la Constitución de la República, que define: el Estado determinará la Política Nacional de Salud y controlará y supervisará su aplicación, en donde corresponde al Ministerio de Salud Pública y Asistencia Social como la cartera de estado el dar cumplimiento a este mandato. El Art. 40, del Código de Salud establece que El Ministerio de Salud Pública y Asistencia Social⁸, es el organismo encargado de determinar, planificar y ejecutar la Política Nacional en materia de Salud; dictar las normas pertinentes, organizar, coordinar y evaluar la ejecución de las actividades relacionadas con la Salud.

7.2.1.2 Dependencias.

Establecimientos o jefaturas que dependen directamente del Ministerio de Salud.

7.2.1.3 Región⁹.

Región o Dirección Regional de Salud, es una estructura del Órgano Ejecutivo en el Ramo de Salud Pública y Asistencia Social que tiene como atribución la gestión y el control de los recursos asignados a los Sistemas Básicos de Salud Integral (SIBASI), así como, proporcionar asistencia técnica, supervisión, monitoreo y evaluación de los mismos, en el área geográfica que cubra y donde sea necesario su existencia.

7.2.1.4 Redes Integrales e Integradas de Servicio de Salud.

Están conformadas por las diferentes dependencias del MINSAL, instituciones del Sistema Nacional de Salud (SNS) y otros proveedores de Servicios de Salud que se incorporen gradualmente;

⁷ Hernández, C. R., Fernández, C., Baptista, P. (1991). *Metodología de la Investigación*. Naucalpan de Juárez, Edo. de México: McGraw - HILL INTERAMERICANA DE MÉXICO, S.A. de C.V.

⁸ Asamblea Legislativa de la República de El Salvador (mayo de 1988). *Código de Salud. Diario Oficial No 86*

⁹ Ministerio de Salud Pública y Asistencia social (2006). *Manual de organización y funciones de la región de Salud. - Manual de Organización y Funciones del Sistema Básico de Salud Integral*. San Salvador.



los cuales desarrollan su trabajo en una población definida, rinden cuentas por sus resultados y por el estado de Salud de la población, que trabajan integradamente¹⁰.

Organización de los Servicios de Salud de las RIIS¹¹.

Para la implementación de las RIIS, el MINSAL estructura los establecimientos y servicios de Salud en tres niveles de atención, los cuales se listan a continuación:

Primer Nivel de Atención:

- **Unidades Comunitarias de Salud Familiar (UCSF):** Son los establecimientos del Primer Nivel de Atención, donde se prestan servicios de Salud integrales de diferente complejidad y resolución, se clasifican en:
 - Unidad Comunitaria de Salud Familiar Básica (UCSF Básica).
 - Unidad Comunitaria de Salud Familiar Intermedia (UCSF Intermedia).
 - Unidad Comunitaria de Salud Familiar Especializada (UCSF Especializada).

- **Equipos Comunitarios de Salud Familiar (Ecos Familiares):** Constituyen el componente básico y la puerta de entrada a las RIIS para la implementación del Modelo de atención integral en Salud con enfoque familiar y comunitario, garantizando la continuidad de la atención integral, a través de la red de servicios de atención en el ciclo de vida en la persona, familia y comunidad. Los Ecos se clasifican en:
 - Ecos Familiares.
 - Ecos Especializados.

Segundo y Tercer Nivel de Atención: El Segundo y Tercer Nivel de Atención en Salud son brindados por la Red Pública de Hospitales del país, lo cual permite la continuidad de la atención médico quirúrgica especializada a los pacientes que así lo requieran, sean referidos o no, por el primer nivel de atención.

Como respuesta a la implementación del Modelo de atención integral en Salud con enfoque familiar y comunitario, el MINSAL ha categorizado a los Hospitales de acuerdo a la capacidad instalada, perfil epidemiológico, área de responsabilidad y conjunto de prestaciones, siendo los siguientes:

- Para el Segundo Nivel de Atención:
 - Hospitales Básicos de Segundo Nivel.
 - Hospitales Departamentales de Segundo Nivel.
 - Hospitales Regionales de Segundo Nivel.
- Para el Tercer Nivel de Atención:

¹⁰ Ministerio de Salud (2012). *Manual de Organización y Funciones de las RIIS San Salvador*. Recuperado de: http://asp.salud.gob.sv/regulacion/pdf/manual/Manual_de_Organizacion_y_Funciones_de_las_RIIS.pdf

¹¹ Para más información sobre la estructura de la RIIS y los establecimientos que la componen ver Anexo 1 – Estructura de las Redes Integrales e Integradas de Servicios de Salud, del documento de Anteproyecto dentro del directorio /Documentos/1 – Anteproyecto/ del DVD anexo.



- Hospitales Nacionales de Tercer Nivel: Son los establecimientos de referencia nacional, que están categorizados para la atención de usuarios que requieran servicios altamente especializados en las áreas de emergencia, consulta externa y hospitalización. Cada uno de estos Hospitales tienen definido un conjunto de prestaciones y especialidades asignadas de acuerdo al perfil epidemiológico nacional, contando para ello con el equipo diagnóstico y de apoyo necesario.

7.2.2 Documentos Requeridos en los Servicios de Mantenimiento.

7.2.2.1 Solicitud de Servicio.

Documento diseñado para la realización de peticiones formales de servicios de mantenimiento al Área de Mantenimiento General y departamentos de mantenimiento de Nivel Regional y Local. El jefe del servicio solicitante la elabora y la hace llegar al jefe de mantenimiento, el cual la revisa y decide si la solicitud procede o no¹².

7.2.2.2 Orden de trabajo.

Documento a través del cual se lleva control del trabajo de mantenimiento y se contabiliza los costos asociados al mismo. La orden de trabajo puede tener como insumo una solicitud de servicio, o puede ser generada a partir del mantenimiento preventivo.

7.2.2.3 Orden de Requisición.

Documento emitido al almacén para solicitar repuestos y materiales después de haber realizado una inspección y diagnóstico al equipo, maquinaria o instalación.

7.2.2.4 Vale de Salida.

Documento que contiene el detalle y costo de los materiales y repuestos que han sido solicitado a través de la orden de requisición, el cual se anexa a la orden de trabajo.

7.2.2.5 Inventario Técnico de Equipo.

Es un registro descriptivo permanente de las principales características de los equipos, sobre el cual se basa la planeación, programación, adquisición y control de partes, y la ejecución de otras acciones operativas propias del servicio de mantenimiento.

¹² Ministerio de Salud. (1998). *Manual de Procedimientos Estandarizados*. San Salvador.



7.2.2.6 Programa Anual de Mantenimiento Preventivo.

Es la planificación y registro de las actividades del mantenimiento preventivo en la que se detallan frecuencia y tiempos para su ejecución. El jefe de mantenimiento es el encargado de elaborarlo, ya sea directamente o a través de los coordinadores de cada sección.

7.2.2.7 Rutinas de Mantenimiento Preventivo.

Es la guía para la ejecución de acciones técnicas de los procedimientos propios del mantenimiento preventivo sobre equipos e instalaciones de un Hospital, con el objeto de obtener la máxima eficiencia y producción del equipamiento existente. Estas son programadas por el jefe de mantenimiento a través del programa anual de mantenimiento preventivo, y ejecutadas por él técnico¹³.

7.2.2.8 Ficha de Vida.

Es el conjunto de Informes de Actividades de Mantenimiento, que contienen la información básica y específica de cada acción de mantenimiento y/o reparación realizada sobre los equipos. Mediante estos informes se puede determinar y/o decidir con el transcurso del tiempo, el estado físico-funcional del equipo, necesidad de descarte o reemplazo, análisis de costo/beneficio, etc.

7.2.2.9 Informe de Actividades de Mantenimiento.

Documento en el que se registran los datos referentes al uso de recurso humano, materiales, repuestos, entre otros, que han sido utilizado en los procesos de mantenimiento relacionados a una orden de trabajo.

¹³ Ministerio de Salud. (1998). *Manual de Procedimientos Estandarizados*. San Salvador.



7.3 Marco Legal

El Reglamento Interno del Órgano Ejecutivo, Artículo 16 Numeral 3, establece la facultad de los Ministros y Viceministros (en este caso del MINSAL) de aprobar y hacer ejecutar los Manuales de Organización; así mismo el Artículo 67, establece que el Manual de Organización y Funciones determinará la estructura administrativa, el funcionamiento de cada unidad, las atribuciones de cada empleado, las relaciones con otros organismos, normas de procedimiento y demás disposiciones administrativas necesarias.

La Ley del Sistema Básico de Salud Integral, Artículo 24 inciso 2, establece que el Órgano Ejecutivo en el Ramo de la Salud Pública y Asistencia Social elaborará los manuales y normas técnicas sanitarias que sean necesarios para el logro del objeto de esta Ley.

En base a la facultad que se le establece al MINSAL, para la elaboración del proyecto se tiene como base legal los siguientes manuales y lineamientos:

7.3.1 Manual de Organización y Funciones de la Gerencia General de Operaciones.

Este manual constituye parte de la base legal del proyecto, ya que enuncia los objetivos, estructura organizativa y funciones de la Gerencia General de Operaciones como tal y de las Unidades, Áreas y Secciones que la conforman, así como las relaciones de trabajo entre las mismas.

7.3.2 Manual de Procedimientos de la Unidad de Conservación y Mantenimiento del Ministerio de Salud.

Este manual constituye parte de la base legal del proyecto, ya que clasifica y describe las actividades de los diferentes procesos de cada una de las cuatro áreas que conforman la Unidad de Conservación y Mantenimiento: Servicios Generales, Activo Fijo, Mantenimiento General y Transporte.

7.3.3 Manual de Organización y Funciones de la Dirección de Tecnologías de Información y Comunicaciones del Ministerio de Salud.

Este manual constituye parte de la base legal del proyecto, el cual establece de forma ordenada y sistemática la organización y funcionamiento de la Dirección, Unidades y Áreas que la conforman, así como el objetivo de proveer la plataforma de tecnologías de información y comunicaciones idónea para la consecución de los objetivos estratégicos institucionales.



Dentro de las unidades que destacan dentro de la DTIC, está la Unidad de Sistemas de Información, cuya función principal es construir los módulos que conforman el Sistema Único de Información en Salud e integrar los diferentes sistemas legados de forma armónica y alineada a los objetivos institucionales, que constituirá una de las políticas del proyecto.

7.3.4 Lineamientos Técnicos para la Administración de Activo Fijo del Ministerio de Salud.

El Artículo 41 numeral 4 del Código de Salud, establece que corresponde al Ministerio de Salud organizar, reglamentar y coordinar el funcionamiento y las atribuciones de todos los servicios técnicos y administrativos de sus dependencias; así mismo el Artículo 28 del Reglamento de Normas Técnicas de Control Interno del MINSAL, establece que La Unidad de Conservación y Mantenimiento, será la responsable de diseñar el Sistema de Control Interno sobre activos y definir los procedimientos para proteger y conservar el activo fijo del MINSAL, especialmente aquellos que tengan mayor grado de vulnerabilidad, siendo responsabilidad de los directores y todas las jefaturas del Nivel Superior, Regional y Local, la aplicación de dichos procedimientos; y el Artículo 30 del mismo Reglamento, establece que La Unidad de Activo Fijo, definirá los procedimientos para la constatación física de los inventarios, los traslados, baja o descarga del equipo y mobiliario y demás procesos concernientes al activo fijo.

Por tanto, los Lineamientos Técnicos para la Administración de Activo Fijo, constituyen parte de la base legal del proyecto, los cuales definen las bases para la administración y control de los bienes muebles e inmuebles propiedad del MINSAL, unificando criterios de contenido para la realización de las funciones y procesos.

7.3.5 Manual de Organización y Funciones de las RIIS.

El Código de Salud, Artículo 40, establece que el MINSAL tiene la facultad de crear y ejecutar las normas o manuales para organizar y coordinar las actividades y funciones de las diferentes dependencias del mismo. Y considerando que a través de la Reforma al Sistema de Salud, se han creado las Redes Integrales e Integradas de Servicios de Salud (RIIS), con el propósito de tener un abordaje poblacional más inmediato, dinámico y equitativo.

Este manual constituye una de las bases legales para el proyecto, el cual presenta las directrices de organización y funcionamiento que conforman las Redes Integrales e Integradas de Servicios de Salud (RIIS).



7.3.6 Lineamientos Técnicos para el Mantenimiento, Administración y Desarrollo de Tecnologías de Información y Comunicaciones.

Los Lineamientos Técnicos para el Mantenimiento, Administración y Desarrollo de Tecnologías de Información y Comunicaciones (TIC's), establecen los criterios y responsabilidades necesarias para garantizar el buen desempeño y sostenibilidad de las plataformas de TIC's del MINSAL y sus dependencias; y uno de sus objetivos es establecer los procesos para la administración, análisis, diseño, codificación, implementación y soporte de los diferentes sistemas de información, incluyendo aquellos desarrollados por terceros para el MINSAL y sus dependencias.

Todo el personal que utiliza elementos de tecnologías de información y comunicaciones en las dependencias del MINSAL, está sujeto al cumplimiento de los criterios y responsabilidades establecidos en el documento de Lineamientos Técnicos. Por tanto, estos Lineamientos constituyen parte de la base legal del proyecto.

7.3.7 Otros Instrumentos Legales

Los manuales y lineamientos anteriormente descritos, se encuentran amparados bajo los siguientes instrumentos legales:

- Constitución de la República de El Salvador.
- Reglamento Interno del Órgano Ejecutivo.
- Ley del Sistema Básico de Salud Integral.
- Reglamento de la Ley del Sistema Básico de Salud Integral.
- Código de Salud.
- Normas Técnicas de Control Interno del MINSAL.



7.4 Antecedentes

El Área de Mantenimiento General se formó aproximadamente en el año 1963, su objetivo siempre ha estado dirigido a asegurar el buen funcionamiento del equipo, maquinaria e infraestructura de los establecimientos de Salud.

Desde sus inicios el Área de Mantenimiento General ha venido evolucionando, desde poseer un personal con conocimientos empíricos (década de los 60 y 70), hasta lograr tener un personal más calificado y capacitado (década de los 90), esto gracias a organizaciones como la OPS, el Gobierno de Japón, la Agencia Alemana de Cooperación GTZ, que brindaron su apoyo a la organización, logrando así la utilización de la tecnología para los procesos de mantenimiento (utilización de software para un mejor control del servicio), la estandarización de los procesos de mantenimiento (creación de manuales de procedimientos), entre otros.

Gracias a la ayuda internacional el Área de Mantenimiento General adquirió hace algunos años (década de los 90) un software que le permitió tener un mejor control de los servicios de mantenimiento que brindaba, pero el avance de la tecnología impidieron continuar utilizando dicho software, volviéndolo obsoleto.

A medida que ha transcurrido el tiempo, el Área de Mantenimiento General y los departamentos de mantenimiento (Nivel Regional y Local) han visto la necesidad de tener un mejor control de los servicios de mantenimiento que brindan, en este sentido los departamentos han realizado esfuerzos independientes por llevar su propio control de los servicios de mantenimiento, realizando implementaciones de software que les permita dicho objetivo, pero con la dificultad que estos esfuerzos no han sido realizado de forma conjunta, sino que han sido esfuerzos que brindan soluciones independientes y aisladas, siendo soluciones propias de algunas dependencias.

Como ejemplo de estos estudios y esfuerzos para automatizar los procesos de mantenimiento se pueden citar los siguientes:

Trabajo de graduación presentado en Julio del 2008 por estudiantes egresados de Ingeniería Industrial de la Universidad de El Salvador titulado: *“Propuesta de un Sistema de Mantenimiento para el Hospital Nacional Rosales”*¹⁴, el cual se realizó con la finalidad de establecer un diagnóstico sobre la problemática del mantenimiento de los equipos, maquinaria e infraestructura del Hospital, brindando con ello una propuesta integral que permita sistematizar las actividades de mantenimiento.

Otro esfuerzo es el Sistema de Mantenimiento (SIMANT) del Hospital Nacional de Ciudad Barrios, que es un software desarrollado en Microsoft Access, para la administración de los principales procesos técnicos y administrativos del Departamento de Mantenimiento del Hospital.

¹⁴ Alegría Molina, Ángel Alberto; Orellana Ramírez, Ricardo Ernesto y Zamora Orellana, Georgina de los Ángeles (2008). *Propuesta de un sistema de mantenimiento para el Hospital Nacional Rosales*. Tesis Ingeniería, Universidad de El Salvador.



El SIMANT no es un software institucional, ha sido desarrollado de forma particular, y aunque los procedimientos a automatizar y problemas a resolver son propios de un Hospital Nacional, el esfuerzo es particular y la solución está adaptada a un Hospital específico.

Teniendo en cuenta lo anterior el Ministerio de Salud ha percibido la necesidad de tener un Sistema Informático que apoye la planificación y control de los procesos de mantenimiento, para el nivel Superior, Regional y Local; y que a la vez se integre a los módulos que conforman el Sistema Único de Información en Salud.



8Capítulo II

Metodología

Metodología, es el conjunto de procedimientos racionales utilizados para alcanzar determinados objetivos que rigen una investigación, una exposición o tareas que requieren conocimientos, habilidades o estudios específicos.

La Metodología, es entonces para este proyecto, el conjunto de Métodos, Técnicas y Herramientas seleccionadas que llevaran al equipo de trabajo, a finalizar el proyecto exitosamente, alcanzando los objetivos previamente definidos, pero, no a cualquier costo, sino, de la forma más conveniente y eficiente, con el fin de lograr mejores resultados. Por tanto, la elección de una determinada Metodología, Método y Técnica comprende un estudio detallado, que proporcione la información necesaria para evaluar las alternativas presentadas y finalmente, la debida justificación de su selección.

La Metodología, es la guía que expresa la forma de llevar a cabo el proyecto, desde el inicio del mismo hasta su finalización, partiendo desde la Metodología de Recolección de Datos y las Técnicas de Diagnóstico y Formulación del Problema, pasando por las Metodología de Desarrollo de Software (una de las medulares dentro del proyecto) hasta la Metodología de Pruebas de Software y los Estándares de Documentación, entre otras.



8.1 Metodología de Recolección de Datos

Para recolectar datos se utiliza una serie de herramientas y técnicas que, en forma genérica, se denominan *instrumentos de recolección de datos*. La metodología de recolección de datos, se refiere al método utilizado para obtener los datos y se describe de la siguiente manera:

- Definir claramente lo que se busca obtener.
- Elegir los instrumentos de recolección de datos a utilizar.
- Diseño de los instrumentos de recolección de datos.
- Aplicación de los instrumentos de recolección de datos.
- Análisis e interpretación de los datos.

8.1.1 Entrevista¹⁵

La finalidad de la utilización de entrevistas, es establecer una relación directa con usuarios del negocio para recolectar testimonios orales sobre los procesos que intervienen en los servicios de mantenimiento.

Una entrevista es una conversación dirigida con un propósito específico que utiliza un formato de preguntas y respuestas. En el desarrollo del proyecto permitirá la determinación de requerimientos y la recolección de datos sobre los problemas que enfrenta el Nivel Superior, Regional y Local, con respecto a los servicios de mantenimiento que se brindan.

Este instrumento ha sido utilizado durante el desarrollo del anteproyecto para recolectar datos sobre los antecedentes, situación actual, planteamiento del problema y determinación de requerimientos.

Los pasos principales para la preparación de una entrevista se detallan a continuación:

1. *Leer los antecedentes*: El propósito es crear un vocabulario común que en un futuro permita expresar preguntas de la entrevista de una manera comprensible para el entrevistado.
2. *Establecer los objetivos de la entrevista*: Se utilizan los antecedentes que se hayan recopilado así como la propia experiencia para establecer los objetivos de la entrevista.
3. *Decidir a quién entrevistar*: Se debe incluir a gente clave de todos los niveles que vayan a ser afectadas por el sistema de alguna manera.
4. *Preparar al entrevistado*: En este paso se debe notificar con anticipación al entrevistado sobre la entrevista.
5. *Decidir el tipo de preguntas y la estructura*: Se deben escribir preguntas que abarquen las áreas clave de la toma de decisiones que se hayan descubierto al determinar los objetivos de la entrevista.

Las entrevistas han sido realizadas al personal de desarrollo de la Dirección de Tecnologías de Información y Comunicaciones, así como al personal del Área de Mantenimiento General que administre y preste los servicios de mantenimiento, ver Tabla 8-1.

¹⁵ E. Kendal, Kenneth y E. Kendal Julie (2005). *Análisis y Diseño de Sistemas (6 ed.)* México. Editorial Pearson Educación



Nombre del Área, Departamento o Sección	Cargo
Área de Mantenimiento General	Jefe Área de Mantenimiento General
	Jefe Sección de Capacitación
Dirección de Tecnologías de Información y Comunicaciones	Jefe de la Unidad de Sistemas de Información
	Analista programador

Tabla 8-1 Aplicación de las encuestas

Herramientas a utilizar para su diseño y aplicación:

- Microsoft Word 2013
- Dropbox
- Google Drive
- Google Docs
- Google Calendar
- Adobe Reader v.11.0

8.1.2 Encuesta¹⁶.

Se ha utilizado con el fin de obtener datos estadísticos sobre opiniones o hechos relevantes, que ayuden a respaldar datos recolectados en las entrevistas.

Es un instrumento de recolección de datos que permite estudiar las actitudes, creencias, comportamientos y características de personas claves en la organización que se encuentran involucradas en el desarrollo del proyecto. El uso principal es cuantificar información que se halla descubierto en la entrevista.

Para la elaboración de encuestas se llevan a cabo los siguientes pasos:

Para su elaboración se deben de tomar en cuenta las siguientes directrices:

1. Las personas que se necesita encuestar se encuentran en ubicaciones dispersas (diferentes instalaciones de la misma organización).
2. Si una gran cantidad de personas está involucrada en el proyecto es importante saber qué proporción de un grupo dado, aprueba o desaprueba una característica específica del sistema propuesto.
3. Se está haciendo un estudio preliminar y desea medir la opinión general antes de que se determine el rumbo que tomará el proyecto de sistemas.
4. Se desea tener la certeza de que en las entrevistas de seguimiento se identificará y abordará cualquier problema relacionado con el sistema actual.

Luego de tomar en cuenta las directrices antes mencionadas es necesario establecer los objetivos de las encuestas.

¹⁶ E. Kendal, Kenneth y E. Kendal Julie (2005). *Análisis y Diseño de Sistemas (6 ed.)* México. Editorial Pearson Educación



Redacción de preguntas.

En la redacción de preguntas puede incluir los tipos de preguntas abiertas y cerradas, a continuación se detalla el uso de cada una:

- **Abiertas:** Son aquellas que dejan abiertas al encuestado todas las posibles opciones de respuesta. Son particularmente adecuadas en situaciones que se desea descubrir las opiniones de los miembros de la organización sobre algún aspecto del sistema, ya sea un producto o un proceso.
- **Cerradas:** Son aquellas que limitan o cierran las opciones de respuesta disponible para el encuestado.

En el Cuadro 8-1 se da un marco de referencia sobre el uso de preguntas abiertas y cerradas.

Abiertas	Aspecto.	Cerradas
Lenta	Velocidad de redacción →	Rápida
Alta	← Descubriendo cuando hay pocos antecedentes	Baja
Alta	← Profundidad	Baja
Fácil	Facilidad de preparación. →	Difícil
Difícil	← Facilidad de análisis	Fácil

Cuadro 8-1 Marco de referencia sobre el uso de preguntas.

Diseño de encuestas.

En el diseño de encuestas, es preciso que las preguntas sean claras y concisas ya que se debe de evitar la ambigüedad. Para el diseño de encuestas se debe de considerar:

- Dejar bastante espacio en blanco (preguntas abiertas), proporcionando el espacio suficiente para escribir las respuestas.
- Facilitar la claridad en el marcado de respuestas a los encuestados.
- Mantener un estilo consistente.

Para el orden de las preguntas se debe de tomar los siguientes lineamientos:

- Colocar las preguntas más importantes para los encuestados.
- Agrupar los elementos de contenido similar.
- Incorporar primero las preguntas menos polémicas.

Aplicación de los encuestas.

La aplicación de encuestas se puede realizar de diferentes formas, esto será determinado por la organización en la cual se ha de elaborar el sistema. Entre algunas formas para aplicar la encuesta se encuentran las siguientes:

- Citar al mismo tiempo a todos los encuestados.
- Entregar personalmente los cuestionarios en blanco y recoger cuando ya estén terminados.
- Permitir a los encuestados que llenen el cuestionario por sí mismos en su trabajo y que lo dejen en alguna caja colocada en un punto central.
- Aplicar los cuestionarios a través de correo electrónico o la web.



Este instrumento se ha utilizado dentro del anteproyecto para cuantificar datos sobre el planteamiento del problema y para sustentar la información obtenida a través de las entrevistas.

En la **Tabla 8-2** se detalla el personal al cual se le realizarán las encuestas en el Anteproyecto.

Área, Departamento o Sección	Cargo
Departamentos de Apoyo y Servicios Hospitalarios del Hospital Nacional Rosales	Jefes o encargados de dicha Área, Departamento o Sección

Tabla 8-2 Aplicación de las encuestas

Herramientas a utilizar para su diseño y aplicación:

- Microsoft Word 2013
- Dropbox
- Google Drive
- Google Docs
- Google Forms
- Google Calendar
- Navegador Web Google Chrome

Investigación

Este instrumento de recolección de datos se ha utilizado con la finalidad de obtener datos cualitativos y cuantitativos que apoyen al desarrollo del proyecto.

La investigación es la acción de descubrir y analizar los datos. Es importante examinar los diferentes tipos de datos reales que ofrecen información no disponible a través de ningún otro instrumento de recolección de datos.

Este instrumento se ha utilizado dentro del anteproyecto para la recolección de datos y conocimiento de la organización sobre antecedentes, el marco teórico, marco legal, procesos, situación actual y planteamiento del problema. Esta técnica también será utilizada para estudios técnicos e investigación de temas de ingeniería, como metodologías de desarrollo de software, técnicas de análisis y diseño, entre otros.

Entre los documentos cualitativos que se utilizan están: páginas web del MINSAL y Hospitales, manuales de procedimientos, manuales de organización y funciones, lineamientos técnicos y manual de políticas. Entre los documentos cuantitativos que se utilizarán están: registros e informes emitidos por el MINSAL, la Dirección General de Estadística y Censos (DIGESTYC) y otros relacionados al desarrollo del proyecto.

Herramientas a utilizar para su aplicación:

- Microsoft Word 2013
- Dropbox
- Google Drive
- Navegador Web Google Chrome
- Adobe Reader v.11.0



Observación Directa¹⁷

La utilización de este instrumento de recolección de datos tiene por finalidad observar conductas, actividades y características de las personas involucradas en los servicios de mantenimiento y el ambiente en que se desarrollan.

La observación es la más común de las técnicas de investigación; la observación sugiere los problemas y muestra a la necesidad de la automatización de los procesos.

En el desarrollo del proyecto se ha utilizado la observación para conocer los procesos de mantenimiento que se realizan en el Área de Mantenimiento General. También se han realizado observaciones en Hospitales claves, para conocer elementos importantes de sus procesos de mantenimiento.

8.2 Técnicas de Diagnóstico y Formulación del Problema

El diagnóstico se refiere principalmente a la identificación y estudio de aquellos elementos que se desean modificar o en los que se ha detectado una situación indeseable¹⁸. La formulación del problema consiste en la definición de elementos que generan insatisfacción, que impiden o dificultan lograr un objetivo.

8.2.1 Diagrama Causa-Efecto.

Es la representación de varios elementos (causas) de un sistema que pueden contribuir a un problema (efecto).

Con esta técnica se pretende identificar las causas principales de un problema específico, ya que la naturaleza gráfica del diagrama permite organizar grandes cantidades de información sobre dicho problema. En la **Figura 8-1**, se puede observar la estructura del diagrama.

La utilización de esta técnica involucra los siguientes pasos:

1. Definición del efecto a analizar.
2. Situar el efecto a examinar en el lado derecho del diagrama enmarcado en un cuadro.
3. Identificar las principales causas que inciden sobre el efecto y situarlo dentro del diagrama.
4. Identificar para cada rama principal, otros factores específicos que puedan ser causa del efecto.
5. Verificar la inclusión de factores.
6. Analizar el diagrama, para identificar causas reales.

¹⁷ Ernesto A. Rodríguez Moguel (2003). *Metodología de la Investigación (5 ed.)* México. Colección Héctor Merino Rodríguez.

¹⁸ ABDALA KURI, J. A. (2013). Apuntes de Planeación del M.I. Recuperado el 25 de marzo de 2013, Facultad de Ingeniería UNAM: http://www.ingenieria.unam.mx/~jkuri/Apunt_Planeacion_internet/TEMAIII.2.pdf.

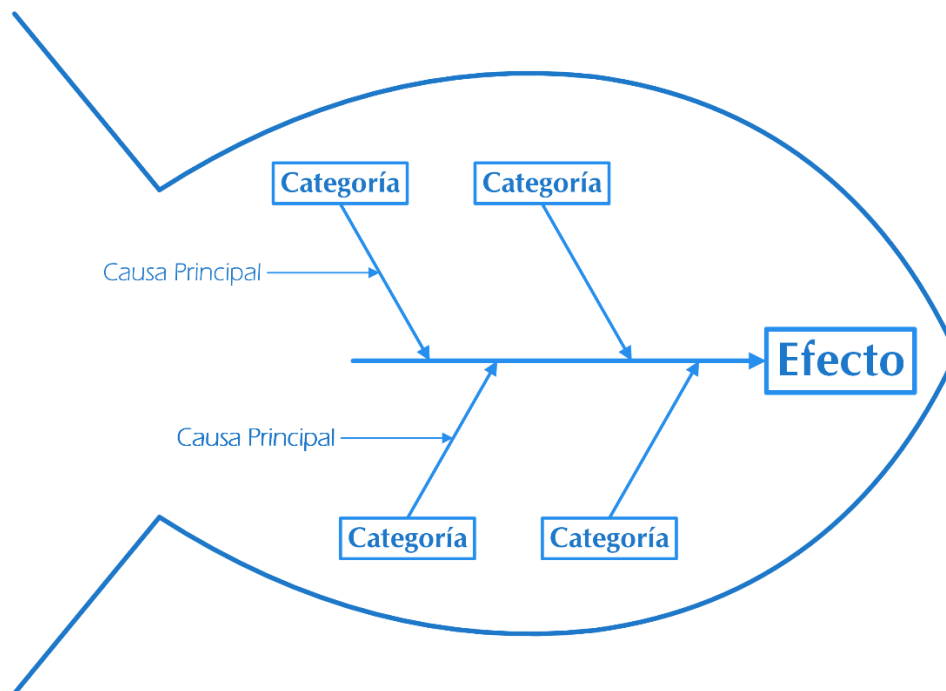


Figura 8-1 Estructura del Diagrama Causa – Efecto

Esta técnica se ha utilizado en el proyecto porque es un instrumento eficaz para el análisis de las diferentes causas que ocasionan el problema, permite que el grupo se concentre en el contenido, no en la historia del problema, ni en los distintos intereses personales (o diversos puntos de vista) de los integrantes del equipo, por tanto, es una herramienta efectiva para estudiar procesos y situaciones, y para desarrollar una solución que ataque las causas del problema.

Herramientas a utilizar para realizar el Diagrama Causa-Efecto:

- Microsoft Visio 2013.



8.2.2 Análisis FODA

Es una metodología de estudio de la situación de una organización o un proyecto, analizando sus características internas (Debilidades y Fortalezas) y su situación externa (Amenazas y Oportunidades).

Esto permitirá conformar un cuadro de la situación actual de la organización, permitiendo de esta manera obtener un diagnóstico preciso, que permita en función de ello, tomar decisiones acordes con los objetivos y políticas formuladas por la organización.

El análisis consta de cuatro pasos:

1. Análisis Externo: La organización no puede existir fuera de un entorno; así que se realiza un análisis externo, fijando las oportunidades y amenazas que el entorno presenta a la organización.
2. Análisis Interno: Se realiza un análisis interno, definiendo las fortalezas y debilidades de la organización, como aspectos o características destacadas que hacen que la organización sea fuerte o sobresalga y también aquellas que generan lo contrario.
3. Creación de la Matriz FODA: Al terminar los pasos anteriores, se tendrán definidas las Fortalezas, Oportunidades, Debilidades y Amenazas, tal como se ilustra en la **Figura 8-2**:

	Positivos Para alcanzar el Objetivo	Negativos Para alcanzar el Objetivo
Origen Interno Atributos de la org.	Fortalezas	Debilidades
Origen Externo Atributos del ambiente	Oportunidades	Amenazas

Figura 8-2 Formato Matriz FODA.

4. Determinación de las estrategias: Se crea una matriz que facilita el cotejamiento entre las amenazas y oportunidades con las debilidades y fortalezas de la organización, permitiendo determinar cuatro conjuntos definidos de alternativas estratégicas, siguiendo el formato presentado en la **Figura 8-3**:



	Fortalezas	Debilidades
Oportunidades	FO Estrategia para Maximizar Fortalezas y Oportunidades	DO Estrategia para Minimizar Debilidades y Maximizar Oportunidades
Amenazas	FA Estrategia para Maximizar Fortalezas y Minimizar Amenazas	DA Estrategia para Minimizar Debilidades y Amenazas

Figura 8-3 Matriz de estrategias

Esta técnica se ha utilizado porque permite conocer la situación real en que se encuentra una organización, así mismo, permite evaluar y diagnosticar características de la misma para que se busquen alternativas de solución que lleven a cambios viables.

Herramientas a utilizar para realizar el Análisis FODA:

- Microsoft Word 2013.

8.2.3 Enfoque de Sistemas

Es un esquema metodológico que sirve como guía para la solución de problemas, en especial hacia aquellos que surgen en la dirección o administración de un sistema, al existir una discrepancia entre lo que se tiene y lo que se desea, su problemática, sus componentes y su solución¹⁹.

El Enfoque de Sistemas requiere especificar detallada y formalmente los siguientes elementos:

- El objetivo o propósito del sistema.
- Los elementos relevantes del sistema, tanto del proceso como del control.
- Las interrelaciones relevantes del medioambiente del sistema.
- Los elementos relevantes del medioambiente del sistema.
- Las relaciones relevantes de elementos del sistema con los de su entorno: Entradas y Salidas.
- La frontera del sistema que indica el límite con el medio ambiente.

El Enfoque de Sistemas, se plasma en una estructura que se denomina Diagrama del Enfoque de Sistemas, que se observa en la **Figura 8-4:**

¹⁹ ABDALA KURI, J. A. (2013). Apuntes de Planeación del M.I. Recuperado el 25 de Marzo de 2013, Facultad de Ingeniería UNAM: http://www.ingenieria.unam.mx/~jkuri/Apunt_Planeacion_internet/TEMAII.5.pdf

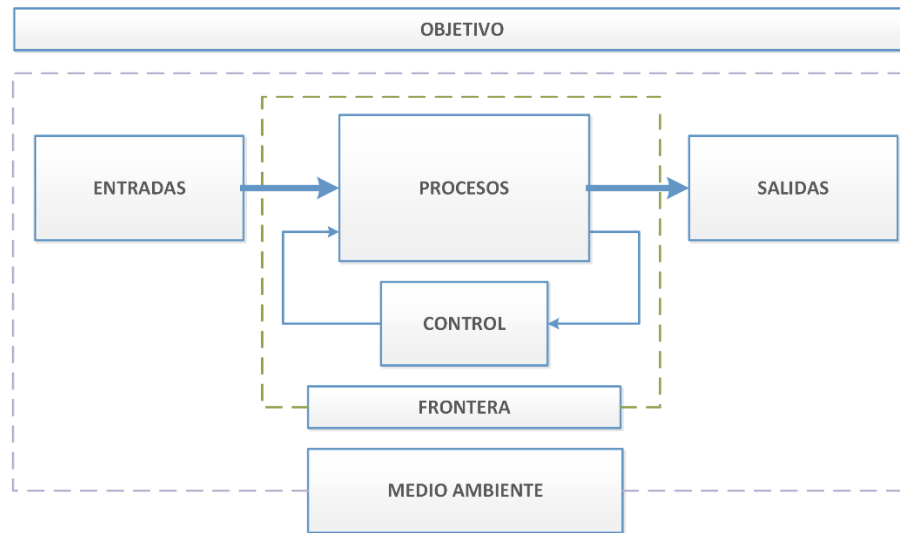


Figura 8-4. Estructura Diagrama del Enfoque de Sistemas.

Se utiliza el Enfoque de Sistemas debido a que permite observar y conocer de forma ordenada e integral a la organización, junto con los elementos con los cuales se relaciona y el ambiente en el que se desenvuelve, es decir, que con este diagrama se puede conocer de forma clara y con una amplia perspectiva la situación actual de la organización.

También permite conocer los procesos de cambio que de manera integral (a nivel humano, de recursos y procesos) serían necesarios implantar para obtener una solución. Esta herramienta se utilizó en la Situación Actual y Solución Propuesta.

Herramientas a utilizar para realizar el Diagrama del Enfoque de Sistemas:

- Microsoft Visio 2013.

8.2.4 Diagrama de Caja Negra

El Diagrama de Caja Negra permite tener una visión general del problema, el cual parte de una situación actual (Estado A) hasta una situación deseada (Estado B), pasando a través de un proceso de transformación para obtener una solución. En la **Figura 8-5** se observa la estructura del Diagrama de Caja Negra:

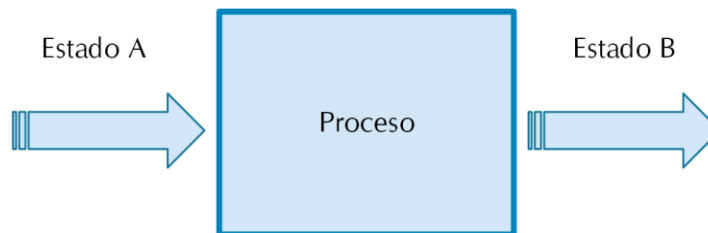


Figura 8-5. Estructura del Diagrama de Caja Negra

Este diagrama se utiliza específicamente para la formulación del problema.

Herramientas a utilizar para realizar el Diagrama de Caja Negra

- Microsoft Visio 2013.



8.3 Gestión de Proyectos

Un proyecto es un esfuerzo temporal que se lleva a cabo para crear un producto, servicio o resultado único.

La dirección de proyectos es la aplicación de conocimientos, habilidades, herramientas y técnicas a las actividades del proyecto para cumplir con los requisitos del mismo.

La metodología para la Administración de Proyectos especificada a continuación está basada en la Guía del PMBOOK 4²⁰, la cual proporciona y promueve un vocabulario común para analizar, escribir y aplicar conceptos de la dirección de proyectos.

El propósito de adoptar la Guía del PMBOOK 4, es la utilización de un estándar internacional que brinde los lineamientos a seguir para que el proyecto se realice con éxito en términos de calidad, tiempo y requerimientos del usuario.

Los procesos de dirección de proyectos se agrupan en cinco categorías conocidas como Grupos de Procesos de la Dirección de Proyectos (o grupos de procesos):

- **Grupo del Proceso de Iniciación:**

Son aquellos procesos realizados para definir un nuevo proyecto o una nueva fase de un proyecto ya existente, mediante la obtención de la autorización para comenzar dicho proyecto o fase.

- **Grupo del Proceso de Planificación:**

Son aquellos procesos requeridos para establecer el alcance del proyecto, refinar los objetivos y definir el curso de acción necesario para alcanzar los objetivos para cuyo logro se emprendió el proyecto.

- **Grupo del Proceso de Ejecución:**

Son aquellos procesos realizados para completar el trabajo definido en el plan para la dirección del proyecto a fin de cumplir con las especificaciones del mismo.

- **Grupo del Proceso de Seguimiento y Control:**

Son aquellos procesos requeridos para dar seguimiento, analizar y regular el progreso y el desempeño del proyecto, para identificar áreas en las que el plan requiera cambios y para iniciar los cambios correspondientes.

²⁰ Project Management Institute Inc. (2008). *Guía de los Fundamentos para la Dirección de Proyectos*. Pennsylvania EE.UU. PMI Book Center.



- **Grupo del Proceso de Cierre:**

Son aquellos procesos realizados para finalizar todas las actividades a través de todos los grupos de procesos, a fin de cerrar formalmente el proyecto o una fase del mismo.

Así mismo, existen Áreas de Conocimiento utilizadas para la Dirección de Proyectos. Las Áreas de Conocimiento mencionadas en el PMBOK que se adecuan al desarrollo de **este** proyecto son:

- **Gestión de la Integración del Proyecto:**

Incluye los procesos y actividades necesarios para identificar, definir, combinar, unificar y coordinar los diversos procesos y actividades de la dirección de proyectos dentro de los grupos de procesos de dirección de proyectos.

- **Gestión del Alcance del Proyecto:**

Incluye los procesos necesarios para garantizar que el proyecto incluya todo (y únicamente todo) el trabajo requerido para completarlo con éxito.

- **Gestión del Tiempo del Proyecto:**

Incluye los procesos requeridos para administrar la finalización del proyecto a tiempo.

- **Gestión de los Costos del Proyecto:**

Incluye los procesos involucrados en estimar, presupuestar y controlar los costos de modo que se complete el proyecto dentro del presupuesto aprobado.

- **Gestión de la Calidad del Proyecto:**

Incluye los procesos y actividades de la organización ejecutante que determinan responsabilidades, objetivos y políticas de calidad a fin de que el proyecto satisfaga las necesidades por la cuales fue emprendido.

- **Gestión de los Recursos Humanos del Proyecto:**

Incluye los procesos que organizan, gestionan y conducen el equipo del proyecto.

- **Gestión de las Comunicaciones del Proyecto:**

Incluye los procesos requeridos para garantizar que la generación, la recopilación, la distribución, el almacenamiento, la recuperación y la disposición final de la información del proyecto sea adecuada, oportuna y entregada a quien corresponda (interesados del proyecto o stakeholders).

- **Gestión de los Riesgos del Proyecto:**

Incluye los procesos relacionados con llevar a cabo la planificación de la gestión, identificación, análisis y planificación de respuesta a los riesgos, así como su monitoreo y control en un proyecto.



El Cuadro 8-2 refleja la relación entre los 5 Grupos de Procesos de Dirección de Proyectos y las 8 Áreas de Conocimiento de la Dirección de Proyectos, generando 38 procesos de la dirección de proyectos, el cual es tomado de la Guía del PMBOOK 4:

Áreas de Conocimiento.	Grupos de Procesos de la Administración del Proyecto.				
	Grupo de Procesos de Iniciación	Grupo de Procesos de Planificación	Grupo de Procesos de Ejecución	Grupo de Procesos de Seguimiento y Control	Grupo de Procesos de Cierre
1. Gestión de la integración del proyecto.	1.1. Desarrollar el Acta de Constitución del Proyecto.	1.2. Desarrollar el Plan para la Dirección del proyecto.	1.3. Dirigir y Gestionar la ejecución del proyecto.	1.4. Monitorizar y controlar el trabajo del proyecto. 1.5. Realizar el control integrado de cambios.	1.6. Cerrar Proyecto o Fase.
2. Gestión del Alcance del Proyecto.		2.1. Recopilar requisitos. 2.2. Definir el Alcance. 2.3. Crear EDT (Estructura del Desglose del Trabajo).		2.4. Verificar el Alcance. 2.5. Controlar el Alcance.	
3. Gestión del Tiempo del Proyecto.		3.1. Definir las actividades. 3.2. Secuenciar las actividades. 3.3. Estimar los recursos de las actividades. 3.4. Estimar la duración de las actividades. 3.5. Desarrollar el cronograma.		3.6. Controlar el cronograma.	
4. Gestión de los Costos del Proyecto.		4.1. Estimar los Costos. 4.2. Determinar el Presupuesto.		4.3. Controlar los costos.	
5. Gestión de la Calidad del Proyecto.		5.1. Planificar la Calidad.	5.2. Realizar el aseguramiento de la calidad.	5.3. Realizar el control de la calidad.	
6. Gestión de los recursos humanos del proyecto.		6.1. Desarrollar el plan de recursos humanos.	6.2. Adquirir el equipo del proyecto. 6.3. Desarrollar el equipo del proyecto. 6.4. Dirigir el equipo del proyecto.		



Áreas de Conocimiento.	Grupos de Procesos de la Administración del Proyecto.				
	Grupo de Procesos de Iniciación	Grupo de Procesos de Planificación	Grupo de Procesos de Ejecución	Grupo de Procesos de Seguimiento y Control	Grupo de Procesos de Cierre
7. Gestión de las Comunicaciones del Proyecto.	7.1. Identificar a los interesados (Stakeholders)	7.2. Planificar las comunicaciones.	7.3. Distribuir la información 7.4. Gestionar las expectativas de los interesados	7.5. Informar el desempeño.	
8. Gestión de los Riesgos del Proyecto.		8.1. Planificar la gestión de riesgos. 8.2. Identificar los riesgos. 8.3. Realizar el análisis cualitativo de los riesgos. 8.4. Realizar el análisis cuantitativo de los riesgos. 8.5. Planificar la respuesta a los riesgos.		8.6. Monitorizar y controlar los riesgos.	

Cuadro 8-2 Correspondencia entre grupos de proceso y áreas de conocimiento de la dirección de proyectos.

La aplicación correcta de estos procesos es de gran importancia para el éxito del proyecto, por lo tanto se ha de integrar estos procesos a cada una de las Etapas del Ciclo de Vida de Desarrollo del Proyecto.

Herramientas a utilizar:

- AceProject.
- Microsoft Project 2013



8.3.1 Acta de Constitución del Proyecto

El Acta de Constitución del Proyecto, es un documento que autoriza formalmente un proyecto y en el que se documenta los requisitos que satisfacen las necesidades y expectativas de los interesados. El desarrollo del Acta de Constitución corresponde con el proceso 1.1 del Cuadro 8-2 presentado en la sección anterior, y esta presenta beneficios tanto para el equipo de desarrollo como para los usuarios finales e interesados del proyecto.

Los principales beneficios son:

- Autoriza formalmente la constitución del nuevo proyecto a desarrollar.
- Define los alcances del proyecto a desarrollar.
- Define lo que el proyecto no incluye o no contempla (evitando que los usuarios realicen suposiciones erróneas o evitando malentendidos).
- La aprobación y firma de este documento, implica la conformidad de cada individuo con el mismo.
- Constituye un marco legal de referencia para la delimitación del proyecto y que define el rumbo del mismo.

Tomando en cuenta lo anterior, ha creado el Acta de Constitución del Proyecto para el desarrollo del “Sistema Informático para planeación y gestión de los procesos de servicio de mantenimiento del Área de Mantenimiento General, Nivel Regional y Local del Ministerio de Salud”²¹

8.3.2 Administración de la Comunicación del Proyecto.

8.3.2.1 Identificar a los interesados (stakeholders)

Los interesados son personas u organizaciones (por ejemplo, clientes, patrocinadores, la organización ejecutante o el público), que participan activamente en el proyecto, o cuyos intereses pueden verse afectados positiva o negativamente por la ejecución o terminación del proyecto. Los interesados también pueden ejercer influencia sobre el proyecto, los entregables y los miembros del equipo.

A continuación en la Tabla 8-3 se muestra a los *interesados claves* dentro del proyecto:

Director de Tecnologías de Información y Comunicación		
Ing. Carlos Juan Martín Pérez	cmartin@salud.gob.sv	2205-7376
Ubicación:	DTIC	

²¹ Para ver el Acta de Constitución del Proyecto del “SIM” vea Anexo 11 – Acta de Constitución del Proyecto, del documento de Anteproyecto, dentro del directorio /Documentos/1 – Anteproyecto/ en el DVD anexo.



Jefe de Área de Mantenimiento General		
Ing. Noel Antonio Ayala	nayala@salud.gob.sv	7803-7495
Ubicación:	Área de Mantenimiento General	
Jefe de la Unidad de Sistemas de Información		
Ing. Sonia Maribel Viana	sviana@salud.gob.sv	2205-7376
Ubicación:	DTIC	
Analista Programador		
Ing. Samuel Menjívar	smenjivar@salud.gob.sv	2205-7376
Ubicación:	DTIC	
Analista Programador		
Lic. Eduardo Rodríguez	erodriguez@salud.gob.sv	2205-7376
Ubicación:	DTIC	

Tabla 8-3 Interesados claves dentro del proyecto

Existen otros interesados que influyen en la realización del proyecto, los cuales se listan a continuación:

- Jefe de Departamento de Mantenimiento (Hospitales y Regionales).
- Jefe Sección de Capacitación (Área de Mantenimiento Local).
- Jefe o encargado de división, departamento o servicio (Realizan solicitudes de servicio).
- Técnicos (Área de Mantenimiento General, Hospitales y Regionales).

Clasificación de los interesados claves

Identificar el impacto o apoyo potencial que cada interesado podría generar en el proyecto, ayudará a tomar una estrategia de comunicación más cuidadosa que se acople al interesado, para poder tener una efectiva comunicación y excelente relación entre el equipo de trabajo y los interesados.

La matriz de poder/interés: agrupa a los interesados basándose en su nivel de autoridad (“poder”) y su nivel de preocupación (“interés”) con respecto a los resultados del proyecto.



		Interés	
		Bajo	Alto
Poder	Bajo	A	B Analista Programador
	Alto	C Jefe de la Unidad de Sistemas de Información	D Jefe del Área de Mantenimiento General Director de Tecnologías de Información y Comunicación

Tabla 8-4 Matriz Poder-Interés

En la Tabla 8-4 únicamente aparecen los interesados claves dentro del proyecto, debido a que los otros interesados son subordinados de estos, y se tendrán que acoplar a las indicaciones o directrices que estos les den, para apoyar la realización del proyecto.

8.3.2.2 Planificación de la Comunicación con los Interesados

El proceso Planificar la Comunicación responde a las necesidades de información y comunicación de los interesados; por ejemplo, quién necesita qué información, cuándo la necesitará, cómo le será proporcionada y por quién. Identificar las necesidades de información de los interesados y determinar una forma adecuada de satisfacer dichas necesidades, constituye un factor importante para el éxito del proyecto.

Según el PMBOOK 4; una comunicación eficaz significa que la información se suministra en el formato adecuado, en el momento justo y con el impacto apropiado. Es por ello que se definen los siguientes lineamientos de comunicación (Tabla 8-5):

Interesado	Requisito de Información	Método o Técnica de Comunicación	Etapa o Actividad de Interés
Director de Tecnologías de Información y Comunicación	Requiere información acerca del avance del proyecto y necesidades que surjan en el desarrollo del mismo	Reuniones Presenciales Correo Electrónico Carta Formal	Validación de cada incremento. Aprobación de cualquier cambio o actualización dentro del proyecto.



Interesado	Requisito de Información	Método o Técnica de Comunicación	Etapa o Actividad de Interés
Jefe Área de Mantenimiento General	Requiere información sobre el documento de requerimientos y la funcionalidad del sistema en base a los procesos a automatizar.	Reuniones Presenciales Correo Electrónico Llamada Telefónica	Todo el ciclo de vida de desarrollo.
Jefe de la Unidad de Sistemas de Información	Requiere información acerca del avance del proyecto, pero su interés está en validar el funcionamiento progresivo del Sistema Informático.	Reuniones Presenciales Correo Electrónico	Validación de Requerimientos. Elaboración del documento de diseño Validación de cada incremento.
Analista Programador	Requiere la entrega de avances del desarrollo del proyecto, sobre el cumplimiento de los requerimientos y estándares establecidos, diseño realizado y el código fuente. Debido al mantenimiento futuro que se hará al sistema.	Reuniones Presenciales Git Correo Electrónico	Todo el ciclo de vida de desarrollo.

Tabla 8-5 Comunicación con los Interesados

Todos los interesados de la DTIC, tendrán acceso a los incrementos (código fuente) del proyecto por medio de la herramienta Git, la cual proporcionará las diferentes versiones de los incrementos desarrollados.

Logística

- Cada reunión llevada a cabo tendrá una agenda y una fecha establecida.
- Cada reunión deberá tener un director quien será el responsable de gestionar la convocatoria y presentar la agenda a desarrollar por medio del secretario de comunicaciones, este a su vez será el responsable de documentar los puntos tratados, las opiniones expuestas y las decisiones acordadas.
- Así mismo habrá un cronometrador en cada reunión.

La forma para convocar a los interesados a una reunión será:

- Se notificará por medio de correo electrónico a los interesados la convocatoria de reunión por lo menos dos días antes, en donde se especificará el motivo, agenda, lugar, fecha, hora de inicio y fin de la reunión.



- Las reuniones podrán ser confirmadas por medio de otro correo electrónico y en casos especiales por medio de llamadas telefónicas.

El objetivo de las reuniones estará orientado exclusivamente a:

- Presentación y revisión del avance de los incrementos del proyecto.
- Planificación, organización de actividades del proyecto.
- Recopilación de requerimientos.
- Solución de problemas y toma de decisiones.
- Discusión de mejoras del proyecto.

Roles

Los roles en las reuniones permitirán definir funciones que cada miembro del equipo debe realizar para el éxito de la reunión, los cuales serán los siguientes:

- *Director de la reunión:* Coordinador del equipo de trabajo, organizará cada reunión en base a lo descrito en la logística, definirá la agenda y dirigirá la reunión al momento de llevarse a cabo.
- *Secretario de comunicación:* Redactará con ayuda del Director la convocatoria a reunión y enviará el respectivo correo a los interesados. Registrará las actividades desarrolladas, anotará puntos importantes y las decisiones consensuadas, los temas o puntos pendientes y tareas a seguir. Así mismo se encargará de realizar el resumen de la reunión y se las hará llegar a los asistentes.
- *Cronometrador:* Lleva el seguimiento y cumplimiento de la agenda y el tiempo dedicado a cada punto. Otra atribución será verificar la puntualidad de los miembros y su firma en la lista de asistencia, para efectos de documentación y registro.
- *Miembros:* Participarán en la reunión proactiva y constructivamente aportando conocimiento y alternativas de solución.

En la tabla Tabla 8-6, se muestra la asignación de roles que cada integrante del equipo realizará en las reuniones que se realicen.

Miembro del Equipo	Rol
Carlos Aarón Romero Delgado.	Director de la Reunión
Emerson Joaquín Minero Sánchez.	Secretario de comunicación
Alber José Romero Salgado.	Cronometrador
Caleb Rubén Rodríguez Orozco.	Miembros

Tabla 8-6 Asignación de Roles al Equipo de Trabajo.

Políticas

- Cada uno de los asistentes deberá recibir previamente la agenda a tratar en cada reunión y respetar el orden en que se han establecido los puntos y su respectiva duración.
- Al final de la reunión deberá realizarse un resumen sobre lo tratado en la agenda, problemas resueltos y/o acuerdos entre los interesados (usuarios de negocios y técnicos).



- Será un deber del secretario de comunicación, enviar a todos los participantes el resumen de la reunión con la información ya mencionada, junto con la lista de asistencia de la misma.
- Enviar la invitación o convocatoria de forma anticipada en base a lo mencionado en la logística, con la ubicación, la fecha y duración de la reunión.
- Mantener el orden y respeto entre los miembros de la reunión.
- Mantener el tiempo estipulado para cada uno de los puntos y de la duración total de la reunión.
- El líder debe garantizar que los miembros asistan con actitud positiva, flexible y proactiva, y que el tiempo de la misma sea exclusivamente para el trato de los temas en beneficio del proyecto.

Normas para elaborar la agenda

- Los puntos a tratar durante la reunión deberán priorizarse por su nivel de importancia.
- Cada uno de los asistentes deberá recibir previamente la agenda a tratar en cada reunión y respetar el orden en que se han establecido los puntos y su respectiva duración.
- Solicitar voluntarios para director, secretario y cronometrador de la reunión.
- Dedicar a los puntos a tratar sobre el avance o retraso del proyecto un máximo de 10 minutos y se recomienda que estos sean los primeros a tratar en la agenda.
- El tiempo de duración de la reunión oscilará entre 25-35 min. En caso de que sea necesario un tiempo adicional, la reunión no deberá exceder de 1 hora. De lo contrario deberán realizarse otras reuniones en diferentes días según la disponibilidad de los interesados.

8.3.3 Gestión de los Riesgos del Proyecto

Un riesgo es un evento o condición incierta que, si sucede, tiene un efecto en por lo menos uno de los objetivos del proyecto. Los objetivos pueden incluir el alcance, el cronograma, el costo y la calidad. Un riesgo puede tener una o más causas y, si sucede, uno o más impactos.

Los riesgos son inherentes a los proyectos. Será necesario realizar una gestión de riesgos eficientes para minimizar su probabilidad e impacto.

8.3.3.1 Plan de Gestión de Riesgos

En el Plan de Gestión de Riesgos se describe cómo se estructurará y realizará en el proyecto la gestión de riesgos del proyecto.

Roles Encargados de Gestionar los Riesgos

La Tabla 8-7 muestra los roles para el manejo de riesgos. Define al líder, al administrador y a los miembros del equipo de gestión de riesgos para cada tipo de actividad del plan de gestión de riesgos, y explica sus funciones.



Roles Encargados de Gestionar los Riesgos	Funciones
Líder del Proyecto	<ul style="list-style-type: none"> • Activar un riesgo (Cambiar su estado a Activo), para realizar su acción de Mitigación
Administrador de Riesgos	<ul style="list-style-type: none"> • Recibir todas las notificación de los riesgos • Analizar los riesgos para ingresarlos al Registro • Monitorear los riesgos ya Ingresados.
Equipo de Trabajo	<ul style="list-style-type: none"> • Notificar de algún riesgo que hayan encontrado. • Realizar las acciones de Mitigación de algún Riesgo.

Tabla 8-7 Roles Encargados de Gestionar los Riesgos

Clasificación de los Riesgos

La clasificación de los riesgos se lista a continuación:

- **Planificación:** Cuando alguna actividad afecte el tiempo planeado para el proyecto, moviendo la fecha de finalización de este.
- **Recurso Humano:** Cuando miembros del equipo de trabajo impiden el desarrollo del proyecto, ya sea por enfermedad o por rendimiento no adecuado. Dentro de esta categoría entra la imposibilidad de reunión con el asesor del proyecto, por no encontrarse en disponibilidad de tiempo o ubicación de llevarse a cabo.
- **Equipo Tecnológico:** Cuando el equipo de desarrollo utilizado presente fallos o en peores casos este sea robado, y esto retrase el avance del proyecto. También se incluye en esta categoría falla de servidores del MINSAL, o cualquier equipo necesario para la continuidad del trabajo.
- **Organizacional:** Cuando surgen cambios en los puestos del Ministerio de Salud, de personal clave que este apoyando el desarrollo del proyecto y esto impida el avance en las actividades.
- **Huelgas:** Cuando en el MINSAL, por su carácter público se vea involucrado en huelgas por parte de las direcciones, unidades, o departamentos que forman parte de los interesados, y esto evita la correcta continuidad del desarrollo del proyecto.
- **Características del Software:** Cuando algún requerimiento impide el desarrollo del proyecto, ya sea por falta de especificación, mal entendimiento, cambio o alguna otra problemática que impida el desarrollo.
- **Toma de Decisiones:** Cuando el desarrollo del proyecto se vea afectado por el retraso en la toma de decisiones, ya sea en cambios de requerimientos o cualquier actualización que requiera la aprobación de los interesados con poder de decisión.
- **Desastres Naturales:** Cuando el desarrollo del proyecto se vea afectado por terremotos, incendios, cambios climáticos como lluvias, huracanes, los cuales afectan la comunicación o reunión entre el equipo de trabajo y los interesados.
- **Servicios Básicos:** Cuando los servicios básicos necesarios como electricidad, internet, agua potable, entre otros, impida la continuidad del desarrollo del proyecto.



Probabilidad de Ocurrencia

La Tabla 8-8 muestra una calificación cualitativa de la probabilidad de ocurrencia de un riesgo a la cual se le ha asignado un peso, esta calificación ha sido definida por el equipo de trabajo.

Calificación	Peso	Descripción
Muy Alta	100	Probabilidad alta de ocurrencia del Riesgo, entre 80% y 99%.
Alta	80	Probabilidad alta de ocurrencia del Riesgo, entre 60% y 80%.
Mediana	60	Probabilidad mediana de ocurrencia del Riesgo, entre 40% y 60%.
Baja	40	Poca probabilidad de ocurrencia del Riesgo, entre 10% a 40%
Muy Baja	20	Muy poca probabilidad de ocurrencia del Riesgo, menos del 10%.

Tabla 8-8 Probabilidad de Ocurrencia de Riesgos

Impacto

La Tabla 8-9 muestra una calificación cualitativa del impacto de un riesgo en el proyecto, a la cual se le ha asignado un peso, esta calificación ha sido definida por el equipo de trabajo.

Calificación	Peso	Descripción
Muy Alto	100	El impacto en el proyecto ya es más de la mitad de los recursos ya planeados para el incremento de desarrollo donde se encontró el riesgo
Alto	80	El impacto en el proyecto es alto, puede ocupar el 30% o 40% más de los recursos ya planeados para el incremento de desarrollo donde se encontró el riesgo
Mediano	60	El impacto que tiene en el proyecto es notable, puede ocupar un 20% más de los recursos de lo ya planeado en el incremento de desarrollo que se encontró el riesgo.
Bajo	40	El Impacto que tendrá en el proyecto no es mucho, es decir, será como un 5% más de los recursos de lo ya planeado en el incremento de desarrollo que se encontró el riesgo.
Muy Bajo	20	Impacto insignificante para el proyecto, no es posible determinar la magnitud del mismo en el proyecto por lo pequeño que este resulta.

Tabla 8-9 Impacto de los Riesgos

Prioridades

Para la prioridad de los Riesgos es necesario llenar el formato de "Prioridad de Riesgos"²². La Información que se ingresa en el campo de Prioridad es el valor del resultado de esta fórmula:

$$\text{Prioridad} = (\text{Probabilidad de Ocurrencia} + \text{Impacto}) / 2$$

²² Para ver el formato de "Prioridad de Riesgos" vea el Anexo 12 – Formatos Utilizados en la Gestión de Riesgos, del documento de Anteproyecto, dentro del directorio /Documentos/1 – Anteproyecto/ en el DVD anexo.



8.3.3.2 Identificación de los Riesgos

Los riesgos podrán ser identificados por cualquier rol involucrado en el desarrollo del proyecto, las actividades que se deben de realizar cuando un riesgo es encontrado son:

1. Llenar el formato de "Notificación de Riesgos".
2. Entregar la notificación al encargado de la Administración de Riesgos.
3. El Encargado de la Administración de Riesgos analizará el Riesgo.

El encargado de registrar un riesgo es el Administrador de Riesgos, y las actividades que debe de realizar para registrar el riesgo son:

1. Analizar el Riesgo Notificado
2. Llenar la plantilla de "Registro de Riesgo".
3. Deberá agregar el riesgo a formato de "Identificación de Riesgos".

El formato del ID que se requiere en la Identificación de Riesgos, será el siguiente:

- **R-<AZ>-##**

Donde:

- **R**, denota que es un riesgo.
- **<AZ>**, son las iniciales de la clasificación a la que pertenece el riesgo. En caso de estar formada por una sola palabra la clasificación, la inicial de esta se repetirá para el formato.
- **##**, es un número correlativo de dos dígitos comenzando desde 01 hasta 99.

Para la Identificación de Riesgos se tendrán en cuenta las siguientes técnicas:

- Tormenta de Ideas
- Técnica Delphi.
- Entrevistas.
- Análisis Causal.

8.3.3.3 Planificación de Respuestas a los Riesgos

Existen varias estrategias de respuesta a los riesgos. Para cada riesgo, se debe seleccionar la estrategia o la combinación de estrategias con mayor probabilidad de eficacia.

Estrategias para Riesgos Negativos o Amenazas

Estas estrategias, descritas a continuación, consisten en evitar, transferir, mitigar o aceptar:

- **Evitar.** Evitar el riesgo implica cambiar el plan para la dirección del proyecto, a fin de eliminar por completo la amenaza.



- **Transferir.** Transferir el riesgo requiere trasladar a un tercero todo o parte del impacto negativo de una amenaza, junto con la propiedad de la respuesta.
- **Mitigar.** Mitigar el riesgo implica reducir a un valor aceptable la probabilidad y/o el impacto de un evento adverso.
- **Aceptar.** Esta estrategia se adopta debido a que rara vez es posible eliminar todas las amenazas de un proyecto. Esta estrategia indica que el equipo del proyecto ha decidido no cambiar el plan para la dirección del proyecto para hacer frente a un riesgo.

Estrategias para Riesgos Positivos u Oportunidades

Estas estrategias, descritas a continuación, son explotar, compartir, mejorar o aceptar.

- **Explotar.** Esta estrategia puede seleccionarse para los riesgos con impactos positivos, cuando la organización desea asegurarse de que la oportunidad se haga realidad.
- **Compartir.** Compartir un riesgo positivo implica asignar todo o parte de la propiedad de la oportunidad a un tercero mejor capacitado para capturar la oportunidad en beneficio del proyecto.
- **Mejorar.** Esta estrategia se utiliza para aumentar la probabilidad y/o los impactos positivos de una oportunidad.
- **Aceptar.** Aceptar una oportunidad consiste en tener la voluntad de tomar ventaja de ella si se presenta, pero sin buscarla de manera activa.

8.3.3.4 Monitoreo de Riesgos

Para el monitoreo de un riesgo, se deberá llenar el formato de “Monitoreo de Riesgos”²³, en el cual se manejan los estados por los que puede pasar un riesgo. A continuación se listan dichos estados:

- **Activo:** Será asignado a un riesgo que ya está registrado y se estén ejecutando las acciones correspondientes a la mitigación de este.
- **Pasivo:** Será asignado a un riesgo ya registrado y este no afecta al desenvolvimiento del proyecto dentro de la clasificación del impacto establecido.
- **En Espera de Registro:** Será asignado a un riesgo que ya fue notificado y ya se le asignó un identificador, pero todavía no se ha planificado ninguna acción para mitigar el riesgo.

²³ Para ver el formato de “Monitoreo de Riesgos” vea el Anexo 12 – Formatos Utilizados en la Gestión de Riesgos, del documento de Anteproyecto, dentro del directorio /Documentos/1 – Anteproyecto/ en el DVD anexo.



8.4 Metodología de Desarrollo de Software

La metodología de desarrollo de software define el entorno de trabajo que se utiliza para estructurar, planificar y controlar el proceso de desarrollo de software²⁴. Precisa un conjunto de actividades, acciones, tareas, fundamentos y productos de trabajo que se requieren para desarrollar software de alta calidad²⁵.

Un Modelo o Enfoque de Proceso de Desarrollo de Software no necesariamente es apropiado, conveniente o el mejor para todos los proyectos. Cada uno de los Modelos o Enfoques existentes es adecuado para determinados tipos de proyectos, es decir, que la elección de un Modelo o Enfoque se debe basar en diversas consideraciones técnicas y en características del mismo proyecto, así como del equipo que va a ejecutar el proyecto.

Para la selección de la Metodología de Desarrollo de Software y su respectivo Enfoque o Modelo se ha realizado una investigación de las Metodologías según la siguiente clasificación²⁶ (ver **Cuadro 8-3**):

Metodologías Convencionales	Metodologías Ágiles
<ul style="list-style-type: none"> • Modelo en Cascada • Modelo Incremental • Modelos Evolutivos <ul style="list-style-type: none"> ○ Prototipo ○ Espiral • Modelos Iterativos 	<ul style="list-style-type: none"> • Programación Extrema (XP) • Modelo Scrum • Proceso Unificado de Rational (RUP)

Cuadro 8-3 Clasificación de Metodologías de Desarrollo de Software.

En base a los resultados e interpretaciones obtenidos a través del estudio de las diferentes Metodologías y Modelos o Enfoques y el análisis de las características o factores del proyecto, que hacen que una Metodología sea conveniente o no de utilizar²⁷, se ha determinado utilizar la **Metodología Convencional con un Modelo o Enfoque Incremental**, para el desarrollo del software.

²⁴ CMS. (2008). *CMS Information Technology*. Recuperado el 26 de Marzo de 2013, de Center for Medicare & Medicaid Services: <https://www.cms.gov/Research-Statistics-Data-and-Systems/CMS-Information-Technology/XLC/Downloads/SelectingDevelopmentApproach.pdf>

²⁵ Pressman, R. S. (2005). *Ingeniería del software, Un Enfoque Práctico*. (6ª Edición). McGraw-Hill.

²⁶ Universidad de Oriente – Venezuela (11 de julio de 2012). *Metodología para el Desarrollo de Software*. Obtenido de Manogas Universidad de Oriente: http://wiki.monagas.uo.edu.ve/index.php/Metodolog%C3%ADas_para_el_desarrollo_de_software

²⁷ vea Anexo 7 – Análisis para Selección de Metodología de Desarrollo de Software, del Documento de Anteproyecto que se encuentra dentro del directorio /Documentos/1 – Anteproyecto/ del DVD anexo.



8.4.1 Ciclo de Vida de Desarrollo de Software Incremental.

El desarrollo incremental se basa en la idea de diseñar una implementación inicial, exponer ésta al comentario del usuario, y luego desarrollarla en sus diversas versiones hasta producir un sistema adecuado²⁸.

La **Figura 8-6** muestra de forma gráfica el desarrollo incremental, en el que se observa que las etapas de especificación, desarrollo y validación se entrelazan en vez de separarse, con una retroalimentación entre ellas, además cada incremento o versión del sistema incorpora algunas de las funciones que necesita el cliente, desarrollando así versiones intermedias hasta lograr obtener una versión final que incorpore todas las funciones especificadas por el cliente.

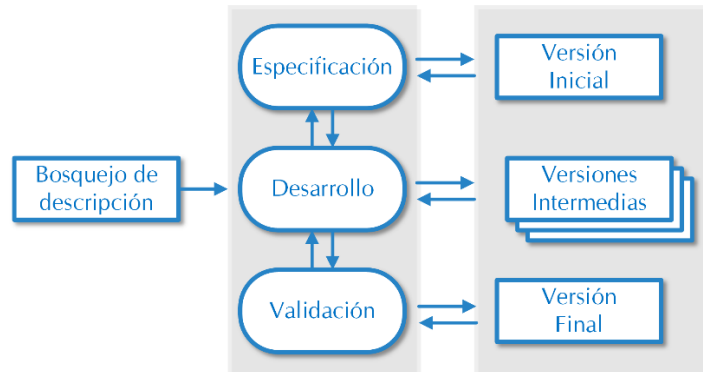


Figura 8-6 Desarrollo Incremental

Para una mejor comprensión del uso del Enfoque Incremental en el desarrollo de software descrito en la **Figura 8-6**, a continuación se presentan en la **Figura 8-7**, las etapas utilizadas en el desarrollo del software para el proyecto.

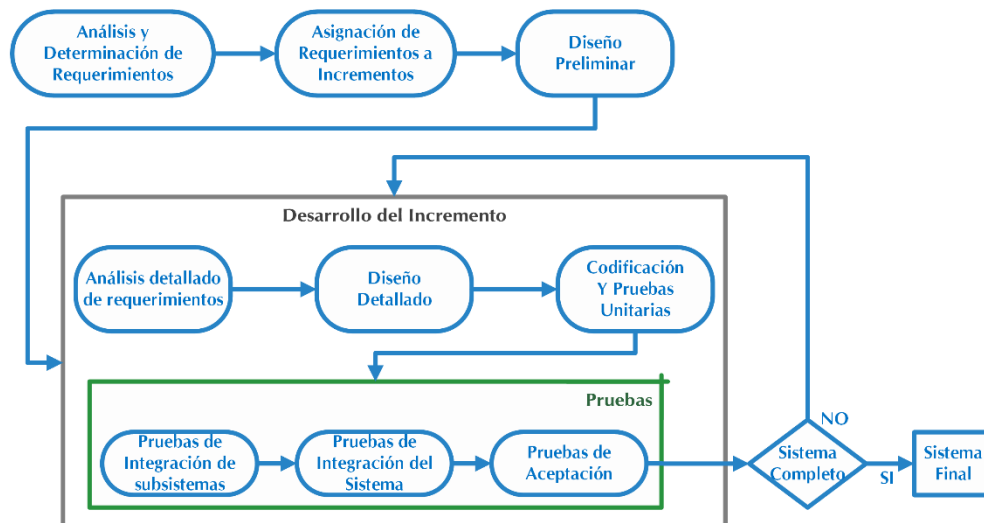


Figura 8-7 Etapas del Enfoque Incremental para el desarrollo de software.

²⁸ Sommerville, I. *Ingeniería de Software* (9a. ed.). Naucalpan de Juárez, México: Pearson Education



Etapas del Enfoque Incremental para el Desarrollo de Software:

- **Análisis y Determinación de Requerimientos:** Etapa que permite establecer el dominio del sistema, los servicios que ha de proporcionar, el desempeño requerido, las restricciones de hardware entre otros, por medio de las actividades de; obtención y análisis de requerimientos, especificación de requerimientos y validación de requerimientos, esta etapa finaliza con la creación del documento de requerimientos que es un comunicado oficial en el que se detallan los requerimientos de usuarios y del sistema.

A continuación se describe las actividades que componen esta etapa:

- **Obtención y Análisis de Requerimiento:** Actividad en el que se establecen los requerimientos del sistema mediante las técnica de recolección de datos descritas en la sección 8.1 de este documento, y que conlleva un conjunto de tareas tales como: descubrimiento, clasificación, organización, priorización, negociación y especificación de requisitos.
 - **Especificación de requerimientos:** Actividad que consiste en la transcripción de la información recopilada durante la actividad de obtención y análisis de requerimiento, en un documento en el que se definen los requerimientos de usuarios y del sistema.
 - **Validación de requerimientos:** Actividad que verifica que los requerimientos sean realistas, coherentes y completos, con el objetivo de seleccionar solo aquellos requerimientos que el usuario realmente quiere, esto a través de diferentes tipos de comprobaciones como: comprobaciones de validez, consistencia, totalidad, realismo y verificabilidad.
- **Asignación de Requerimientos a Incrementos:** Etapa en la que se asignando a cada incremento un conjunto de requisitos o servicios que deben de cumplir. La asignación de servicios dependerá de la prioridad de éstos.
 - **Diseño Preliminar:** Etapa en la que se realiza una descripción de la estructura del sistema a implementar, los modelos y la estructura de datos utilizados por el sistema, las interfaces con otros componentes del sistema, entre otros. En esta etapa no se realiza un diseño detallado del sistema, el diseño total del sistema se ha realizado de manera iterativa a medida se han desarrollado los incrementos. Las actividades que involucra el desarrollo de esta etapa son las siguientes: diseño arquitectónico, diseño de interfaz, diseño de componentes y diseño de base de datos, para una mejor comprensión de cómo se relacionan estas actividades ver **Figura 8-8**. El resultado final de esta etapa comprende el documento de diseño.

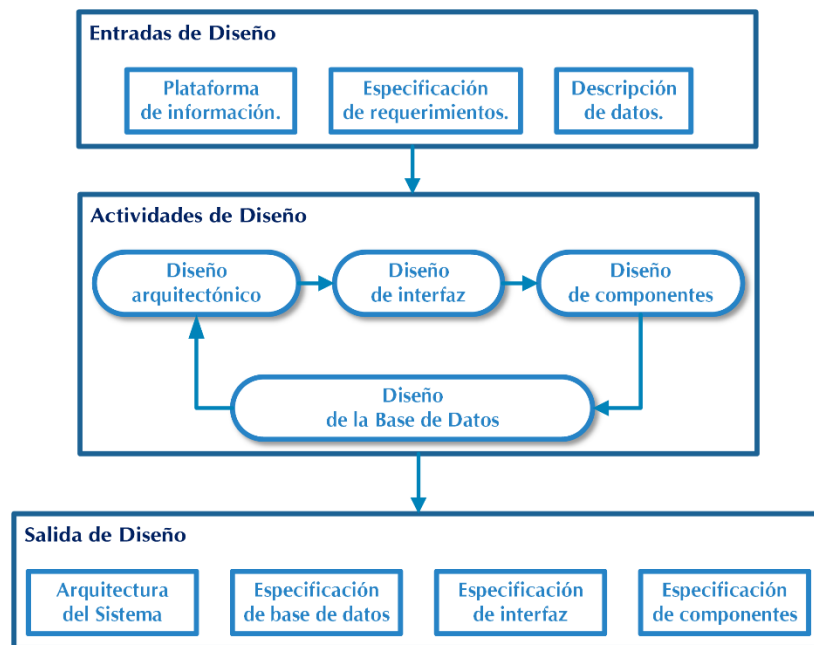


Figura 8-8 Modelo del proceso de diseño.

A continuación se describe las actividades que componen esta etapa:

- Diseño arquitectónico: Esta actividad identifica la estructura global del sistema, los principales componentes (subsistemas), sus relaciones y como se distribuyen.
 - Diseño de interfaz: Esta actividad comprende la definición de las interfaces entre los componentes del sistema, permitiendo crear interfaces precisas que permitan la utilización de componentes sin necesidad de saber cómo se han implementado.
 - Diseño de componentes: En esta actividad se diseña cómo funcionará cada componente de software, es decir, convertir el diseño de datos, interfaces y arquitectura en una representación intermedia que se pueda transformar fácilmente en código.
 - Diseño de base de datos: Actividad en donde se diseñan las estructuras del sistema de datos y cómo se representará en una base de datos.
- Análisis Detallado de Requerimientos: Etapa del desarrollo del incremento cuya finalidad es el esclarecimiento de requerimientos que pudieron quedar incompletos en la etapa de análisis y determinación de requerimientos, permitiendo realizar un análisis más profundo de los requerimientos.
 - Diseño Detallado: Etapa del desarrollo del incremento que se basa en la etapa de Diseño Preliminar para realizar un diseño más profundo de los diferentes requerimientos del sistema, esta etapa involucra el uso de técnicas y herramientas descritas en la sección 8.6 y 8.7 de este documento.
 - Programación: Etapa del desarrollo del incremento que tiene como insumo el documento de requerimiento y de diseño, que son utilizados como base para la creación del incremento, las



técnicas y herramientas que se utilizan en esta etapa se describen en el apartado 8.8, esta etapa tiene como resultado una versión de software con una mayor funcionalidad que la obtenida en el incremento anterior.

- Pruebas: Etapa del desarrollo del incremento en la que se verifica y valida que el sistema cumple tanto con sus especificaciones, así como con las expectativas del usuario. Esto se logra mediante diferentes pruebas que se ejecutan en el desarrollo de cada incremento, en la cual se realizan procesos de comprobación, inspección y revisión. Entre las actividades que conforman esta etapa se encuentran: prueba de desarrollo, prueba del sistema y prueba de aceptación, tal como se muestra en la Figura 8-9:

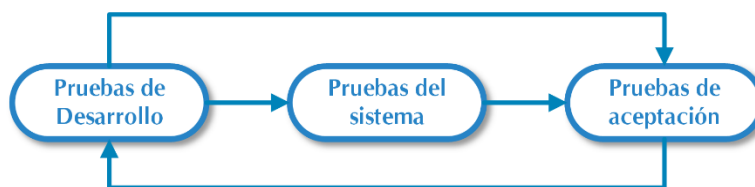


Figura 8-9 Actividades de la etapa de pruebas.

A continuación se describen las actividades que componen esta etapa:

- Pruebas de desarrollo: En esta actividad se pone a prueba los componentes que constituyen los subsistemas, cada componente se prueba de manera independiente.
- Pruebas del sistema: Actividad que tiene por finalidad descubrir errores que resulten de la interacción no anticipada entre componentes y problemas de interfaz de componentes, así como demostrar que cubre con sus requisitos y las propiedades del sistema.
- Pruebas de aceptación: Actividad en la que se utilizan datos de pruebas simulados, los cuales revelan los errores y las omisiones en la definición de requerimientos del sistema.

Para más información sobre la metodología de pruebas y técnicas a utilizar vea la sección 8.9 de este documento.

8.5 Metodología de Análisis de Requerimientos

La Especificación de Requerimientos de Software (en adelante SRS, por sus siglas en ingles), es una descripción completa del comportamiento del sistema que se va a desarrollar. Incluye un conjunto de casos de uso que describe todas las interacciones que tendrán los usuarios con el software. Además de los casos de uso incluye requerimientos no funcionales como restricciones de diseño, de implementación, y atributos de calidad.

Para realizar la Especificación de Requerimientos de Software se tomaran como base las practicas recomendadas en el estándar *IEEE 830-1998, Recommended Practice for Software Requirements Specifications*.



La utilización del estándar *IEEE 830-1998* para la Especificación de Requerimientos de Software permite al equipo de desarrollo, describir y definir con claridad el resultado de la investigación, recolección y análisis de requerimientos, al mismo tiempo que los stakeholder podrán contemplar y revisar lo que ellos esperan obtener, visualizándolo de forma ordenada y organizada. Siendo este uno de los principales documentos de comunicación y entendimiento entre el equipo de desarrollo y el cliente. Sirviendo además como el punto de partida, para las demás Etapas del Proyecto y del Ciclo de Vida de Desarrollo de Software.

A continuación, se muestra la estructura del Documento de Especificación de Requerimientos de Software para el Proyecto que se desarrollará basado en estándar *IEEE 830-1998*:

1. Introducción

- 1.1. Propósito
- 1.2. Alcance del Proyecto
- 1.3. Definiciones, Acrónimos y Abreviaturas
- 1.4. Referencias
- 1.5. Resumen del Documento

2. Descripción general

- 2.1. Perspectiva del Producto
- 2.2. Funciones del Producto
- 2.3. Características de los Perfiles de Usuarios
- 2.4. Restricciones
- 2.5. Suposiciones y Dependencias

3. Especificación de Requerimientos

- 3.1. Interfaces Externas
 - 3.1.1. Interfaces del Sistema
 - 3.1.2. Interfaces de Usuario
 - 3.1.3. Interfaces de Software
 - 3.1.4. Interfaces de Comunicación
 - 3.1.5. Requerimientos de Adaptación del Sitio
- 3.2. Requerimientos Funcionales
- 3.3. Otros Atributos y Requerimientos

4. Anexos

A continuación se describe brevemente cada uno de los apartados de estructura del SRS presentadas anteriormente:

Introducción

Esta sección proporciona una visión general de todo el documento de Especificación de Requerimientos de Software.

Propósito

Apartado en el que se define el propósito que se pretende alcanzar con la utilización del SRS y a quién está dirigido.

Alcance del Proyecto

En esta sección se identifica el producto a construir o desarrollar, y se asigna un nombre. Se explica lo que el sistema ha de realizar, los objetivos, características y los beneficios que se espera alcanzar con el futuro sistema.

**Definiciones, Acrónimos y Abreviaturas**

Sección en la que se define aquí los términos, acrónimos y abreviaturas que se utilizarán en el desarrollo de la Especificación de Requerimientos de Software.

Referencias

Apartado en el que se presenta una lista completa de todos los documentos referenciados en la Especificación de Requerimientos de Software.

Resumen del Documento

Subsección que brinda al lector un panorama del contenido de las secciones del documento.

Descripción General

Sección en la que se describen todos los factores que afectan al producto y a sus requerimientos, brindando con ello un contexto de los requerimientos, pero no la descripción en sí de estos. Los detalles de los requerimientos se describen en una sección posterior, detallándolos y haciendo más fácil su comprensión.

Perspectiva del Producto

Subsección en la que se relaciona el futuro sistema con otros productos, brindando una descripción de cómo el sistema a desarrollar formará parte de otro sistema mayor, las interfaces que tendrá y el modo de operación que tendrán sus distintos usuarios.

Funciones del Producto

Subsección que proporcionará un resumen de las funciones principales que el software debe llevar a cabo. Las funciones deben estar organizadas de manera que el cliente o cualquier otra persona las entiendan perfectamente.

Característica de los Perfiles de Usuarios.

Apartado en el que se indica los tipos de usuarios y/o actores que han de utilizar la aplicación, especificando los requerimientos de conocimientos y experiencias que deben de cumplir para una correcta utilización de la aplicación.

Restricciones

Sección en la que se indica los tipos de restricciones que se deben de cumplir en el desarrollo del producto, tales como: restricciones, políticas y estándares de la organización, restricciones de diseño, limitaciones de hardware, seguridad, protocolos de comunicación, interfaces con otras aplicaciones, etc.

Suposiciones y Dependencias

Apartado en el que se presentan los factores que pueden afectar los requerimientos, por causa de modificación de dichos factores o por falta de cumplimiento de estos, por ejemplo: Asumir la disponibilidad de un Sistema Operativo o la presencia de cierta información requerida en el desarrollo del proyecto, el cambio o la falta de alguno de estas suposiciones generará un impacto en el proyecto puede afectar el cumplimiento exitoso de las actividades del mismo.



Especificación de Requerimientos

Sección del SRS que contiene todos los requerimientos hasta un nivel de detalle suficiente, que permita al Equipo de Trabajo diseñar un sistema que satisfaga dichos requerimientos, y que permita el diseño de las pruebas que ratifiquen que el sistema cumple con las necesidades requeridas.

Esta sección debe ser perfectamente entendible por el cliente y por personas sin conocimiento técnico relacionado a la Informática. Es importante tener en cuenta que esta sección se identifica cada uno de los requerimientos mediante un código o sistema de numeración.

Interfaces Externas

Subsección que definirá los requerimientos que se encuentran relacionados a interfaces de usuario e interfaces con otros sistemas (hardware y software), así como a interfaces de comunicaciones y los requerimientos de adaptación del sitio.

Requerimientos Funcionales

El estándar IEEE 830, permite la organización de esta subsección de múltiples formas y sugiere algunas maneras de hacerlo, brinda la oportunidad de utilizar cualquier otra, justificando suficientemente la utilización de ésta.

La forma de organización que se utilizará será por *Caso de Uso*, en la que se especifican los requerimientos funcionales agrupados por los elementos que conforman el entorno de negocio, tales como: Establecimientos, Dependencias, Solicitud de Servicio, Órdenes de Trabajo, Ficha de Vida entre otros.

Para este proyecto, se especificarán requerimientos generales los cuales son aquellas acciones expresables normalmente como “El sistema deberá”, y requerimientos específicos donde se hará uso de la notación de casos de uso.

Otros Atributos y Requerimientos

Apartado en la que se detallan atributos como la fiabilidad, mantenibilidad, seguridad, mecanismos de acceso restringido (inicio de sesión), usuarios autorizados a realizar ciertas tareas críticas, mecanismos de respaldos, características del hardware, entre otros.

Anexos

Sección en la que se colocará cualquier información que ayude a clarificar o respaldar, lo que se ha detallado en las secciones de la especificación de requerimientos.



8.6 Técnicas de Análisis y Diseño

Para la realización del análisis de requerimientos y el diseño del sistema informático se han de utilizar las siguientes técnicas:

8.6.1 Enfoque Orientado a Objetos

El Análisis y Diseño utilizando el Enfoque Orientado a Objetos consiste en situar el dominio de un problema y su solución lógica dentro de la perspectiva de los objetos²⁹ (entidades paralelas de entes del “mundo real”). Y como ha sido mencionado, el Enfoque Orientado a Objetos se utilizará en la etapa de análisis y diseño.

La adopción del Enfoque Orientados a Objetos permite una mejor aproximación a la realidad (puesto que los objetos se asocian con entidades del mundo real), hasta el punto de representar cómo las diferentes partes del sistema funcionan conjuntamente de forma dinámica. Otra de las ventajas de los sistemas orientados a objetos es que son más fáciles de darles mantenimiento que aquellos desarrollados utilizando enfoques funcionales. Esto se debe a que los objetos incluyen datos o atributos y las operaciones para modificar dicho atributos, pueden entenderse y modificarse como entidades independientes.

El Análisis Orientado a Objetos se centra en la investigación del problema, no en la manera de definir una solución. Durante el Análisis Orientado a Objetos se procura ante todo identificar y describir los objetos o conceptos dentro del dominio del problema.

El Diseño Orientado a Objetos presenta una solución lógica (cómo el sistema cumple con los requerimientos), por lo que es necesario contar con descripciones detalladas y de alto nivel de la solución y saber cómo satisface los requerimientos y las restricciones.

Para desarrollar un sistema orientado a objetos desde el concepto hasta el diseño detallado, se siguen de forma general los siguientes pasos³⁰:

- 1) Comprender y definir el contexto y las interacciones externas con el sistema.
- 2) Diseñar la arquitectura del sistema.
- 3) Identificar los objetos principales del sistema.
- 4) Desarrollar modelos de diseño.
- 5) Especificar interfaces.

Herramientas a utilizar:

El Lenguaje de Modelado a utilizar para el Enfoque Orientado a Objetos será UML en su versión 2.0.

²⁹ Larman, C. (1999). UML y Patrones, Introducción al Análisis y diseño Orientado a Objetos (1ª ed.). Pearson Education.

³⁰ Sommerville, Ian. (2011). Ingeniería de Software. (9ª ed.). Naucalpan de Juárez, México: Pearson Educación.



8.6.2 UML 2.0

Se definirá la especificación de requerimientos y el diseño del sistema por medio del lenguaje de modelado UML (Lenguaje Unificado de Modelado), que es un lenguaje que permite modelar, visualizar, especificar, construir y documentar los artefactos de un sistema, desde una perspectiva orientada a objetos³¹. Constituye el lenguaje de modelado estándar más utilizado para el análisis, diseño, implementación y documentación de sistemas orientados a objetos. Se utiliza la versión 2.0 de UML, la cual fue adoptada por el OMG (Consortio dedicado al cuidado y el establecimiento de diversos estándares de tecnologías orientadas a objetos, tales como UML, XMI y CORBA.) en 2005.

UML proporciona varios tipos de diagramas a través de una notación específica, que cuando se utilizan dentro de una metodología dada, aumentan la facilidad de comprensión de una aplicación en desarrollo. Las razones principales para la utilización de UML son las siguientes:

- a) UML proporciona un lenguaje común para los analistas de negocios y desarrolladores, esto permite la comunicación y comprensión.
- b) UML permite modelar visualmente cómo opera un negocio. Quién, Cómo y Qué actividad realiza, está representado en términos de símbolos y diagramas.
- c) UML es utilizado para diagramar software desde una perspectiva orientada a objetos. Los objetos que componen los modelos UML, reflejan mejor las entidades del mundo real, por lo que son más flexibles e intuitivos.
- d) UML tiene muchos tipos de diagramas que permiten representar la información desde diferentes puntos de vista.
- e) UML ayuda a enfocarse en el cliente, debido a que gira en torno a los casos de uso de negocio, que muestran cómo un proceso de negocio entrega valor al cliente.
- f) UML ayuda a obtener requisitos del sistema a través de las descripciones de las relaciones entre los componentes, los actores y otras entidades.

El resultado final de la aplicación de UML, es el modelado de la lógica y los procesos del negocio, que resuelve un problema de negocio específico, responde a las necesidades reales de la organización, y ofrece un valor agregado a los clientes.

Diagramas a utilizar

Por su nivel de importancia y el apoyo que dan en la representación de los requerimientos y los procesos de negocio del proyecto a desarrollar³², se utilizarán los siguientes diagramas:

³¹ López, P. H. (Ciclo II - 2010). Tecnología Orientada a Objetos, Unidad I - Desarrollo de Sistemas Orientado a Objetos. Universidad de El Salvador, San Salvador, El Salvador.

³² Para conocer más sobre los Diagramas UML y el nivel de importancia ver Anexo 8 – Diagramas UML 2.0, del Documento de Anteproyecto que se encuentra dentro del directorio /Documentos/1 – Anteproyecto/ del DVD anexo.



- Diagramas de Comportamientos
 - Diagrama de Casos de Uso
 - Diagramas de Interacción
 - Diagrama de Secuencia
- Diagramas Estructurales
 - Diagrama de Clases
 - Diagrama de Componentes

Resumen de la nomenclatura de los Diagramas de UML.

Diagrama de Casos de Uso

El diagrama de casos de uso actúa como foco en la descripción de los requisitos del usuario. En él se describen las relaciones entre los requisitos, los usuarios y los componentes primarios. En los diagramas de casos de uso se resume quién utiliza la aplicación o sistema y qué puede hacer con ella. A continuación se presenta un bosquejo del diagrama de casos de uso (ver **Figura 8-10**) con su respectivo significado (ver **Tabla 8-10**).

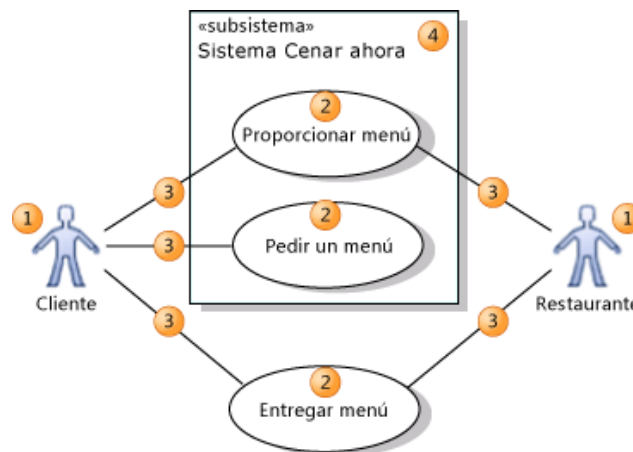


Figura 8-10 Diagrama de Casos de Uso.

Nº	Elemento	Descripción
1	Actor	Representa un usuario, organización o sistema externo que interactúa con la aplicación o sistema.
2	Caso de Uso	Representa las acciones que realizan uno o varios actores para conseguir un objetivo determinado.
3	Asociación	Indica que un actor forma parte de un caso de uso.
4	Subsistema	Sistema o aplicación (o alguno de sus componentes) en el que se está trabajando. Los casos de uso que un sistema o componente admiten aparecen dentro de su rectángulo.

Tabla 8-10 Descripción de Elementos del Diagrama de Casos de Uso.



Diagrama de Secuencia

El diagrama de secuencia muestra una interacción, que representa la secuencia de mensajes entre las instancias de clases, componentes, subsistemas o actores. El tiempo fluye hacia abajo en el diagrama y muestra el flujo de control de un participante a otro. A continuación se presenta un bosquejo del diagrama de secuencia (ver **Figura 8-11**) con su respectivo significado (ver **Tabla 8-11**).

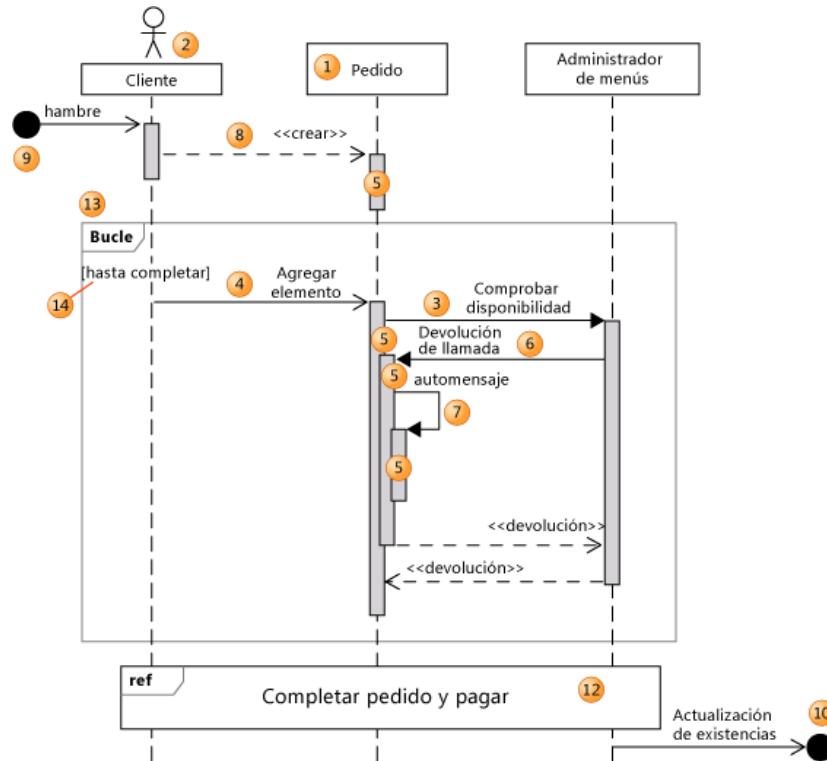


Figura 8-11 Diagrama de Secuencia.

Nº	Elemento	Descripción
1	Línea de Vida	Una línea vertical que representa la secuencia de eventos que se producen en un participante durante una interacción, mientras el tiempo avanza. Este participante puede ser una instancia de una clase, componente o actor.
2	Actor	Un participante que es externo al sistema que está desarrollando.
3	Mensaje Síncrono	El remitente espera una respuesta a un mensaje síncrono antes de continuar. El diagrama muestra la llamada y el retorno. Los mensajes síncronos se utilizan para representar llamadas de función ordinarias dentro de un programa, así como otros tipos de mensaje que se comportan de la misma manera.



Nº	Elemento	Descripción
4	Mensaje Asíncrono	Un mensaje que no requiere una respuesta antes de que el remitente continúe. Un mensaje asíncronico muestra sólo una llamada del remitente. Se utiliza para representar la comunicación entre subprocesos diferentes o la creación de un nuevo subproceso.
5	Incidencia de Ejecución	Un rectángulo sombreado vertical que aparece en la línea de la vida de un participante y representa el período durante el que el participante está ejecutando una operación.
6	Mensaje de Devolución de Llamada	Un mensaje que vuelve a un participante que está esperando la devolución de una llamada anterior. La aparición de ejecución resultante aparece encima de la existente.
7	Auto mensaje	Un mensaje de un participante a sí mismo
8	Crear Mensajes	Un mensaje que crea un participante.
9	Mensaje Encontrado	Un mensaje asíncronico de un participante desconocido o no especificado.
10	Mensaje Perdido	Un mensaje asíncronico a un participante desconocido o no especificado.
11	Comentario	Un comentario se puede adjuntar a cualquier punto de una línea de vida.
12	Uso de Interacción	Agrega una secuencia de mensajes que se definen en otro diagrama.
13	Fragmento Combinado	Una colección de fragmentos. Cada fragmento puede agregar uno o más mensajes. Existen distintos tipos de fragmentos combinados.
14	Protección de Fragmentos	Se puede utilizar para enunciar una condición relativa a si el fragmento se producirá

Tabla 8-11 Descripción de Elementos del Diagrama de Secuencia.

Diagrama de Clases

En este diagrama se describe el objeto y las estructuras de información que se utilizan en una aplicación, tanto de forma interna como en la comunicación con los usuarios. Esta información se describe sin hacer referencia a ninguna implementación concreta. Las clases y relaciones se pueden implementar de muchas maneras, por ejemplo, en tablas de bases de datos, en nodos XML o en composiciones de objetos de software. A continuación se presenta un bosquejo del diagrama de clases (ver **Figura 8-12**) con su respectivo significado (ver **Tabla 8-12**).

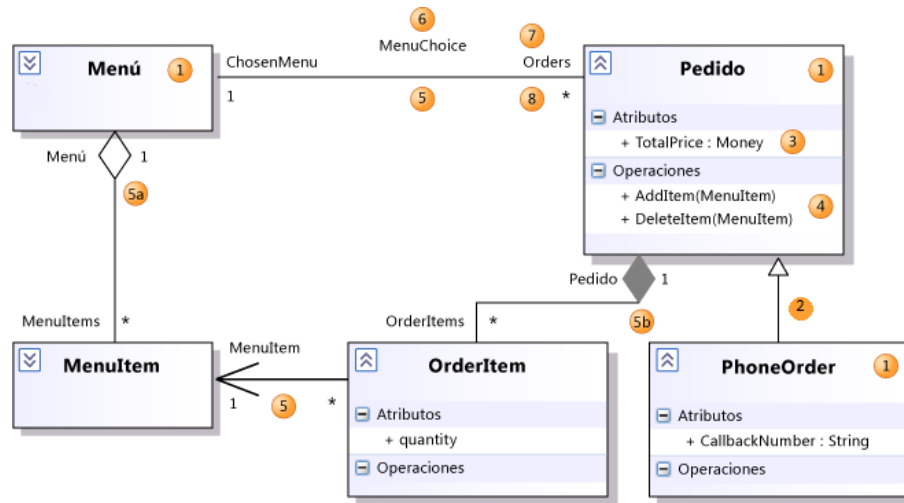


Figura 8-12 Diagrama de Clases.

Nº	Elemento	Descripción
1	Clase	Definición de objetos que comparten ciertas características estructurales y de comportamiento.
1	Clasificador	Nombre general de una clase, interfaz o enumeración. Los componentes, casos de uso y actores también son clasificadores.
2	Generalización	El clasificador específico hereda parte de su definición del clasificador general. El clasificador general se encuentra en el extremo del conector de la flecha. El clasificador específico hereda los atributos, las asociaciones y las operaciones
3	Atributo	Valor con tipo que se adjunta a cada instancia de un clasificador.
4	Operación	Método o función que pueden realizar las instancias de un clasificador.
5	Asociación	Relación entre los miembros de dos clasificadores
5a	Agregación	Asociación que representa una relación de propiedad compartida.
5b	Composición	Asociación que representa una relación parte/todo.
6	Nombre de Asociación	Nombre de una asociación. El nombre puede quedarse vacío.
7	Nombre de Rol	Nombre de un rol, es decir, un extremo de una asociación. Puede utilizarse para hacer referencia al objeto asociado
8	Multiplicidad	Indica cuántos de los objetos de este extremo se pueden vincular a cada objeto del otro.

Tabla 8-12 Descripción de Elementos del Diagrama de Clases.



Diagrama de Componentes

El diagrama de componentes muestra los elementos de diseño de un sistema de software. Permite visualizar con más facilidad la estructura general del sistema y el comportamiento del servicio que estos componentes proporcionan y utilizan a través de las interfaces. A continuación se presenta un bosquejo del diagrama de componentes (ver Figura 8-13) con su respectivo significado (ver Tabla 8-13).

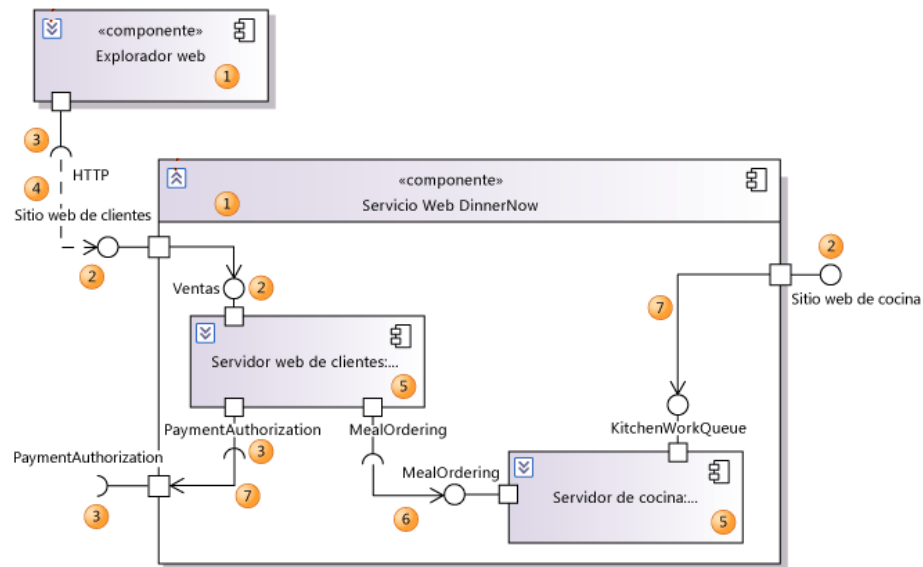


Figura 8-13 Diagrama de Componentes.

Nº	Elemento	Descripción
1	Componente	Elemento de funcionalidad del sistema reutilizable. Un componente proporciona y utiliza el comportamiento a través de las interfaces y puede hacer uso de otros componentes.
2	Puerto de interfaz proporcionada	Representa un grupo de mensajes o llamadas que un componente implementa y que otros componentes o sistemas externos pueden utilizar
3	Puerto de interfaz necesaria	Representa un grupo de mensajes o llamadas que el componente envía a otros componentes o sistemas externos.
4	Dependencia	Se puede utilizar para indicar que una interfaz necesaria de un componente se puede satisfacer mediante una interfaz proporcionada de otro.
5	Parte	Atributo de un componente cuyo tipo normalmente es otro componente



Nº	Elemento	Descripción
6	Ensamblado de Elementos	Conexión entre los puertos de la interfaz necesaria de un elemento y los puertos de la interfaz proporcionada de otro.
7	Delegación	Vincula un puerto a una interfaz de uno de los elementos del componente. Indica que los mensajes enviados al componente se administran en el elemento o que los mensajes enviados desde el elemento se envían fuera del componente primario.

Tabla 8-13 Descripción de Elementos del Diagrama de Componentes.

Herramienta a utilizar para los Diagramas UML 2.0: StartUML

8.6.3 Modelado de Procesos de Negocio

BPMN 2.0

Business Process Model and Notation (BPMN) es una notación gráfica que describe la lógica de los pasos de un proceso de negocio. Esta notación ha sido especialmente diseñada para coordinar la secuencia de los procesos y los mensajes que fluyen entre los participantes de las diferentes actividades.

BPMN proporciona un lenguaje común para que las partes involucradas puedan comunicar los procesos de forma clara, completa y eficiente. De esta forma BPMN define la notación y semántica de un Diagrama de Procesos de Negocio (Business Process Diagram, BPD).

BPD es un diagrama diseñado para ser usado por los analistas, quienes diseñan, controlan y gestionan procesos. La notación BPMN será utilizada para describir los procesos de la situación actual, también será utilizado en la etapa de diseño detallado, el cual reemplazará al diagrama de actividades de UML.

Nomenclatura BPMN

Dentro de un Diagrama de Procesos de Negocio (BPD) se utiliza un conjunto de elementos gráficos, agrupados en categorías, que permite el desarrollo de diagramas simples que sean de fácil comprensión, los cuales se describen en la Tabla 8-14:



Elemento	Descripción	Notación
Evento	Es algo que sucede durante el curso del proceso de negocio. Afectan al flujo del proceso. Normalmente tienen una causa (disparador) o un impacto (resultado). Dependiendo de cuando afectan al flujo serán eventos iniciales, intermedios o finales.	
Actividad	Es un término genérico para el trabajo que realiza una compañía. Puede ser atómica (tarea) o compuesta (sub-proceso). Para indicar la no atomicidad se coloca un signo + en la esquina del símbolo de actividad.	
Gateway	Se utiliza para controlar la convergencia o divergencia de flujos.	
Secuencia	Se utiliza para mostrar el orden o secuencia en que las actividades se realizan en un proceso.	
Mensaje	Se utiliza para mostrar el flujo de mensajes entre dos participantes separados.	
Asociación	Se utiliza para mostrar entradas y salidas de actividades.	
Pool	Representa un participante en un proceso. Actúa como contenedor gráfico para particionar un conjunto de actividades.	
Lane	Es una sub-partición dentro de un pool y puede extenderse a todo lo largo o ancho del pool. Se utilizan para organizar y categorizar actividades.	
Objeto de Datos	Mecanismo para mostrar como los datos son requeridos y producidos por las actividades. Se conectan a las actividades por asociaciones.	




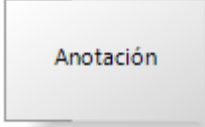
Elemento	Descripción	Notación
Grupo	Se utiliza para documentación o para propósitos de análisis, pero no afecta al Flujo de Secuencias.	
Anotación	Mecanismo para que quien está modelando provea información adicional para el lector del diagrama.	

Tabla 8-14 Descripción de Elementos Básicos de BPMN 2.0.

Herramientas a utilizar:

- Bizagi Process Modeler v2.4.

8.7 Metodología de Diseño de Base de Datos

Una base de datos es una colección de datos relacionados que representan algún aspecto del mundo real³³. El buen diseño de la base de datos es clave en el desarrollo del sistema informático, debido a que un sistema bien diseñado es fácil de desarrollar y mantener, mientras que los sistemas mal diseñados son costosos de mantener, difíciles de probar y poco fiables.

Para lograr un buen diseño de base de datos se utilizarán modelos de datos que permitan representar diferentes niveles de abstracción (conceptual, lógico y físico) de la base de datos.

Los modelos de datos son una colección de herramientas conceptuales que permiten describir y comprender datos, las relaciones entre estos, restricciones de consistencia, entre otros³⁴. En la Figura 8-14 se observa las fases del diseño de base de datos por medio de los modelos de datos y la relación de estos con el desarrollo del sistema.

³³ Elmasri, Ramez B., & Navathe, S. B. (2007). *Fundamentos de Sistemas de Base de Datos*. Madrid: Pearson Education S.A.

³⁴ Silberschatz, A., Korth, Henry F., & Sudarsha, S (2002). *Fundamentos de Base de Datos*. Aravaca Madrid: McGraw Hill Interamericana de España, S.A. U.

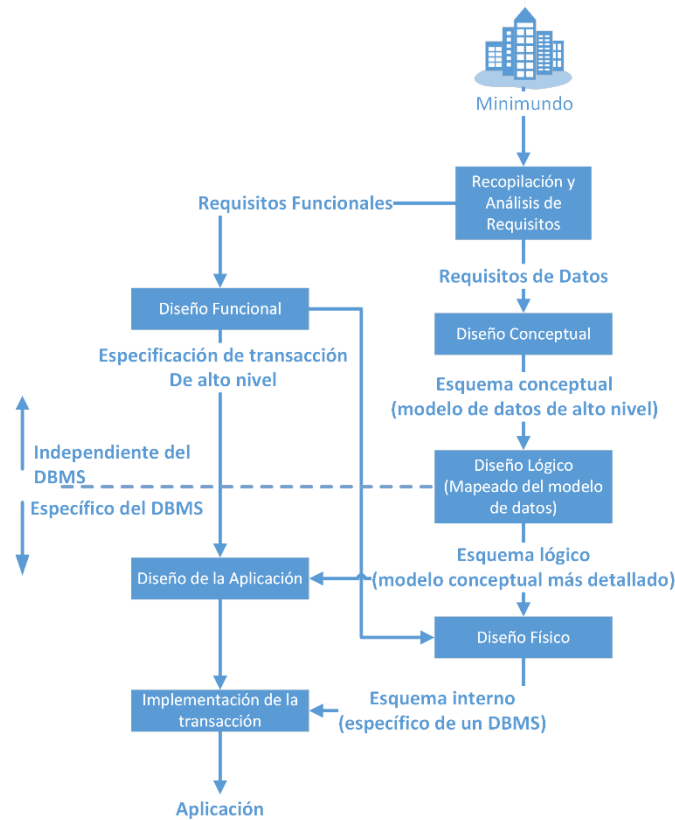


Figura 8-14. Ilustración de las principales fases del diseño de base de datos.

A continuación se describen los modelos de datos utilizados en el diseño de la base de datos:

Modelo de Datos Conceptual

Es un Modelo orientado a la descripción de estructuras de datos y restricciones de integridad, en este modelo se incluye descripciones detalladas de los tipos de entidades, atributos, relaciones y restricciones. Para poder representar el Modelo de Datos Conceptual se utilizará el Modelo Entidad-Relación Extendido (EER).

Modelo Entidad Relación Extendido (EER).

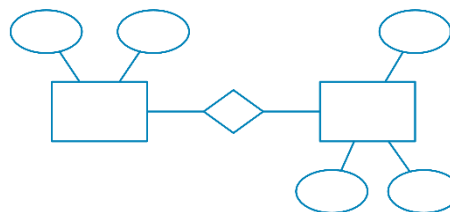


Figura 8-15 Bosquejo Diagrama Entidad Relación.



El EER se deriva del Modelo Entidad-Relación que es un modelo basado en una percepción del mundo real que consta de una colección de objetos básicos, como entidades, relaciones y atributos. El Modelo Entidad-Relación Extendido implementa mejoras con respecto al Modelo Entidad-Relación, como lo son los conceptos de subclase, superclase, especialización, generalización y unión, correspondientes al Enfoque Orientado a Objetos.








El uso de esta técnica de diagramación brindará una mejor abstracción de datos a un alto nivel, permitiendo con ello una mejor aproximación de la representación de datos en el modelo.

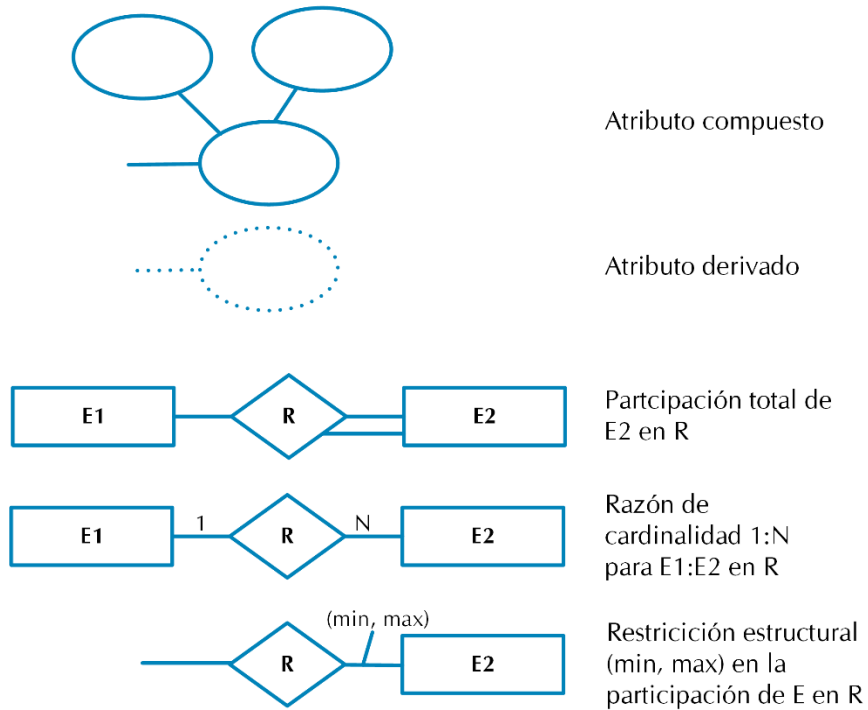
Herramienta a utilizar para el diseño del EER.

- Microsoft Visio Professional 2013.

Notación del Diagrama ER

A continuación se presenta un resumen de la notación para los diagramas ER según la notación de Peter Chen:

Símbolo	Significado.
	Entidad
	Entidad débil
	Relación
	Relación de identificación
	Atributo
	Atributo clave
	Atributo multivalor



Modelo de Datos Lógico

Este modelo consiste en transformar el diseño conceptual en el modelo interno de un Sistema Gestor de Base de Datos (DBMS), es decir, en este modelo se realiza la proyección de todos los objetos en las construcciones específicas utilizadas por el DBMS, lo que incluye el diseño de tablas, índices, vistas, transacciones, etc.³⁵

Para la representación del Modelo de Datos Lógico se utilizará el **Modelo Relacional** que representa la base de datos como una colección de relaciones, asemejándose este conjunto de relaciones a una tabla de valores.

Modelo Relacional

Este modelo está basado en la lógica de predicados y en la teoría de conjuntos, es el modelo más utilizado para modelar problemas reales y administrar datos dinámicamente.

Este modelo se compone de los siguientes elementos:

- Dominio: Conjunto de valores permitidos para un atributo.
- Atributo: Propiedad de una relación.
- Relación: Subconjuntos del producto cartesiano de la lista de dominios.
- Tupla: Fila que contiene valores de los atributos de una relación.

Para una mejor comprensión de este modelo, en la **Figura 8-16** se muestra una representación de una relación como una tabla, cuyos elementos se describen en la **Tabla 8-15**.

³⁵ Rob, P., & Coronel, C. (2004). *Sistema de Base de Datos Diseño, Implementación y Administración* (5 ed.). CI, México: International Thomson Editores S.A. de C.V.



Figura 8-16 Representación de una relación como tabla en el Modelo Relacional.

Número	Elemento	Descripción
1	Nombre	Elemento en el que se especifica el nombre de la relación.
2	Atributos	Elemento en el que se especifica el nombre del atributo, una relación está compuesta por un conjunto de atributos, que representan las propiedades de la tabla.
3	Tuplas	Conjunto de filas que contienen los valores de los atributos para cada relación.

Tabla 8-15 Descripción de los elementos de una relación en el Modelo Relacional.

Herramienta a utilizar para el diseño del Modelo Relacional.

- Sysbase Power Designer 16.1.

Modelo de Datos Físico

Son estructuras de datos a bajo nivel implementadas para un determinado DBMS, ayuda a analizar las tablas, índices, triggers, vistas y procedimientos almacenados, así como las referencias entre ellos. El resultado es un esquema de base de datos, del cual, con una herramienta adecuada, se puede obtener el script de base de datos del DBMS para el que fue diseñado.

Herramientas a utilizar:

- Power Designer 16.1
- DBMS: PostgreSQL 9.1



8.8 Técnicas de Programación

En relación con el enfoque a utilizar en el desarrollo del software y las técnicas de análisis y diseño seleccionadas, se utiliza la Programación Orientada a Objetos para la etapa de codificación.

Programación Orientada a Objetos

Es un paradigma de programación que usa los objetos en sus interacciones. Está basado en varias características, incluyendo herencia, cohesión, abstracción, polimorfismo, acoplamiento y encapsulamiento³⁶. Este enfoque de Programación va de la mano con el Enfoque Orientado a Objetos definido en la sección 8.6.1 de este documento.

Los principales conceptos de esta técnica de programación son:

- **Clase:** Define las propiedades y comportamientos generales de un tipo de objeto concreto. La instanciación es una copia de estas propiedades y comportamientos, y la creación de un objeto a partir de ellas.
- **Herencia:** Es la relación entre una clase general y otra clase más específica, implica que una subclase obtiene todo el comportamiento (métodos) y eventualmente los atributos (variables) de su superclase.
- **Objeto:** Instancia de una clase. Entidad provista de un conjunto de propiedades o atributos (datos) y de comportamiento o funcionalidad (métodos), los mismos que consecuentemente reaccionan a eventos. Se corresponden con los objetos reales del mundo que nos rodea, o con objetos internos del sistema.
- **Método:** Algoritmo o función asociada a un objeto (o a una clase de objetos), cuya ejecución se desencadena tras la recepción de un "mensaje". Desde el punto de vista del comportamiento, es lo que el objeto puede hacer.
- **Evento:** es un suceso en el sistema (tal como una interacción del usuario con la máquina, o un mensaje enviado por un objeto).
- **Atributos:** Propiedades o características que tiene la clase.
- **Mensaje:** una comunicación dirigida a un objeto, que le ordena que ejecute uno de sus métodos con ciertos parámetros asociados al evento que lo generó.

Lenguaje de Programación

El lenguaje utilizado para la etapa de codificación es el lenguaje de programación PHP 5.3. PHP es un lenguaje de programación de uso general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico³⁷, el cual desde la versión 5.0 tiene soporte para la Programación Orientada a Objetos.

³⁶ Fundación Wikipedia, Inc. (3 de abril de 2013). *Programación Orientada a Objetos*. Obtenido de Wikipedia: http://es.wikipedia.org/wiki/Programaci%C3%B3n_orientada_a_objetos

³⁷ Fundación Wikipedia, Inc. (11 de abril de 2013). *PHP*. Obtenido de Wikipedia: <http://es.wikipedia.org/wiki/PHP>.



Framework

El framework a utilizar es Symfony 4.2.x, el cual está desarrollado completamente en PHP 5.3. Symfony es un completo framework diseñado para optimizar el desarrollo de las aplicaciones web basado en el patrón Modelo Vista Controlador³⁸.

Descripción del Modelo MVC en Symfony 2

Modelo Vista Controlador (MVC) es un patrón o modelo de abstracción de desarrollo de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de negocio en tres componentes distintos³⁹.

- **Modelo:** representa el dominio de los datos. Define la lógica de negocio (la base de datos pertenece a esta capa).
- **Vista:** es la interfaz de usuario.
- **Controlador:** es el responsable de ejecutar las peticiones del usuario. Es el que invoca peticiones al modelo y llama a las vistas.

A pesar de que se pueden hacer cosas muy complejas con Symfony2, el funcionamiento interno siempre es el mismo:

1. El Controlador manda y ordena
2. El Modelo busca la información que se le pide.
3. La Vista crea páginas web con plantillas y datos.

La Figura 8-17 muestra el modelo MVC en Symfony 2, en el cual se ilustra funcionamiento interno de este framework. De esta forma es como se aplicará la Programación Orientada a Objetos y el Mapeo Objeto Relacional (que se describe en el apartado siguiente).

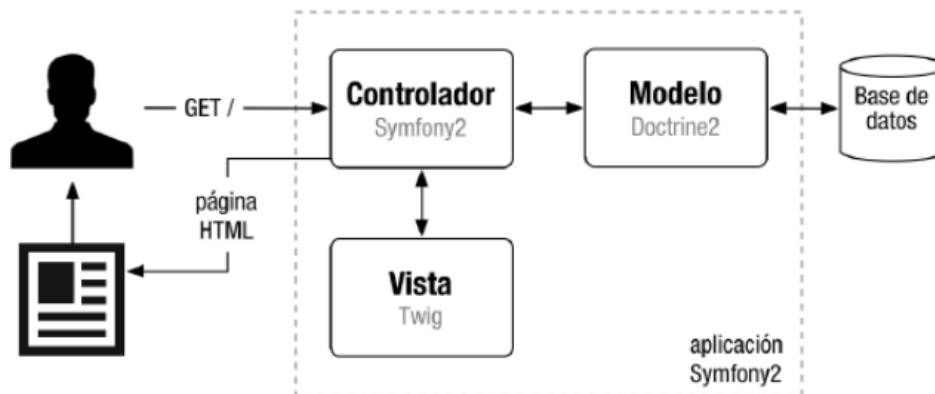


Figura 8-17 Modelo MVC en Symfony 2

³⁸ Fundación Wikipedia, Inc. (12 de marzo de 2013). *Symfony*. Obtenido de Wikipedia: <http://es.wikipedia.org/wiki/Symfony>.

³⁹ Ministerio de Salud. (4 de marzo de 2013). *Desarrollo web Symfony 2 parte 2*. Obtenido de wiki de la DTIC-MINSAL: http://wiki.salud.gob.sv/wiki/Desarrollo_web_Symfony2_parte_2.



Para más información sobre el funcionamiento interno del framework Symfony vea la Sección 11.3.4 de este documento.

Herramientas a utilizar para programación de código:

- Sublime Text.

Mapeo Objeto Relacional

El mapeo objeto-relacional, también conocido por su nombre en inglés, Object-Relational mapping, o sus siglas (ORM) es una técnica de programación para convertir datos entre el sistema de tipos utilizado en un lenguaje de programación orientado a objetos y el utilizado en una base de datos relacional, utilizando un motor de persistencia⁴⁰.

Para realizar la conversión de datos se utilizará a **Doctrine 2.3.2**, el cual es un mapeador de objetos-relacional (ORM) escrito en PHP que proporciona una capa de persistencia para objetos PHP. Es una capa de abstracción que se sitúa justo encima de un SGBD.

Herramientas a utilizar para el ORM y su configuración:

- Sublime Text.

⁴⁰ Fundación Wikipedia, Inc. (12 de marzo de 2013). *Symfony*. Obtenido de Wikipedia: http://es.wikipedia.org/wiki/Mapeo_objeto-relacional



8.9 Metodología de Pruebas del Software.

La Metodología de Pruebas del Software comprende un conjunto de técnicas que permitirán la validación y verificación (V&V) del software, aunque ambas no son lo mismo, se suelen confundir con frecuencia. La verificación es el proceso de determinar si los productos de una etapa específica del ciclo de vida de desarrollo de software cumple con los requisitos establecidos previamente para esa etapa, es decir, si se ha construido de manera correcta, mientras que la validación es el proceso de evaluación del software que garantiza el cumplimiento de los requisitos establecidos del software⁴¹, lo que significa que se ha construido el software correcto.

La finalidad de establecer una metodología de pruebas es asegurar la calidad del software. Con las pruebas se intenta demostrar que el software hace lo que tiene que hacer, así como descubrir defectos antes de la puesta en marcha. El proceso de pruebas tiene dos metas distintas:

- Demostrar al desarrollador y al cliente que el software cumple con los requerimientos.
- Encontrar situaciones en donde el comportamiento del software sea incorrecto.

La primera meta conduce a las pruebas de validación, la que demuestra que el sistema se desempeña de manera correcta, mediante un conjunto de casos de prueba, mientras que la segunda está orientada a encontrar defectos en el software.

8.9.1 El Plan de Pruebas y el Ciclo de Vida de Desarrollo de Software.

Para una mejor comprensión de la relación existente entre un plan de pruebas y las etapas del ciclo de vida de desarrollo de software, se puede observar en la **Figura 8-18** que cada etapa del ciclo de vida está relacionada a un plan de pruebas que contiene los lineamientos a seguir en el desarrollo de la prueba.

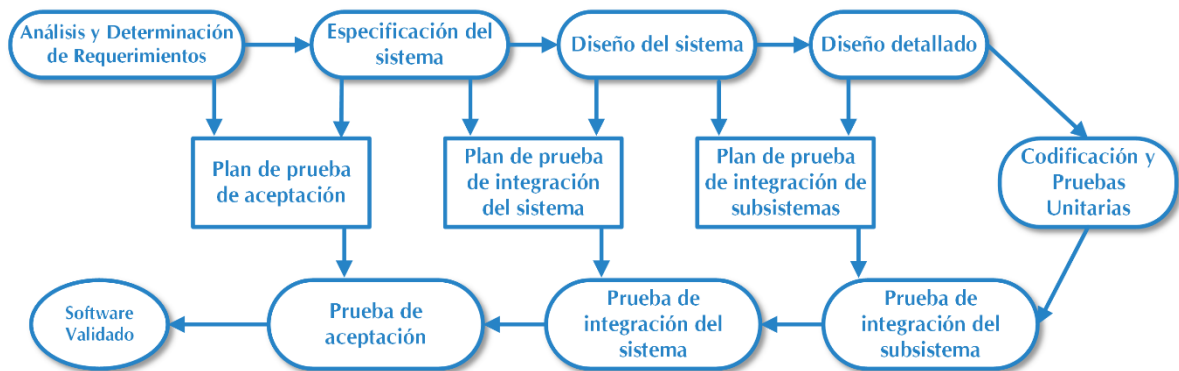


Figura 8-18 Integración del Plan de Pruebas con el Ciclo de Vida de Desarrollo de Software.

⁴¹ IEEE 729-1983. (1982). *Glossary of Software Engineering Terminology*.



8.9.1.1 Etapas del Proceso de Pruebas

Para la correcta validación del software y la aplicación de los planes de pruebas mostrados en la **Figura 8-18**, se han dividido el proceso de pruebas en las siguientes etapas:

- Pruebas de Desarrollo.
- Pruebas del Sistema.
- Pruebas de Aceptación.

Pruebas de Desarrollo

Etapa en la que los desarrolladores ponen a prueba los componentes que constituyen el sistema a medida que se va desarrollando, esta etapa se desarrolla en conjunto con la etapa de codificación (ver **Figura 8-7** del apartado 8.4.1), esta etapa involucra el siguiente tipo de prueba:

- **Pruebas Unitarias:** Es el proceso de poner a prueba todas las características del componente del software como métodos o clases de objeto. Con este tipo de prueba se pretende verificar todas las operaciones asociadas al objeto, establecer y verificar el valor de todos los atributos relacionados con el objeto y poner el objeto en todos los estados posibles.

Pruebas del Sistema.

Etapa en que los componentes del sistema se integran para crear un sistema completo. Este proceso tiene la finalidad de descubrir errores que resulten de interacciones no anticipadas entre componentes, y problemas de interfaz de componentes, así como de demostrar que el sistema cubre con los requerimientos funcionales y no funcionales. Esta etapa involucra las pruebas de integración de subsistemas y pruebas de integración del sistema.

- **Pruebas de integración de subsistemas:** Proceso en el que se pone a prueba los componentes del sistema y las interfaces con otros componentes, con este tipo de pruebas se pretende verificar la funcionalidad del componente y encontrar errores al establecer una comunicación con otros componentes.
- **Pruebas de integración del sistema:** proceso en el que se pone a prueba la funcionalidad del sistema como un todo, el objetivo de esta prueba es verificar que la integración de nuevos componente no generen errores en el sistema, este proceso involucra las pruebas de regresión.

Pruebas de Aceptación

Etapa en donde los usuarios o clientes proporcionan datos de entrada y asesoría sobre las pruebas del sistema. Este tipo de prueba es esencial aún y cuando se haya realizado pruebas al sistema completo, debido a que la influencia del entorno de trabajo del usuario tiene un gran efecto sobre la fiabilidad, rendimiento, uso y robustez del sistema, ya que es casi imposible replicar el entorno de trabajo del usuario. Esta etapa involucra las pruebas alfa, beta y de aceptación.



- **Pruebas alfa:** Proceso en donde los usuarios del software trabajan con el equipo de diseño para probar el software a medida que se desarrolla, esto es debido a que los usuarios pueden identificar problemas y conflictos que no son fácilmente aparentes para el equipo, permitiendo así un diseño de pruebas más realista.
- **Pruebas beta:** Procesos en donde una versión del software se pone a disposición de un grupo selecto de usuarios, para permitirles experimentar y descubrir problemas, esto debido a que es imposible que los desarrolladores conozcan y repliquen todos los entornos donde se usará el software, permitiendo así descubrir problemas de interacción entre el software y las características del entorno en donde se emplea.
- **Pruebas de aceptación:** Proceso en el cual los usuarios prueban un sistema para decidir si está o no listo para ser aceptado.

8.9.1.2 Modelo del Proceso de Pruebas

El Modelo de Proceso de Pruebas del software define un conjunto de pasos a seguir para la realización de las pruebas en cada una de las Etapas del Proceso de Prueba, la **Figura 8-19** describe los pasos a seguir por las etapas de Pruebas de Desarrollo y Pruebas del Sistema, para la Etapa de Pruebas de Aceptación los pasos a seguir se describen en la **Figura 8-20**.

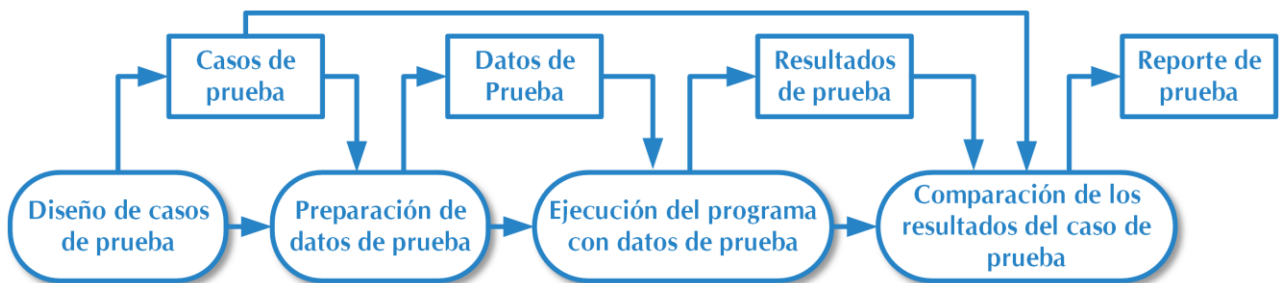


Figura 8-19 Modelo de Proceso de Pruebas de Software.

La **Figura 8-19** muestra que para la realización del proceso de prueba es necesario definir primeramente un diseño de caso de prueba, el cuál luego de la preparación de los datos de prueba y de la ejecución de la prueba, se realiza una comparación de los resultados de la prueba contra los resultados del caso de prueba, el cuál indicará si existe una variación de los resultados esperados contra los resultados obtenidos, identificando así posibles errores dentro del software.

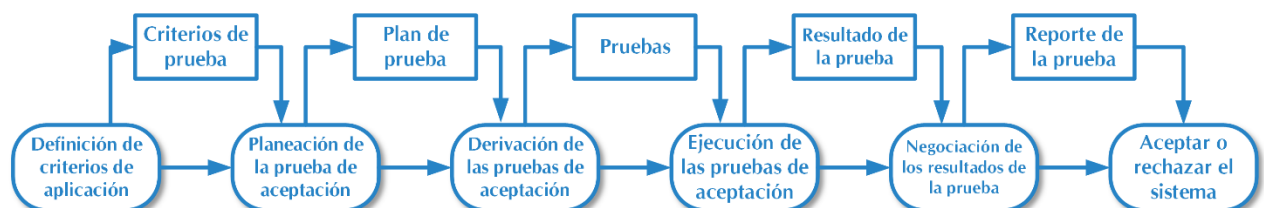


Figura 8-20 Modelo de Proceso de Pruebas de Aceptación.



El Modelo de Proceso de Prueba presentado en la **Figura 8-20** es diferente del presentado en la **Figura 8-19**, esto es debido a que las pruebas de aceptación involucran ciertos pasos extras como lo son la definición de los criterios de aceptación, la negociación de los resultados de la prueba y la aceptación o negación del sistema, esto con el objeto de tener un modelo de proceso que se adecúe más al proceso de validación del software por parte del usuario.

8.9.2 Diseño y Ejecución de Casos de Prueba.

Los Casos de Prueba son especificaciones de una característica o componente de software que se pone a prueba, las entradas a la prueba y las salidas esperadas⁴². Los Casos de Prueba, también especifican un conjunto de pasos que deben ser realizados para obtener las salidas esperadas. Luego que los casos de prueba son realizados o ejecutados, se compara la salida real obtenida con la salida esperada, determinando si la prueba tuvo éxito (en caso de que la salida obtenida es igual a la salida esperada) o un fallo (en caso de que la salida obtenida difiera de la salida esperada).

8.9.2.1 Lineamientos de Diseño y Ejecución:

Los casos de prueba tienen que demostrar dos cosas:

1. Cuando se utiliza el componente o característica que se somete a prueba, hace lo que se espera que debe hacer.
2. Si hay defectos en el componente o característica, deben ser revelados mediante el caso de prueba.

Esto significa, que para probar un componente, característica o funcionalidad del software, se deben diseñar al menos dos Casos de Prueba para dicha característica. Es decir, un Caso de Prueba que refleje una operación normal, con datos que se encuentren dentro del dominio permitido y mostrar que el componente funciona; y otro Caso de Prueba con entradas anormales (datos fuera del dominio permitido) para comprobar que se procesan de manera adecuada (generar mensajes de error, cancelar acciones, etc.) sin que se colapse el componente.

Los Casos de Prueba se deben organizar en Suites. Una Suite de Pruebas agrupa Casos de Prueba que se relacionan o que están asociados a determinada funcionalidad del Sistema. Por Ejemplo:

- Suites de Pruebas de Inicio de Sesión de Usuario: Agrupa Casos de Pruebas asociados a las pruebas de inicio de sesión de usuario.
- Suite de Pruebas de Gestión de Empleados: Agrupa Casos de Prueba asociados al registro de empleados, actualización de empleados, eliminación de empleados, etc.

⁴² Sommerville, I. *Ingeniería de Software* (9a. ed.). Naucalpan de Juárez, México: Pearson Education.



Plantilla para el Diseño de Casos de Prueba.

Para el diseño de casos de prueba se utiliza la siguiente plantilla (Cuadro 8-4), especificando lo que contendrá cada campo:

Nombre del Caso de Prueba:	Entendible por personas, para la fácil comprensión del propósito del caso de prueba.
Código:	CP-00-00
Versión:	1.0
Suite	Nombre de la Suite en la que se agrupa el Caso de Prueba.
Descripción General	
Propósito:	Contiene una breve descripción del propósito de la prueba, y la funcionalidad que se prueba.
Requerimiento Funcional a Probar:	Nombre del Caso de Uso que se Prueba (Código del Caso de Uso)
Dependencias:	Otros subsistemas que están involucrados y en qué grado.
Ambiente de Pruebas	
Hardware	Contiene información acerca de la configuración del hardware.
Software	Contiene información acerca de la configuración del Software. Sistema Operativo: Navegador Web: Herramienta de Pruebas: Otros:
Inicialización o Pre-Condiciones:	Describe acciones, que deben ser ejecutadas antes de que los casos de prueba se hayan inicializado. Por ejemplo, si se debe abrir algún archivo.
Finalización o Post-Condiciones:	Describe acciones, que deben ser ejecutadas después de realizado el caso de prueba. Por ejemplo si el caso de prueba afecta la base de datos, el analista debe restaurarla antes de que otro caso de prueba sea ejecutado.
Desarrollo de la Prueba	
Acciones:	Describe la secuencia de acciones a realizar para completar la prueba.
Datos de entrada:	Definir las entradas requeridas para ejecutar el caso de prueba.
Resultados	
Resultados/Datos Esperados:	Contiene una descripción de lo que el analista debería ver tras haber completado todos los pasos de la prueba, especificando el/los criterios para determinar si ha sido una prueba exitosa (correcta) o no (fallida).
Severidad:	Indica el impacto del defecto en el sistema en caso de ser una prueba fallida: Grave, Mayor, Normal, Menor.

Cuadro 8-4. Plantillas para Diseño de Casos de Prueba.



Plantilla para la Ejecución de Casos de Prueba.

Al seguir los pasos especificados en el diseño de cada Caso de Prueba, se debe documentar, de forma paralela a su ejecución, los resultados del mismo. Se utilizará para ellos la siguiente plantilla (Cuadro 8-5):

Caso de Prueba:	Nombre del Caso de Prueba definido en su Diseño.
Código:	CP-00-00 (Código Definido en su Diseño).
Suite	Nombre de la Suite en la que se agrupa el Caso de Prueba.
Correlativo:	Numero correlativo de la ejecución del Caso de Prueba.
Desarrollo de la Prueba	
Comandos de Selenium IDE:	Describe la secuencia de comandos a realizar para completar la prueba en el Software de pruebas Selenium IDE.
Datos de entrada:	Definir las entradas para ejecutar el caso de prueba.
Resultados	
Resultados/Datos Esperados:	Contiene una descripción de lo que el analista debería ver tras haber completado todos los pasos de la prueba, especificando el/los criterios para determinar si ha sido una prueba exitosa (correcta) o no (fallida).
Resultados/Datos Obtenidos:	Contiene una breve descripción de lo que el analista obtiene realmente después de que los pasos del caso de prueba se hayan ejecutado.
Resultado:	Éxito <input type="checkbox"/> Fallo <input type="checkbox"/>
Código de Seguimiento:	Si un caso de prueba falla, la referencia al defecto implicado se debe enumerar en esta columna. (ID del Bug reportado en MantisBT.)
Evidencia:	Referencia al/los Screenshot o Capturas de pantalla del resultado obtenido, siendo evidencia para hacer constar que se ha completado con éxito o no.
Cumplió Post-Condición:	¿Se realizó la Post-Condición que se especificó en el diseño del Caso de Prueba?

Cuadro 8-5. Plantilla para Ejecución de Casos de Prueba.

8.9.2.2 Software para Implementación y Ejecución de Casos de Prueba.

Luego del Diseño de los Casos de Prueba, estos deber ser ejecutados por el desarrollador o analista. Es imposible la generación automática de los Casos de Prueba, pues deben estar diseñados por personas que entiendan lo que se supone que debe hacer el sistema, para especificar los resultados esperados o el comportamiento del mismo. Sin embargo, aunque no pueda automatizarse su diseño, se puede automatizar la ejecución de estos⁴³. Los Casos de Prueba, serán implementados para su ejecución mediante el software Selenium IDE.

Selenium IDE, es un ambiente de desarrollo integrado que permite la creación, edición, depuración y ejecución de Casos de Prueba. Este software esta implementado como una extensión del navegador Mozilla Firefox y su equivalente Iceweasel.

Los pasos especificados en el diseño del Caso de Uso (secuencia de acciones a realizar para completar la prueba), pueden implementarse en Selenium IDE, agruparse en Suites de Prueba y posteriormente ejecutarse. Lógicamente, esto facilitará la implementación de los Casos de Prueba, ya

⁴³ Sommerville, I. *Ingeniería de Software* (9a. ed.). Naucalpan de Juárez, México: Pearson Education.



que se realiza en un entorno gráfico que agiliza la productividad; la ejecución de los Casos de Prueba, y así mismo, el cambio en los datos de entrada, las verificaciones y validaciones.

En esencia, el software imita la forma en que un usuario interactúa con el Sistema, como el llenado y envío de formularios, clic en enlaces, clic en botones, etc. Esto se logra mediante el uso de comandos que el software proporciona y que el desarrollador/analista ordena lógicamente, según las especificaciones de diseño del Caso de Prueba, añadiendo datos de entrada, verificaciones y validaciones (conocidas en inglés como Asserts), que ayudan a determinar si el Caso de Prueba fue exitoso o fallido.

Ejemplo de Implementación de un Caso de Prueba en Selenium IDE a partir de las acciones descritas en el diseño (Tabla 8-16):

Acciones (Especificadas en la plantilla de diseño)		Implementación en Selenium IDE (comandos)		
#	Acción	Comand	Target	Valor
1	Ingresar a la página de inicio de sesión.	open	http://localhost/web/app.php/admin/login	
2	Ingresar un Nombre de Usuario válido.	type	id = Username	admin
3	Ingresar el Password válido para el Usuario.	type	id = Password	pass
4	Iniciar Sesión.	clickAndWait	id = submit	
5	Verificar el despliegue de la página de inicio.	assertLocation	http://localhost/web/app.php/admin/index	

Tabla 8-16. Implementación de un Caso de Prueba en Selenium IDE.

8.9.3 Sistema de Seguimiento de Errores (Bug Tracking System).

Un Sistema de Seguimiento de Errores (en adelante BTS, por sus siglas en ingles), es una aplicación diseñada para ayudar a asegurar la calidad del software, por medio del seguimiento de los defectos o errores encontrados en la ejecución de las pruebas de software. Estos defectos o errores, también se denominan ‘Bugs’.

Al tener un Caso de Prueba, cuyo resultado sea una Falla, se ha encontrado un defecto. La persona que realizó el Caso de Prueba está en la obligación de registrarlo en el BTS. Luego de realizar el registro, se debe especificar en la Plantilla de Ejecución del Caso de Prueba en cuestión, la referencia al código correlativo (generado y asignado por el BTS) del defecto. Posteriormente, el defecto será asignado a uno de los miembros del equipo de desarrollo para verificarlo, realizar las modificaciones correspondientes y solucionarlo. Estas acciones, también deben ser registradas en el BTS, las cuales son soportadas por el mismo sin ningún problema.

El BTS a utilizar es **Mantis Bug Tracker**, está desarrollado en PHP, es decir, para un entorno web, con soporte para múltiples bases de datos como MySQL, PostgreSQL y DB2.



Los datos a registrar con respecto al defecto son los mostrados en el Cuadro 8-6:

Datos.	Descripción
Categoría	Categoría a la que el defecto pertenece. (Ej. Interfaz de Usuario, Base de Datos, Mensajes, etc.).
Reproducibilidad	La forma en que se reproduce el defecto (Siempre, a veces, aleatorio, etc.).
Severidad	Alta, Critica, Baja, Trivial, etc.
Resumen	Breve y clara descripción del defecto.
Descripción	Descripción detallada del defecto encontrado.
Pasos de ejecución	Pasos para forzar que el defecto sea reproducido u observado.
Otras información	Otra información de ayuda.
Archivo	Archivo que posee información del defecto (imagen, evidencia, código fuente, etc.).

Cuadro 8-6. Datos a Registrar de los Defectos Encontrados.

8.9.4 Técnicas de Pruebas de Software.

Una vez establecido las etapas del proceso de pruebas, sus tipos y el modelo de procesos, es necesario establecer los tipos de técnicas de pruebas a utilizar para la validación del software, estas técnicas permitirán examinar desde diferentes enfoques el funcionamiento del software, con lo cual se asegurará una mejor calidad del software, a continuación se presentan las técnicas a utilizar en el proceso de pruebas.

- Caja Negra.
- Caja Blanca

Caja Negra



Figura 8-21 Metodología de Caja Negra

En la metodología de la caja negra se asume que se conoce el funcionamiento interno de los elementos que se desean probar, esta metodología se interesa en las salidas que genera a través de las entradas que son proporcionadas para la prueba (como se esquematiza en la Figura 8-21). La metodología de la caja negra involucra dos tipos de pruebas:

- Pruebas del Sistema
- Pruebas de Aceptación

El objetivo de la utilización de la caja negra es la evaluación de los resultados de las salidas que se generan a partir de un conjunto de datos de entrada, que se seleccionan para probar que el



software cumple con los requerimientos y funciona de manera correcta, para lo cual se realizan casos de prueba los que se clasifican en las siguientes categorías (ver Figura 8-22):

- Valores de Entrada Esperados: Esto incluye los valores que se esperan que la prueba reciba la mayoría de las veces.
- Límite de Valores: Caso en el que se prueba el límite máximo de valores que acepta el elemento en prueba, es decir, si un elemento tiene como límites de 1 a 999, se debe de realizar un caso en el que se prueben todos esos valores.
- Valores erróneos: Tomando de base el caso de límite de valores, realizar un caso de prueba en el que los valores seleccionados se encuentren fuera del dominio del elemento, con esto se pretende probar además los mensajes de ayuda, error, entre otros, o se pretende probar el comportamiento del software ante dichos casos.

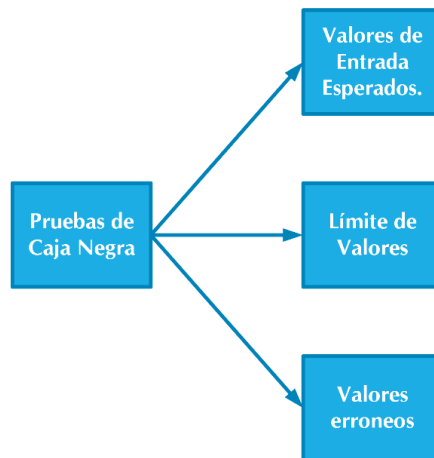


Figura 8-22 Casos de Prueba Metodología de la Caja Negra.

Caja Blanca

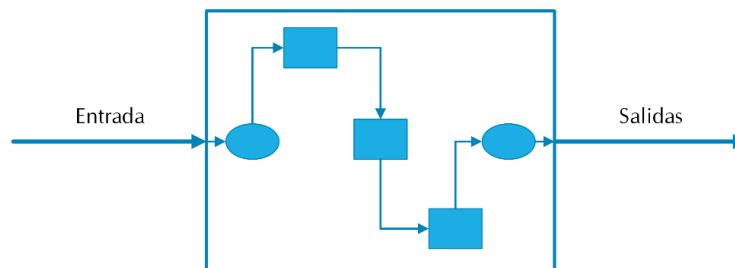


Figura 8-23 Metodología de la Caja Blanca.

El objetivo de esta metodología es centrarse en los detalles procedimentales del software, por lo que su diseño está ligado al código fuente, se escogen distintos valores de entrada para examinar cada uno de los posibles flujos de ejecución del programa y verificar que los valores de salida son los esperados (como se esquematiza en la Figura 8-22). Esta técnica involucra los siguientes tipos de pruebas:



- Pruebas Unitarias
- Pruebas de Integración
- Pruebas de Regresión

El desarrollo de la metodología de la Caja Blanca involucra los siguientes pasos (ver Figura 8-24):

1. Prueba de sentencias: Se provee un conjunto de valores de pruebas para cada método diseñado que asegure que cada sentencia en el método se ejecuta al menos una vez, esto incluye todas las sentencias del método.
2. Pruebas de decisión: Se provee un caso de prueba que asegure que cada rama de una decisión es ejecutada al menos una vez.
3. Pruebas de Condición: requiere diseñar un caso de prueba que asegure que cada condición en la decisión toma todos los valores posibles de salida al menos una vez, y cada punto de entrada del programa o subrutina es invocado al menos una vez.

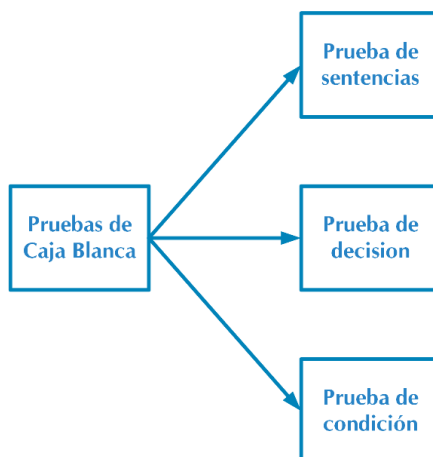


Figura 8-24 Pasos de la Metodología de la Caja Blanca.



8.10 Estándares de Documentación.

Para tener un marco de referencia y control sobre la edición de cada uno de los documentos correspondientes al proyecto es necesario establecer estándares, los cuales brindaran los detalles necesarios para que cada uno de los miembros del equipo pueda colaborar en la edición y mantener el mismo formato en todo el documento. Dicha documentación poseerá los siguientes estándares:

Elemento	Subelemento	Tipo	Valor
Página	Tamaño	Ancho	21.59 cm
		Alto	27.94 cm
	Margen	Inferior, Superior	2.0 cm
		Izquierdo	3.0 cm
Derecho		1.8 cm	
Capítulo	Titulo 1	Tipo de Letra	Century Gothic
		Tamaño	36
		Color	RGB(23,54,93)
	Titulo 2	Tipo de Letra	CG Omega
		Tamaño	26
		Color	RGB(0,0,0)
Texto	Tipo de Letra	CG Omega	
	Tamaño	10	
	Color	RGB(0,0,0)	
Documento	Titulo 1	Tipo de letra	Century Gothic
		Tamaño	18
		Color	RGB(0,0,0)
		Interlineado	Múltiple 1.15
	Titulo 2	Espaciado	Superior - 24, Inferior - 0
		Tipo de letra	CG Omega
		Tamaño	15
		Color	RGB(0,0,0)
	Titulo 3	Interlineado	Múltiple 1.15
		Espaciado	Superior - 2, Inferior - 0
		Tipo de letra	CG Omega
		Tamaño	13
	Titulo 4	Color	RGB(0,0,0)
		Interlineado	Múltiple 1.15
		Espaciado	Superior - 10, Inferior - 0
		Tipo de letra	CG Omega
	Texto	Tamaño	12
		Color	RGB(0,0,0)
		Interlineado	Múltiple 1.15
		Espaciado	Superior - 2, Inferior - 0
Titulo 3	Tipo de letra	CG Omega	
	Tamaño	11	
	Color	RGB(0,0,0)	
	Interlineado	Múltiple 1.15	
Titulo 4	Espaciado	Superior - 0, Inferior - 10	
	Tipo de letra	CG Omega	
	Tamaño	11	
	Color	RGB(0,0,0)	
Texto	Interlineado	Múltiple 1.15	
	Espaciado	Superior - 0, Inferior - 10	
	Tipo de letra	CG Omega	
	Tamaño	11	

Tabla 8-17. Estándares de documentación.



Estándares de tablas.

Los estándares para el formato de las tablas ayudaran a tener una mejor visualización de los datos. Así también, en mantener un mismo formato para toda la documentación del proyecto, tal como se muestra en la Tabla 8-18.

Elemento	Detalle
Encabezado	Negrita
Tamaño de letra	11 pto
Tipo de Letra	CG Omega
Relleno encabezado	RGB(219,229,241)
Color de Línea	RGB(149,179,215)
Grosor de Línea	1 pto

Tabla 8-18. Estándares de tablas.

Estándar para Referencias Bibliográficas

El estándar a utilizar para las referencias bibliográficas es el estilo **APA**. Es el estándar adoptado por la Asociación Americana de Psicología (American Psychological Association, APA) que los autores utilizan al momento de presentar sus documentos o textos para las revistas publicadas por la entidad.

El Manual de publicaciones de la APA contiene directrices para todos los aspectos relacionados con la redacción de referencias bibliográficas, entre otras.

Es utilizado frecuentemente para las citas a textos en un artículo, libro, Internet y otras formas de documentos.

La versión que se utilizará para el desarrollo de la documentación del proyecto será la versión 6.0.

8.11 Control de Versiones.

8.11.1 Control de Versiones de Software

El control de versiones para el desarrollo de software, es un sistema que registra los cambios realizados sobre un archivo o conjunto de archivos (directorios y archivos fuente) a lo largo del tiempo, de modo que se pueda regresar a un estado anterior.

Con la aplicación de este sistema de control, se seguirá la historia de los directorios y archivos fuentes del software a desarrollar, manteniendo un registro de todo el trabajo y los cambios en los archivos fuente, teniendo un control sobre las líneas afectadas y fomentar las buenas prácticas en las etapas de desarrollo. Así también, fomentara la comunicación, la reutilización de código y minimizara los riesgos en pérdidas de información.

El control de versiones ayudará a mantener un esquema de trabajo entre los desarrolladores, con el fin de agilizar el proceso de integración. La Figura 8-25 muestra la forma de trabajo para el



control de versiones durante el desarrollo del software: se puede observar cuatro desarrolladores, un repositorio central y las líneas que muestran la comunicación entre los desarrolladores.

En la Figura 8-25 se puede observar además que cada desarrollador posee un repositorio local, independiente del repositorio en línea, esta ventaja permite trabajar de manera independiente y sin necesidad de conexión a internet, que es una gran ventaja en aquellos momentos en el que el acceso a internet es muy difícil de obtener.

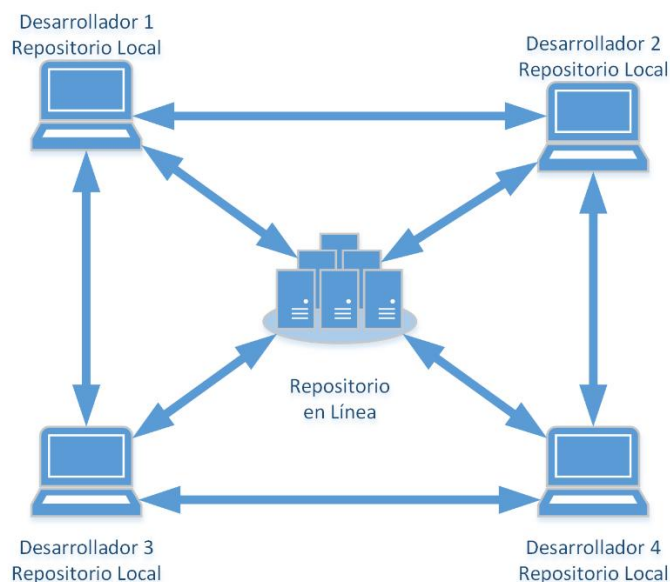


Figura 8-25 Flujo del Control de Versión para el Desarrollo del Software.

GIT

La herramienta a utilizar para lograr el objetivo será Git, que es un Sistema de Control de Versiones distribuido Open Source, diseñado para el manejo de pequeños o grandes proyectos de manera rápida y eficiente. GIT permite además el uso de ramas (branches), lo que permitirá a los desarrolladores crear diferentes ramas para trabajar de forma independiente, sin afectar la rama principal de desarrollo.

8.11.1.1 Metodología de Desarrollo con GIT.

Creación de Ramas.

Durante el desarrollo del proyecto es necesario la creación de las siguientes ramas:

- **Master:** Rama principal del software del proyecto, se crea automáticamente se inicia un repositorio. Esta rama contendrá las versiones finales (funcionales y libres de errores), del software del proyecto.
- **Desarrollador:** Se creará una rama por desarrollador, cada rama tendrá el nombre del desarrollador.



Cabe aclarar que cada repositorio tiene sus propias ramas, es decir se tendrán 2 ramas locales en las cuales se estarán trabajando (master y la del desarrollador) ubicada en el equipo del desarrollador y se tendrán 2 ramas remotas (master y la del desarrollador) que se encontrará en la nube (bitbucket.org), que servirán para sincronizar los cambios realizados y que siempre se tenga una versión final actualizada del software. La creación y administración de las ramas por medio de GIT permitirá el desarrollo separado de las funcionalidades del software, permitiendo con ello que cada desarrollador pueda trabajar de manera independiente una parte del software sin interferir en el desarrollo de otros, la **Figura 8-26** muestra un ejemplo del funcionamiento del control de versiones con la implementación de ramas en GIT.

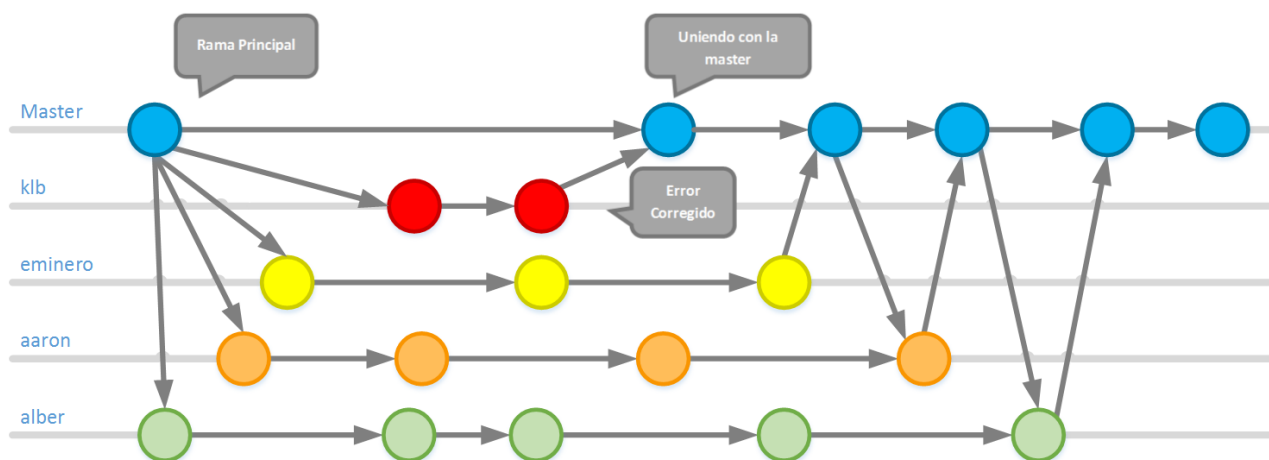


Figura 8-26 Ramas de desarrollo en GIT.

Flujo de Trabajo en Git.

Git es una herramienta potente que brinda grandes funcionalidades en el control de versiones, por lo cual para un uso óptimo de la herramienta se establece el siguiente flujo de trabajo:

Tomando en cuenta que cada Desarrollador posee una rama con su nombre, lo primero que se debe de realizar al momento de iniciar a trabajar, es la verificación de la rama en la que se encuentra (`git branch`), si no se encuentra en la rama de Desarrollo asignada, es necesario cambiar dicha rama, pero primero antes de cambiarse a la rama de desarrollo asignada se deberá verificar el estado de la rama actual para corroborar que no existan modificaciones pendientes que guardar (`git status`).

Si existen modificaciones pendientes que guardar es necesario agregar dichas modificaciones (`git add --all`) para que sean incluidas en el commit que se debe de realizar para guardar los últimos cambios (`git commit -a -m "mensaje de descripcion"`).

Una vez guardadas las modificaciones es necesario observar nuevamente el estado para verificar que no quedaron cambios sin agregar, si no existen cambios, se procederá a cambiar a la rama asignada (`git checkout nombre_rama`).

El desarrollador ya en la rama correcta realizará las modificaciones al software según la funcionalidad que se le haya asignado codificar. El desarrollador no podrá guardar los cambios mientras no haya terminado una funcionalidad completa (componente, característica, una parte funcional del sistema), una vez terminado de realizar las modificaciones en el software, el desarrollador



tendrá que realizar las pruebas de subsistemas según la metodología de pruebas de software (apartado 8.9) y corroborar de que no existan errores, si existen errores, el desarrollador tendrá corregirlos y volver a realizar las pruebas de subsistemas.

Al tener el software libre de errores y haber finalizado una funcionalidad del sistema completa, el desarrollador tendrá que agregar los cambios realizados a Git (`git add --all`), guardar los cambios realizados (`git commit -a -m "mensaje"`), y sincronizar los cambios realizados en la rama local con su rama remota (`git push origin nombre_rama`) del repositorio que se encuentra en la nube, ya realizada la sincronización el desarrollador puede optar de realizar una unión (merge) de su rama con la rama master o de realizarla en otro momento, cuando tenga otras funcionalidades implementadas.

La unión de la rama con la rama master implica que el software se encuentra funcional y libre de errores, por lo tanto lo primero que se debe realizar es unir los cambios (si existen), de la rama master en la rama asignada (`git merge master`), si el proceso de unión genera conflictos, es necesario que el desarrollador los resuelva (`git mergetool nombre_archivo`), una vez que haya solucionado todos los conflictos, es necesario que se realicen las pruebas de regresión según la metodología establecida (apartado 8.9), y comprobar que no existen errores en el software.

Ya solucionado los errores es necesario cambiarse a la rama master local para aplicar las modificaciones realizadas y agregar las nuevas funcionalidades al software (`git checkout master`), una vez ingresado a la rama master es necesario aplicar las modificaciones de la rama asignada (`git merge nombre_rama`), y solucionar los conflictos de unión que se hayan generado (`git mergetool nombre_archivo`), cuando los conflictos se hayan solucionado es necesario actualizar la rama master local con cualquier cambio que otro desarrollador haya realizado en la rama master del repositorio remoto (`git pull origin master`), y solucionar cualquier conflicto que se haya generado (`git mergetool nombre_archivo`). Cuando todos los conflictos se encuentren solucionados y el software se encuentre libre de errores, el paso final es sincronizar los cambios de la rama master local, con la rama master remota (`git push origin master`). Para una mejor comprensión de la anteriormente descrito ver **Figura 8-27**, en donde se presenta el Flujo de Trabajo en GIT mediante un diagrama de procesos en notación BPMN.

Herramienta a utilizar para el Control de Versiones de Software:

- Git.
- Bitbucket.



8.11.2 Control de Versiones de Documentación.

El objetivo de utilizar un control de versiones en la documentación, es poder controlar los cambios que realiza cada miembro del equipo en los documentos pertenecientes al desarrollo del proyecto. Esto es útil cuando varios usuarios están editando el mismo documento, para lograr este objetivo se utilizará **SkyDrive** como servidor de almacenamiento en línea y el procesador de palabras Microsoft Word 2013. El uso de estas herramientas permitirá al equipo de trabajo realizar la edición del documento en tiempo real, brindando la ventaja de que no se crearán documentos por separados y no se tendrá que invertir tiempo en la unión de estos; se pueden realizar comentarios de errores detectados para su posterior corrección y a la vez se respeta el formato final del documento a presentar, que fue establecido en los estándares de documentación.

La **Figura 8-28** describe el funcionamiento del control de versiones de la documentación, en ella se puede observar que para realizar dicho control, el o los documentos del proyecto se encuentran almacenados en línea, a los cuales se acceden de forma simultánea desde diferentes dispositivos que tengan como procesador de palabras Microsoft Word, permitiendo así la edición de el o los documentos, sin necesidad de tener partes separadas de estos.



Figura 8-28 Control de Versión de Documentación.

En la Figura 8-29 se puede observar las acciones básicas que permite SkyDrive:

- **Crear:** Carpetas, Documentos de Word, Libro de Excel, Presentación de PowerPoint y Bloc de Notas de OneNote.
- **Cargar:** Despliega un cuadro de dialogo el cual permite subir archivos para sincronizarlos en la nube por medio de nuestra cuenta de correo Outlook.com.
- **Compartir:** Esta opción permite compartir carpetas y archivos con otros usuarios.
- **Acciones de Carpeta:** Permite descargar y eliminar carpetas, así como ver sus propiedades.

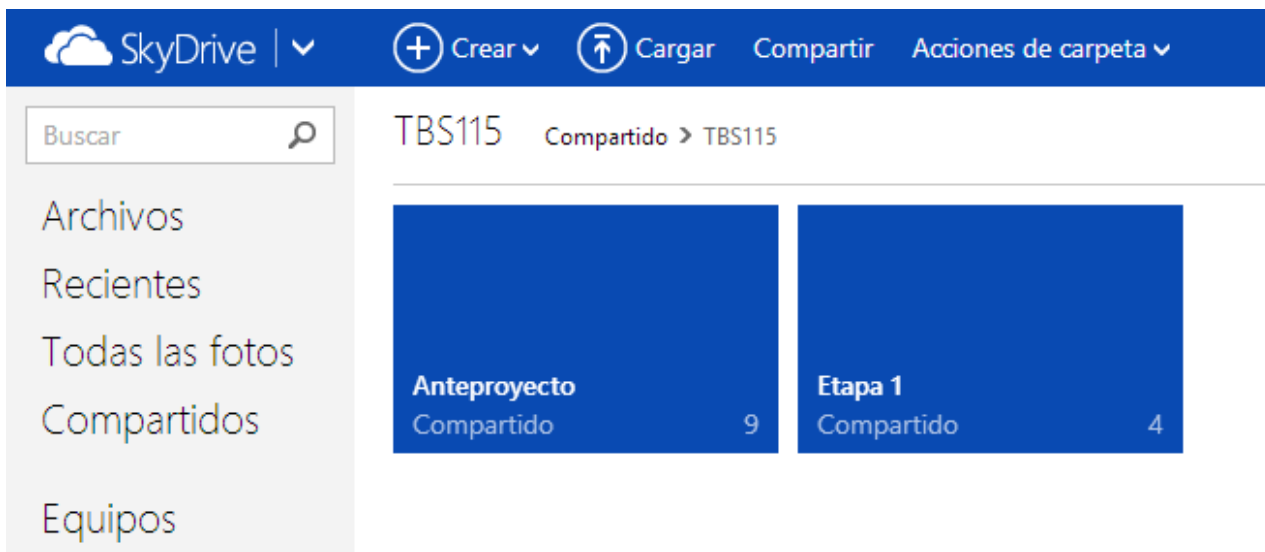


Figura 8-29 Acciones Básicas en SkyDrive.

Estructura de Carpetas

Para una mejor organización de los archivos del proyecto que se almacenarán en SkyDrive, se establecerá una estructura de carpetas que permita una organización intuitiva y lógica de estos, la cual se puede observar en la **Figura 8-30**, la carpeta raíz será nombrada TBS115, en correspondencia con el código asignado en el pensum de la carrera al Trabajo de Graduación. A esta carpeta tendrán acceso los miembros del equipo de desarrollo así como el asesor del Trabajo de Graduación.

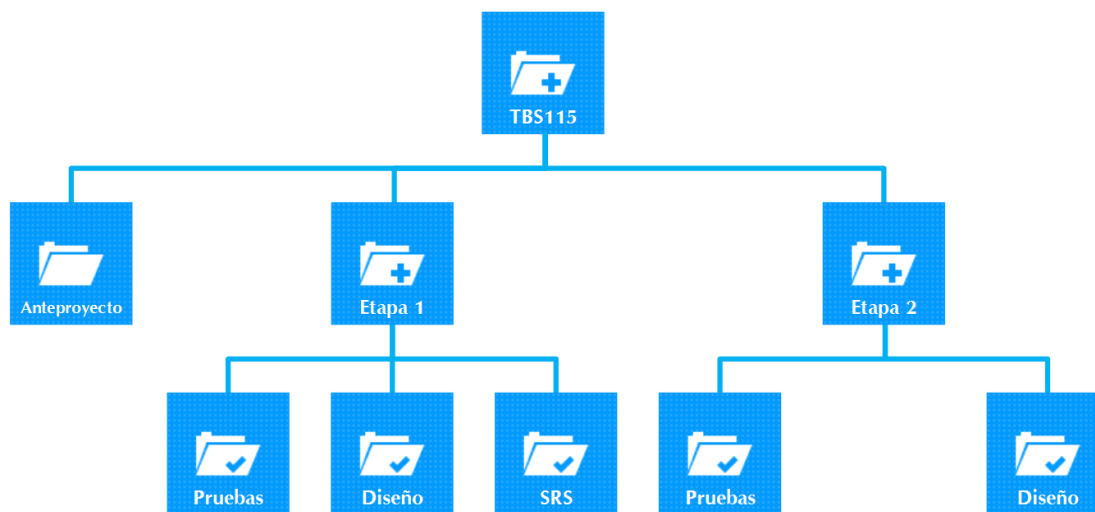


Figura 8-30 Estructura de Archivos en SkyDrive

Dentro de cada una de estas carpetas se podrán crear otras que tengan relación con ellas y dentro almacenar los archivos necesarios que correspondan, tal como se puede apreciar en la **Figura 8-30**.



SkyDrive proporciona un espacio gratuito de 7 GB de almacenamiento como se puede apreciar en la Figura 8-31, lo cual será suficiente para sincronizar los archivos a utilizar durante el desarrollo del proyecto.

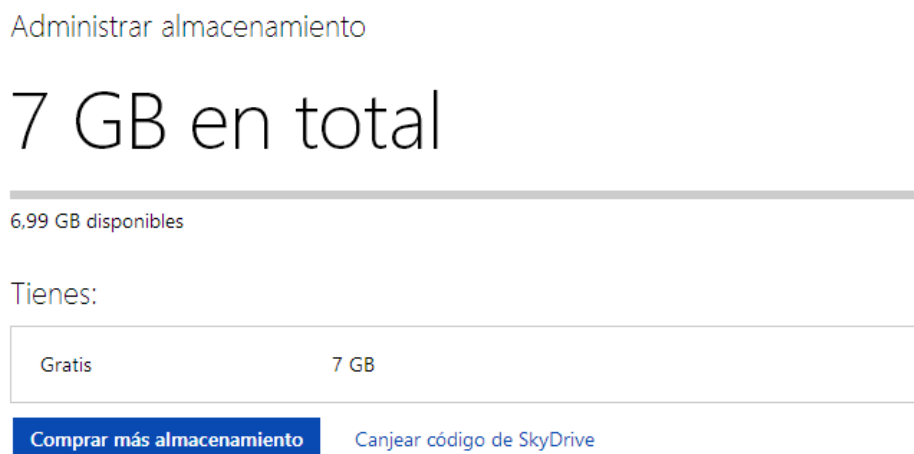


Figura 8-31 Almacenamiento Gratis SkyDrive

Herramienta a utilizar para el Control de Versiones de Documentación:

- Skydrive.
- Microsoft Word 2013

8.12 Almacenamiento en la Nube

El almacenamiento en la nube es un modelo de almacenamiento que permite que un mismo archivo esté disponible para uno o varios usuarios desde diferentes lugares. El objetivo de utilizar almacenamiento en la nube es que todos los archivos pertenecientes al desarrollo del proyecto estén accesibles para todo el equipo de trabajo desde cualquier lugar que se encuentren. Para lograr tal objetivo se utilizará el servicio que ofrece Dropbox el cual permitirá que desde cualquier dispositivo (Smartphone, PC, Laptop, Tablet) se tenga acceso a todos los documentos que se haya sincronizado, esto se puede apreciar en la Figura 8-32.

Estructura de Carpetas

Para una mejor organización de los archivos del proyecto que se almacenarán en Dropbox, se establecerá una estructura de carpetas que permita una organización intuitiva y lógica de estos, la cual se puede observar en la **Figura 8-33**, la carpeta raíz será nombrada TBS115, en correspondencia con el código asignado en el pensum de la carrera al Trabajo de Graduación. A esta carpeta tendrán acceso los miembros del equipo de desarrollo así como el asesor del Trabajo de Graduación.



Figura 8-32 Almacenamiento en la Nube

Dentro de cada uno de estos directorios se podrán crear otros que permitan mantener organizado los archivos pertenecientes al proyecto, una representación gráfica de la estructura de directorios establecidas se puede observar en la **Figura 8-33**.

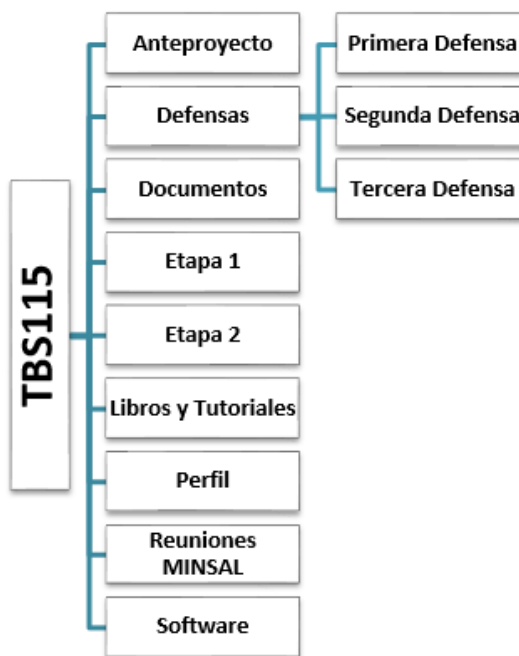


Figura 8-33 Estructura de Archivos en Dropbox



8.13 Descripción de Herramientas.

Además de la aproximación metodológica, uno de los factores determinantes en el éxito de un proyecto informático, lo constituye la acertada selección de las herramientas de trabajo que se utilizarán durante su desarrollo, es por eso que en la actualidad la administración de proyectos atiende cuidadosamente la designación de herramientas como una tarea de importancia. Por tal razón, para determinar las herramientas a utilizar en el proyecto se ha realizado un estudio detallado de las mismas, seleccionando la más conveniente según determinados factores⁴⁴.

A continuación se describen las herramientas a utilizar en el desarrollo del proyecto (Tabla 8-19):

Herramienta.	Descripción.
Suite de Microsoft Office 2013	Conjunto de herramientas de ofimática. Será utilizada para la elaboración de documentos y presentaciones.
Microsoft Visio 2013	Herramienta de dibujo vectorial. Será utilizada para la elaboración de esquemas y diagramas.
Dropbox 2.0.7	Es un servicio de alojamiento de archivos en la nube. El servicio permite almacenar y sincronizar archivos en línea entre ordenadores y compartir archivos y carpetas con otros. Será utilizado para almacenar y compartir documentos entre los miembros del equipo de desarrollo del proyecto.
Google Drive.	Es un servicio de almacenamiento de archivos en línea. Será utilizado para el almacenamiento de archivos y documentos correspondientes al proyecto.
Skydrive.	Es una herramienta que permite subir archivos de una computadora y almacenarlos en línea, y acceder a ellos desde un navegador web o un dispositivo local, así también, nos permite sincronizar documentos y editarlos de forma simultánea. Será utilizado para el almacenamiento de documentos y edición de los mismos. Se utilizará en conjunto con la suite de Microsoft Office 2013.
Google Calendar.	Es una agenda y calendario. Permite sincronizar con los contactos del servidor de correos gmail de manera que podamos compartir e invitar a eventos. Será utilizado para calendarización de reuniones, entregables, actividades y tareas correspondientes al proyecto.
Google Forms.	Es una herramienta útil que permite la creación de encuestas en línea, facilitando la tabulación y presentación de los datos obtenidos a través de gráficos. Será utilizado para la elaboración de encuestas en línea.
Notepad++ 6.3.2.	Es un editor de texto con soporte para varios lenguajes de programación, el cual trabaja en el sistema operativo Windows. Será utilizado para el desarrollo del software.
Navegador Web Google Chrome 26.	Es una herramienta que opera a través de internet, interpretando la información de archivos y sitios web para que estos puedan ser leídos. Será utilizado en el transcurso del proyecto.

⁴⁴ Para más detalle sobre el estudio de selección de herramientas realizado, ver Anexo 9 – Evaluación Técnica de Herramientas, del documento de Anteproyecto dentro del directorio /Documentos/1 – Anteproyecto/ del DVD anexo.



Herramienta.	Descripción.
Power Designer 16.1.	Es una aplicación para crear modelos de datos. Será utilizado en el diseño de la estructura de la base de datos.
Netbeans 7.3.	Es un entorno de desarrollo integrado libre, es una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas. Sera utilizado para el desarrollo del código fuente.
Sublime Text 2.0.1.	Es un editor de texto y editor de código fuente creado en Python, el cuál será utilizado en el Sistema Operativo Linux, como una alternativa a Notepad+ +. Será utilizado para el desarrollo del software.
Git 1.7.10.2	Es un software de control de versiones que ayuda en la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones cuando estas tienen un gran número de archivos de código fuente. Será utilizado en el desarrollo del software.
BitBucket.	Es un servicio de alojamiento basado en la web, para proyectos que utilizan sistemas de control de versiones, en el caso, se usará juntamente con Git. El servicio brinda alojamiento gratuito y permite tener un repositorio privado de forma gratuita para un máximo de 5 desarrolladores. Será utilizado en el desarrollo del software
AceProject.	Esta es una herramienta disponible en la web que permite la gestión de tareas y proyectos para equipos de trabajo. Sera utilizada para la gestión del proyecto
Bizagi 2.4.	Es una herramienta líder BPM que permite diseñar, modelar, integrar, automatizar y monitorear los procesos de negocio por medio de un ambiente gráfico. Será utilizada para el modelado de procesos del negocio.
Selenium IDE	Selenium IDE, es un ambiente de desarrollo integrado que permite la creación, edición, depuración y ejecución de Casos de Prueba. Este software esta implementado como una extensión del navegador Mozilla Firefox y su equivalente Iceweasel.
Mantis Bug Tracking	es una aplicación diseñada para ayudar a asegurar la calidad del software, por medio del seguimiento de los defectos o errores encontrados en la ejecución de las pruebas de software.

Tabla 8-19. Descripción de Herramientas



8.14 Cronograma de Actividades.

La estimación del Esfuerzo de las actividades que se muestran a continuación fue realizada con ayuda del Método Delphi⁴⁵.

A continuación se listan las macro actividades del proyecto.

Nombre de tarea	Duración	Comienzo	Fin
Sistema Informático SIM	231 días	lun 06/05/13	vie 31/01/14
Inicio	231 días	lun 06/05/13	jue 30/01/14
Etapa I	117 días	lun 06/05/13	jue 19/09/13
Análisis y Determinación de Requerimientos	44.88 días	lun 06/05/13	jue 27/06/13
Asignación de Requerimientos a Incrementos	1 día	jue 27/06/13	vie 28/06/13
Diseño Preliminar	14 días	vie 28/06/13	lun 15/07/13
Primer Incremento: Administración Elementos del Sistema Establecimientos, Dependencias y Personal.	27 días	lun 15/07/13	jue 15/08/13
Validación del Primer Incremento	7 días	jue 15/08/13	vie 23/08/13
Segundo Incremento: Administración de Roles y Usuarios.	27 días	lun 15/07/13	mié 14/08/13
Validación del Segundo Incremento	7 días	jue 15/08/13	jue 22/08/13
Tercer Incremento: Inventario Técnico, Empresas	9 días	vie 23/08/13	lun 02/09/13
Validación del Tercer Incremento	3 días	mar 03/09/13	jue 05/09/13
Elaboración de Documento Etapa I	4 días	vie 06/09/13	mar 10/09/13
Elaboración de Presentación Etapa I	7 días	mié 11/09/13	mié 18/09/13
Defensa Etapa I	1 día	jue 19/09/13	jue 19/09/13
Etapa II	76 días	lun 28/10/13	jue 23/01/14
Cuarto Incremento: Contrato	24 días	lun 28/10/13	sáb 23/11/13
Validación del Cuarto Incremento	6 días	lun 25/11/13	sáb 30/11/13
Quinto Incremento: Solicitud de Servicio	24 días	lun 28/10/13	sáb 23/11/13
Validación del Quinto Incremento	6 días	lun 25/11/13	sáb 30/11/13
Sexto Incremento: Ordenes de Trabajo	24 días	lun 28/10/13	sáb 23/11/13
Validación del Sexto Incremento	6 días	lun 25/11/13	sáb 30/11/13
Séptimo Incremento: Planificación Preventiva, Bitácora	24 días	lun 28/10/13	sáb 23/11/13
Validación del Séptimo Incremento	6 días	lun 25/11/13	sáb 30/11/13
Octavo Incremento: Creación de Reportes	6 días	lun 02/12/13	sáb 07/12/13
Validación del Octavo Incremento	6 días	lun 09/12/13	sáb 14/12/13
Elaboración del Plan de Implementación	15 días	lun 16/12/13	mié 01/01/14
Elaboración de Documento Etapa II	12 días	jue 02/01/14	mié 15/01/14
Elaboración de Presentación Etapa II	6 días	jue 16/01/14	mié 22/01/14
Defensa Etapa II	1 día	jue 23/01/14	jue 23/01/14
Correcciones Finales	6 días	vie 24/01/14	jue 30/01/14
Fin	0 días	vie 31/01/14	vie 31/01/14

⁴⁵ Para más detalle de la estimación del esfuerzo ver Anexo 14, página 197 del Documento de Anteproyecto que se encuentra dentro del directorio /Documentos/1 – Anteproyecto/ del DVD anexo.



A continuación se muestran las actividades detalladas del Proyecto:

Nombre de tarea	Duración	Comienzo	Fin
Sistema Informático SIM	231 días	lun 06/05/13	vie 31/01/14
Inicio	231 días	lun 06/05/13	jue 30/01/14
Etapa I	117 días	lun 06/05/13	jue 19/09/13
Análisis y Determinación de Requerimientos	44.88 días	lun 06/05/13	jue 27/06/13
Obtención y Análisis de Requerimientos	30 días	lun 06/05/13	lun 10/06/13
Determinación de Requerimientos del Sistema	15 días	lun 06/05/13	jue 23/05/13
Seguridad Física	15 días	lun 06/05/13	jue 23/05/13
Seguridad de Acceso	15 días	lun 06/05/13	jue 23/05/13
Interfaces	15 días	lun 06/05/13	jue 23/05/13
Operación	15 días	lun 06/05/13	jue 23/05/13
Mantenimiento	15 días	lun 06/05/13	jue 23/05/13
Desarrollo	15 días	lun 06/05/13	jue 23/05/13
Determinación de requerimientos de software	15 días	vie 24/05/13	lun 10/06/13
Usuario	15 días	vie 24/05/13	lun 10/06/13
Interfaces externas	15 días	vie 24/05/13	lun 10/06/13
Seguridad e Acceso	15 días	vie 24/05/13	lun 10/06/13
Datos y Base de Datos	15 días	vie 24/05/13	lun 10/06/13
Validación de Requerimientos	1 día	mar 11/06/13	mar 11/06/13
Análisis de Casos de Uso	1 día	mié 12/06/13	mié 12/06/13
Elaboración del Documento de Requerimientos	19.88 días	mié 05/06/13	jue 27/06/13
Asignación de Requerimientos a Incrementos	1 día	jue 27/06/13	vie 28/06/13
Diseño Preliminar	14 días	vie 28/06/13	lun 15/07/13
Diseño de Estándares	2 días	vie 28/06/13	lun 01/07/13
Diseño Arquitectónico	2 días	lun 01/07/13	mié 03/07/13
Diseño de Interfaz	3 días	mié 03/07/13	sáb 06/07/13
Diseño de Componentes	3 días	sáb 06/07/13	mié 10/07/13
Diseño de Base de Datos	4 días	mié 10/07/13	lun 15/07/13
Elaboración del Documento de Diseño	14 días	vie 28/06/13	lun 15/07/13
Primer Incremento: Administración Elementos del Sistema - Establecimientos, Dependencias y Personal.	27 días	lun 15/07/13	jue 15/08/13
Análisis detallado de requerimientos	3 días	lun 15/07/13	jue 18/07/13
Diseño Detallado	16 días	jue 18/07/13	mar 06/08/13
Diagramas UML y BPMN	16 días	jue 18/07/13	mar 06/08/13
Interfaz de Usuario y Casos de Prueba	16 días	jue 18/07/13	mar 06/08/13
Codificación y pruebas unitarias	24 días	jue 18/07/13	jue 15/08/13
Pruebas	24 días	jue 18/07/13	jue 15/08/13
Pruebas de integración de sub-sistemas	24 días	jue 18/07/13	jue 15/08/13
Pruebas de Integración del Sistema	24 días	jue 18/07/13	jue 15/08/13



Nombre de tarea	Duración	Comienzo	Fin
Validación del Primer Incremento	7 días	jue 15/08/13	vie 23/08/13
Presentación de incremento a Usuarios de Negocios	1 día	jue 15/08/13	vie 16/08/13
Retroalimentación	1 día	jue 15/08/13	vie 16/08/13
Pruebas de Aceptación	4 días	vie 16/08/13	mié 21/08/13
Correcciones de errores detectados	2 días	mié 21/08/13	vie 23/08/13
Segundo Incremento: Administración de Roles y Usuarios.	27 días	lun 15/07/13	mié 14/08/13
Análisis detallado de requerimientos	3 días	lun 15/07/13	mié 17/07/13
Diseño Detallado	16 días	jue 18/07/13	lun 05/08/13
Diagramas UML y BPMN	16 días	jue 18/07/13	lun 05/08/13
Interfaz de Usuario y Casos de Prueba	16 días	jue 18/07/13	lun 05/08/13
Codificación y pruebas unitarias	15.88 días	jue 18/07/13	lun 05/08/13
Pruebas	24 días	jue 18/07/13	mié 14/08/13
Pruebas de integración de sub-sistemas	24 días	jue 18/07/13	mié 14/08/13
Pruebas de Integración del Sistema	24 días	jue 18/07/13	mié 14/08/13
Validación del Segundo Incremento	7 días	jue 15/08/13	jue 22/08/13
Presentación de incremento a Usuarios de Negocios	1 día	jue 15/08/13	jue 15/08/13
Retroalimentación	1 día	jue 15/08/13	jue 15/08/13
Pruebas de Aceptación	4 días	vie 16/08/13	mar 20/08/13
Correcciones de errores detectados	2 días	mié 21/08/13	jue 22/08/13
Tercer Incremento: Inventario Técnico, Empresas	9 días	vie 23/08/13	lun 02/09/13
Análisis detallado de requerimientos	1 día	vie 23/08/13	vie 23/08/13
Diseño Detallado	8 días	sáb 24/08/13	lun 02/09/13
Diagramas UML y BPMN	8 días	sáb 24/08/13	lun 02/09/13
Interfaz de Usuario y Casos de Prueba	8 días	sáb 24/08/13	lun 02/09/13
Codificación y pruebas unitarias	8 días	sáb 24/08/13	lun 02/09/13
Pruebas	8 días	sáb 24/08/13	lun 02/09/13
Pruebas de integración de sub-sistemas	8 días	sáb 24/08/13	lun 02/09/13
Pruebas de Integración del Sistema	8 días	sáb 24/08/13	lun 02/09/13
Validación del Tercer Incremento	3 días	mar 03/09/13	jue 05/09/13
Presentación de incremento a Usuarios de Negocios	1 día	mar 03/09/13	mar 03/09/13
Retroalimentación	1 día	mar 03/09/13	mar 03/09/13
Pruebas de Aceptación	1 día	mié 04/09/13	mié 04/09/13
Correcciones de errores detectados	1 día	jue 05/09/13	jue 05/09/13
Elaboración de Documento Etapa I	4 días	vie 06/09/13	mar 10/09/13
Elaboración de Presentación Etapa I	7 días	mié 11/09/13	mié 18/09/13
Defensa Etapa I	1 día	jue 19/09/13	jue 19/09/13
Etapa II	76 días	lun 28/10/13	jue 23/01/14
Cuarto Incremento: Contrato	24 días	lun 28/10/13	sáb 23/11/13
Análisis detallado de requerimientos	6 días	lun 28/10/13	sáb 02/11/13



Nombre de tarea	Duración	Comienzo	Fin
Diseño Detallado	6 días	lun 04/11/13	sáb 09/11/13
Diagramas UML y BPMN	6 días	lun 04/11/13	sáb 09/11/13
Interfaz de Usuario y Casos de Prueba	6 días	lun 04/11/13	sáb 09/11/13
Codificación y pruebas unitarias	12 días	lun 11/11/13	sáb 23/11/13
Pruebas	12 días	lun 11/11/13	sáb 23/11/13
Pruebas de integración de sub-sistemas	12 días	lun 11/11/13	sáb 23/11/13
Pruebas de Integración del Sistema	12 días	lun 11/11/13	sáb 23/11/13
Validación del Cuarto Incremento	6 días	lun 25/11/13	sáb 30/11/13
Presentación de incremento a Usuarios de Negocios	1 día	lun 25/11/13	lun 25/11/13
Retroalimentación	1 día	lun 25/11/13	lun 25/11/13
Pruebas de Aceptación	2 días	mar 26/11/13	mié 27/11/13
Correcciones de errores detectados	3 días	jue 28/11/13	sáb 30/11/13
Quinto Incremento: Solicitud de Servicio	24 días	lun 28/10/13	sáb 23/11/13
Análisis detallado de requerimientos	6 días	lun 28/10/13	sáb 02/11/13
Diseño Detallado	6 días	lun 04/11/13	sáb 09/11/13
Diagramas UML y BPMN	6 días	lun 04/11/13	sáb 09/11/13
Interfaz de Usuario y Casos de Prueba	6 días	lun 04/11/13	sáb 09/11/13
Codificación y pruebas unitarias	12 días	lun 11/11/13	sáb 23/11/13
Pruebas	12 días	lun 11/11/13	sáb 23/11/13
Pruebas de integración de sub-sistemas	12 días	lun 11/11/13	sáb 23/11/13
Pruebas de Integración del Sistema	12 días	lun 11/11/13	sáb 23/11/13
Validación del Quinto Incremento	6 días	lun 25/11/13	sáb 30/11/13
Presentación de incremento a Usuarios de Negocios	1 día	lun 25/11/13	lun 25/11/13
Retroalimentación	1 día	lun 25/11/13	lun 25/11/13
Pruebas de Aceptación	2 días	mar 26/11/13	mié 27/11/13
Correcciones de errores detectados	3 días	jue 28/11/13	sáb 30/11/13
Sexto Incremento: Ordenes de Trabajo	24 días	lun 28/10/13	sáb 23/11/13
Análisis detallado de requerimientos	6 días	lun 28/10/13	sáb 02/11/13
Diseño Detallado	6 días	lun 04/11/13	sáb 09/11/13
Diagramas UML y BPMN	6 días	lun 04/11/13	sáb 09/11/13
Interfaz de Usuario y Casos de Prueba	6 días	lun 04/11/13	sáb 09/11/13
Codificación y pruebas unitarias	12 días	lun 11/11/13	sáb 23/11/13
Pruebas	12 días	lun 11/11/13	sáb 23/11/13
Pruebas de integración de sub-sistemas	12 días	lun 11/11/13	sáb 23/11/13
Pruebas de Integración del Sistema	12 días	lun 11/11/13	sáb 23/11/13
Validación del Sexto Incremento	6 días	lun 25/11/13	sáb 30/11/13
Presentación de incremento a Usuarios de Negocios	1 día	lun 25/11/13	lun 25/11/13
Retroalimentación	1 día	lun 25/11/13	lun 25/11/13
Pruebas de Aceptación	2 días	mar 26/11/13	mié 27/11/13



Nombre de tarea	Duración	Comienzo	Fin
Correcciones de errores detectados	3 días	jue 28/11/13	sáb 30/11/13
Séptimo Incremento: Planificación Preventiva, Bitácora	24 días	lun 28/10/13	sáb 23/11/13
Análisis detallado de requerimientos	6 días	lun 28/10/13	sáb 02/11/13
Diseño Detallado	6 días	lun 04/11/13	sáb 09/11/13
Diagramas UML y BPMN	6 días	lun 04/11/13	sáb 09/11/13
Interfaz de Usuario y Casos de Prueba	6 días	lun 04/11/13	sáb 09/11/13
Codificación y pruebas unitarias	12 días	lun 11/11/13	sáb 23/11/13
Pruebas	12 días	lun 11/11/13	sáb 23/11/13
Pruebas de integración de sub-sistemas	12 días	lun 11/11/13	sáb 23/11/13
Pruebas de Integración del Sistema	12 días	lun 11/11/13	sáb 23/11/13
Validación del Séptimo Incremento	6 días	lun 25/11/13	sáb 30/11/13
Presentación de incremento a Usuarios de Negocios	1 día	lun 25/11/13	lun 25/11/13
Retroalimentación	1 día	lun 25/11/13	lun 25/11/13
Pruebas de Aceptación	2 días	mar 26/11/13	mié 27/11/13
Correcciones de errores detectados	3 días	jue 28/11/13	sáb 30/11/13
Octavo Incremento: Creación de Reportes	6 días	lun 02/12/13	sáb 07/12/13
Diseño y Creación de Reportes	6 días	lun 02/12/13	sáb 07/12/13
Pruebas	6 días	lun 02/12/13	sáb 07/12/13
Validación del Octavo Incremento	6 días	lun 09/12/13	sáb 14/12/13
Presentación de incremento a Usuarios de Negocios	1 día	lun 09/12/13	lun 09/12/13
Retroalimentación	1 día	lun 09/12/13	lun 09/12/13
Correcciones de errores detectados	5 días	mar 10/12/13	sáb 14/12/13
Elaboración del Plan de Implementación	15 días	lun 16/12/13	mié 01/01/14
Elaboración de Documento Etapa II	12 días	jue 02/01/14	mié 15/01/14
Elaboración de Presentación Etapa II	6 días	jue 16/01/14	mié 22/01/14
Defensa Etapa II	1 día	jue 23/01/14	jue 23/01/14
Correcciones Finales	6 días	vie 24/01/14	jue 30/01/14
Fin	0 días	vie 31/01/14	vie 31/01/14



8.15 Planeación de recursos y costo del proyecto.

La planeación de Recursos es el proceso de definir los recursos necesarios que se utilizarán en el desarrollo del proyecto y como se asignarán a las etapas del ciclo de vida de desarrollo de proyectos. La planeación de recursos se basa en un conjunto de estimados.

Un estimado es un valor de medición acerca de un elemento de un proyecto, generado a partir del juicio de una persona. El objetivo es comunicar formalmente la cantidad de recursos que se requieren⁴⁶.

Los estimados siempre son imperfectos y normalmente se afinan a medida que avanza el proyecto.

8.15.1 Presupuesto y costo del Proyecto.

En la Tabla 8-20 se muestra un resumen de la estimación del presupuesto necesario y costo para el desarrollo del sistema SIM⁴⁷.

Recurso	Total (USD \$)
Fuerza de programación	117,734.40
Recursos de Hardware	606.95
Equipo de Desarrollo	15,903.88
Sub-total sin Factores	134,245.23
Factores Contingenciales (40%)	53,698.09
Sub-total con Fact. Contigencial	187,943.32
Factores Ponderables (154%)	289,432.72
Sub-total con Fac. Ponderable	477,376.03
Costos indirectos	984.40
Total del proyecto	478,360.43

Tabla 8-20. Presupuesto del Proyecto SIM.

Lo anterior muestra que el presupuesto necesario y el costo para el desarrollo del Sistema SIM es de \$ **478,360.43**.

⁴⁶ Carlos Ernesto García. *Gerencia Informática* (6 edición) San Salvador. Guía para estimaciones. Editorial Informatik, S.A de C.V

⁴⁷ Para conocer a detalle la Estimación de Costos del Proyecto y los factores que inciden en este, ver Anexo 16 del Documento de Anteproyecto que se encuentra dentro del directorio /Documentos/1 – Anteproyecto/ del DVD anexo.



9 Capítulo III

Análisis

Etapa que tiene por objetivo realizar un estudio de los elementos que se ven involucrados en el dominio del negocio del Sistema a desarrollar y cuyo resultado es la presentación de una propuesta de solución al problema planteado y contribuir a la Especificación de Requerimientos. Para lograr esto, se realiza una investigación de la situación actual, dando como resultado una propuesta de solución a la problemática establecida, mediante la utilización de diferentes herramientas de diagnósticos que permiten formular un correcto planteamiento del problema.

En este capítulo, se muestra la utilización de diferentes técnicas en la etapa de Análisis del proyecto de desarrollo del SIM, entre las cuales se encuentran las técnicas de diagnóstico: Análisis Causa-Efecto y Análisis FODA.

La utilización de estas técnicas en conjunto con el estudio de la situación actual permitió establecer un Planteamiento del Problema mediante el método de la Caja Negra, para que posteriormente a través del Enfoque de Sistema se estableciera una Propuesta de Solución al problema planteado.



9.1 Análisis de la Situación Actual

9.1.1 Descripción de la Organización

9.1.1.1 Organización del Mantenimiento en el Nivel Superior⁴⁸

La Figura 9-1 muestra una parte de la estructura organizativa del Ministerio de Salud, en la que se destacan las principales gerencias, y unidades con las que el Área de Mantenimiento General tiene relación directa, así como las secciones que dependen de ésta y que están relacionadas con los procesos de mantenimiento⁴⁹.

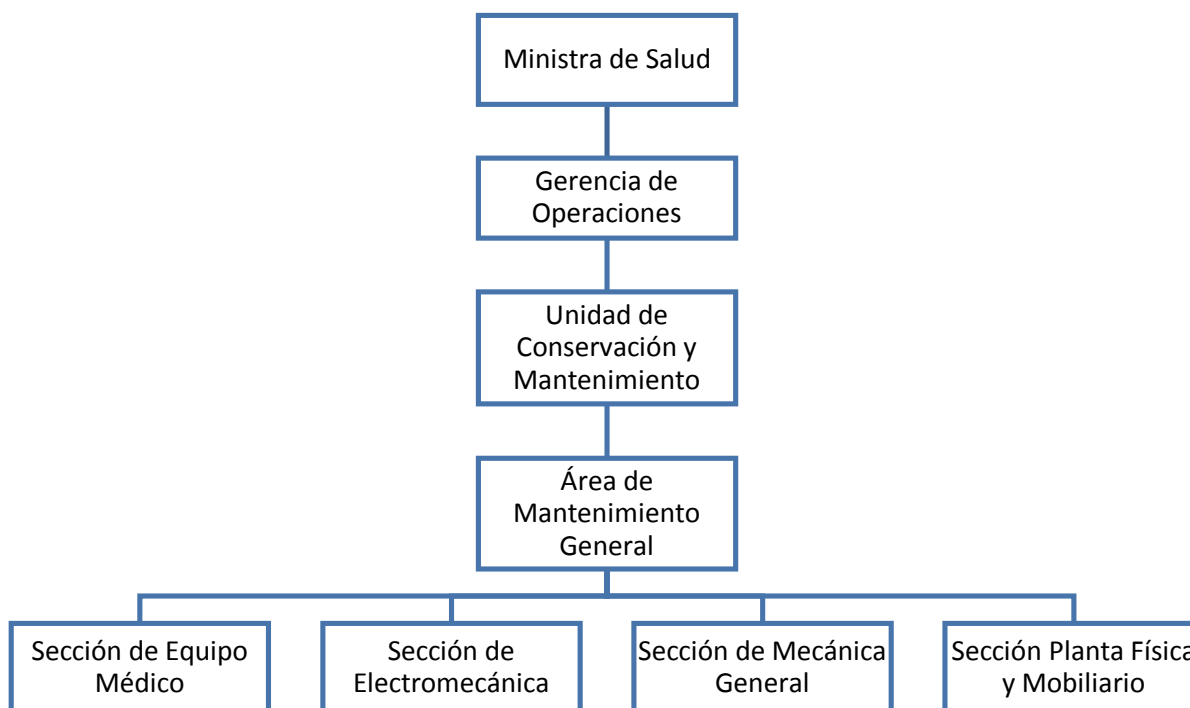


Figura 9-1 Estructura Orgánica para el Nivel Superior.

Gerencia General de Operaciones.

Es la entidad institucional responsable de velar por la administración integral de los recursos humanos y materiales, mediante la difusión, facilitación y verificación de la normativa legal establecida para tal efecto por las diferentes instancias reguladoras del Estado, y se encuentra conformada por las

⁴⁸ Ministerio de Salud (2012). *Manual de Organización y Funciones de la Gerencia General de Operaciones*. Recuperado de:

http://asp.salud.gob.sv/regulacion/pdf/manual/manual_organizacion_y_funciones_gerencia_general.pdf

⁴⁹ Para más detalle sobre la estructura orgánica del Ministerio de Salud y del Área de Mantenimiento General ver Anexo 2 – Estructura Organizativa del Ministerio de Salud y Anexo 3 – Estructura Organizativa de la Unidad de Conservación y Mantenimiento del Documento de Anteproyecto que se encuentra dentro del directorio /Documentos/1 – Anteproyecto/ del DVD anexo.



Unidades de Administración de Recursos Humanos, de Conservación y Mantenimiento y de Abastecimientos.

Unidad de Conservación y Mantenimiento

Misión

Instancia responsable de la conservación y mantenimiento de las instalaciones, infraestructura y equipos; proporcionar servicios de transporte, vigilancia y control de bienes muebles e inmuebles de las dependencias del Nivel Superior, asegurando así la disponibilidad oportuna de estos recursos para beneficio de la población.

Visión

Ser la unidad fortalecida para la gestión y provisión de servicios de conservación y mantenimiento, en forma oportuna y eficiente a los establecimientos del MINSAL, contando para ello con los recursos humanos, financieros y tecnológicos, que permitan crear el entorno adecuado en armonía con el medio ambiente, para ofrecer servicios de Salud con calidad a la población.

Objetivo General

Proveer el adecuado funcionamiento de la infraestructura, instalaciones, equipos y demás recursos materiales y tecnológicos, que se requieren para la eficiente prestación de servicios en las diferentes dependencias del nivel superior, y apoyar a los niveles regional y local, aplicando la normativa técnica existente.

Las áreas que dependen de la Unidad de Conservación y Mantenimiento son las siguientes:

- Área de Mantenimiento General.
- Área de Servicios Auxiliares
- Área de Transporte
- Área de Activo fijo

Área de Mantenimiento General

Objetivo

Planificar, coordinar y controlar el mantenimiento de los equipos, instalaciones e infraestructura de las dependencias del Nivel Superior y apoyar a los Niveles Regional y Local.

Funciones

Algunas de las funciones del Área de Mantenimiento General son las siguientes:

- Proponer y supervisar la ejecución de normas y lineamientos técnicos de mantenimiento.
- Elaborar el plan operativo anual del Área de Mantenimiento General del nivel superior.
- Elaborar el presupuesto anual del Área de Mantenimiento General.



- Identificar las necesidades de repuestos e insumos que deben tenerse en existencia.
- Elaborar programas de mantenimiento preventivo para el nivel superior y proporcionar apoyo en los niveles regional y local.
- Elaborar presupuestos para trabajos especializados, solicitados por las diferentes dependencias del nivel superior, regional y local.
- Proporcionar asistencia técnica especializada a los establecimientos de Salud del nivel regional y local, que carecen del recurso idóneo.
- Dar seguimiento a los programas de mantenimiento preventivo de los niveles regional y local.

Secciones del Área de Mantenimiento General

Las secciones que dependen del Área de Mantenimiento General son las siguientes:

- Sección Administrativa.
- Sección de Capacitación.
- Sección de Equipo Médico.
- Sección de Electromecánica.
- Sección de Mecánica General.
- Sección de Planta Física y Mobiliario.

Las secciones específicas que brindan servicios de mantenimiento por parte del Área de Mantenimiento General son las siguientes:

Sección de Equipo Médico

Objetivo

Proporcionar mantenimiento preventivo a los equipos médicos de las dependencias del nivel superior, y apoyo al nivel regional y local, y la asistencia técnica conveniente para el buen funcionamiento de los mismos.

Sección de Electromecánica

Objetivo

Asegurar el correcto funcionamiento de los equipos e instalaciones industriales de los establecimientos de Salud, tales como generadores eléctricos, aire acondicionado y refrigeración, sistemas eléctricos en general, equipos de bombeo de agua y equipos de fumigación, mediante la prestación de los servicios de mantenimiento preventivo y correctivo, en coordinación con el Área de Mantenimiento del nivel regional y local.



Sección de Mecánica General

Objetivo

Mantener en buen funcionamiento los equipos de calderas, lavandería, autoclaves, equipos de alimentación y dietas, incineradores, maquinado y obra de banco, de las dependencias del nivel superior, regional y local.

Sección de Planta Física y Mobiliario

Objetivo

Realizar trabajos de mantenimiento en sistemas hidráulicos, pintura, carpintería, tapicería, evaluación de obras civiles, remodelaciones y pequeñas construcciones, en las dependencias de los niveles superior, regional y local.

9.1.1.2 Organización del Mantenimiento en el Nivel Regional

El Nivel Regional está conformado por cinco Direcciones Regionales de Salud:

- Dirección Regional de Salud Occidental
- Dirección Regional de Salud Central
- Dirección Regional de Salud Metropolitana
- Dirección Regional de Salud Paracentral
- Dirección Regional de Salud Oriental

Estas regiones tienen bajo su cargo la gestión y control de los recursos asignados al Primer y Segundo Nivel de Atención, pero los servicios de mantenimiento solo los brindan al Primer Nivel que está compuesto de las Unidades Comunitarias de Salud Familiar (UCSF), que están bajo su jurisdicción. Los servicios de mantenimiento del Segundo Nivel son proporcionados por los Departamentos de Mantenimiento Interno, que son propios de cada Hospital. La Figura 9-2 muestra una parte de la estructura organizativa de una Dirección Regional de Salud, en la cual se puede observar la división y unidad de las que depende el Departamento de Mantenimiento, este último a diferencia del Nivel Superior y Local, no posee secciones definidas.

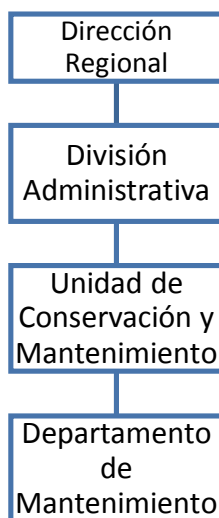


Figura 9-2 Estructura Organizativa para el Nivel Regional.

9.1.1.3 Organización del Mantenimiento en el Nivel Local

La estructura organizativa de los 30 Hospitales⁵⁰ que forman parte del Ministerio de Salud, tienen en común que en todos existe un Departamento de Mantenimiento Interno, el cual depende de la División Administrativa del respectivo Hospital, a diferencia del Nivel Superior, el Departamento de Mantenimiento Interno está compuesto por tres Secciones, las cuales pueden observarse en la Figura 9-3.



Figura 9-3 Estructura Organizativa para el Nivel Local.

⁵⁰ Para más información de la estructura organizativa de los Hospitales ver Anexo 4 – Estructura Orgánica de Establecimientos de Segundo y Tercer Nivel del Documento de Anteproyecto que se encuentra dentro del Directorio /Documentos/1 – Anteproyecto/ del DVD Anexo.



9.1.2 Proceso Principal de los Servicios de Mantenimiento

La gestión de los servicios de mantenimiento de los equipos, infraestructura e instalaciones del MINSAL, se logra a través de las acciones combinadas y coordinadas del Área de Mantenimiento General, los Departamentos de Mantenimiento de las Regiones, y los Departamentos de Mantenimiento Interno de los Hospitales, siendo complementadas con servicios contratados a empresas privadas.

Tomando en cuenta lo anterior, recae en el Departamento de Mantenimiento Interno, el mayor volumen de responsabilidades relacionadas con el mantenimiento preventivo y correctivo de complejidad intermedia, y en el Área Central, aquellas funciones de apoyo técnico y de asesoría que requieran servicios de mantenimiento más complejos.

Actualmente el Área de Mantenimiento General, atiende solicitudes de servicio de los treinta Hospitales a nivel Nacional y de las Regionales cuando la ocasión lo amerita.

En la Figura 9-4, se presenta el proceso principal que se realiza para el Servicio de Mantenimiento, mediante un Diagrama de Procesos de Negocio utilizando la notación BPMN (para conocer la nomenclatura de BPMN ver apartado 8.6.3) y a continuación se describe este proceso.

Descripción del Proceso de Servicio de Mantenimiento.

En este proceso existen dos clases de servicios de mantenimiento, el Mantenimiento Preventivo y el Mantenimiento Correctivo.

Para el Mantenimiento Preventivo, el Jefe de Mantenimiento debe elaborar el Plan Anual de Mantenimiento Preventivo para cada equipo. El Plan de Mantenimiento Preventivo se compone de diversas Rutinas de Mantenimiento Preventivo que deben aplicarse a determinado equipo en ciertos plazos de tiempo. En base a este Plan, el Jefe de Mantenimiento crea semanalmente el Programa de Trabajo Semanal, es decir, las Rutinas que un determinado equipo necesita en esa semana.

Las Rutinas de Mantenimiento Preventivo son asignadas a los Técnicos de Mantenimiento para ser aplicadas por ellos, mediante Órdenes de Trabajo. El Técnico entonces, al recibir la Orden de Trabajo, necesita revisar la correspondiente Rutina de Mantenimiento para preparar los materiales y herramientas que utilizará para completar satisfactoriamente la Rutina. Seguidamente, el Técnico visita el lugar donde se encuentra ubicado el equipo y notifica al Jefe del Servicio (Unidad, Área o Departamento) u Operador del Equipo (persona que tiene a cargo el equipo en el lugar visitado) sobre el trabajo a realizar. Luego de esto, el Técnico verifica el equipo y realiza la rutina correspondiente.

Al finalizar el Técnico de realizar sus actividades, el Jefe de Servicio (Unidad, Área o Departamento) u Operador de Equipo debe dar conformidad del servicio realizado. Luego, el Técnico ingresa en la Ficha de Vida del equipo el servicio realizado y notifica al Jefe de Mantenimiento sobre el trabajo realizado.

Cuando un equipo, instalación o infraestructura presenta problemas o fallas en una Unidad (Servicio, Área o Departamento), el Jefe de dicha Unidad es el encargado de elaborar y enviar una Solicitud de Servicio de Mantenimiento al Departamento de Mantenimiento correspondiente. En este caso entonces, se realiza un Mantenimiento Correctivo. Cuando el Jefe de Mantenimiento recibe la Solicitud, la revisa y determina si procede o no el trabajo solicitado. Si no procede debido a que la



solicitud no es relevante, es decir, que lo solicitado no califica realmente para ser catalogado como una Solicitud de Mantenimiento, lo informa al solicitante. Pero, si una Solicitud no procede debido a la capacidad de resolución del Departamento de Mantenimiento, esta es transferida al Nivel Superior. Si la Solicitud Procede, se crea un Orden de Trabajo que también es asignada a un Técnico.

Con la Orden de Trabajo, el Técnico visita el lugar donde se encuentra ubicado el equipo para su debida inspección. Si el Técnico puede realizar el trabajo solicitado (reparación, ajustes) en el momento, lo ejecuta. De lo contrario, si necesita repuestos o materiales estos deben ser gestionados. Al obtener los materiales o repuestos requeridos, el Técnico ejecuta el trabajo.

Al finalizar el Técnico de realizar sus actividades, el Jefe de Servicio (Unidad, Área o Departamento) u Operador de Equipo debe dar conformidad del servicio realizado. Luego, el Técnico ingresa en la Ficha de Vida del equipo el servicio realizado y notifica al Jefe de Mantenimiento sobre el trabajo realizado.



Proceso de Servicio de Mantenimiento de la Situación Actual.

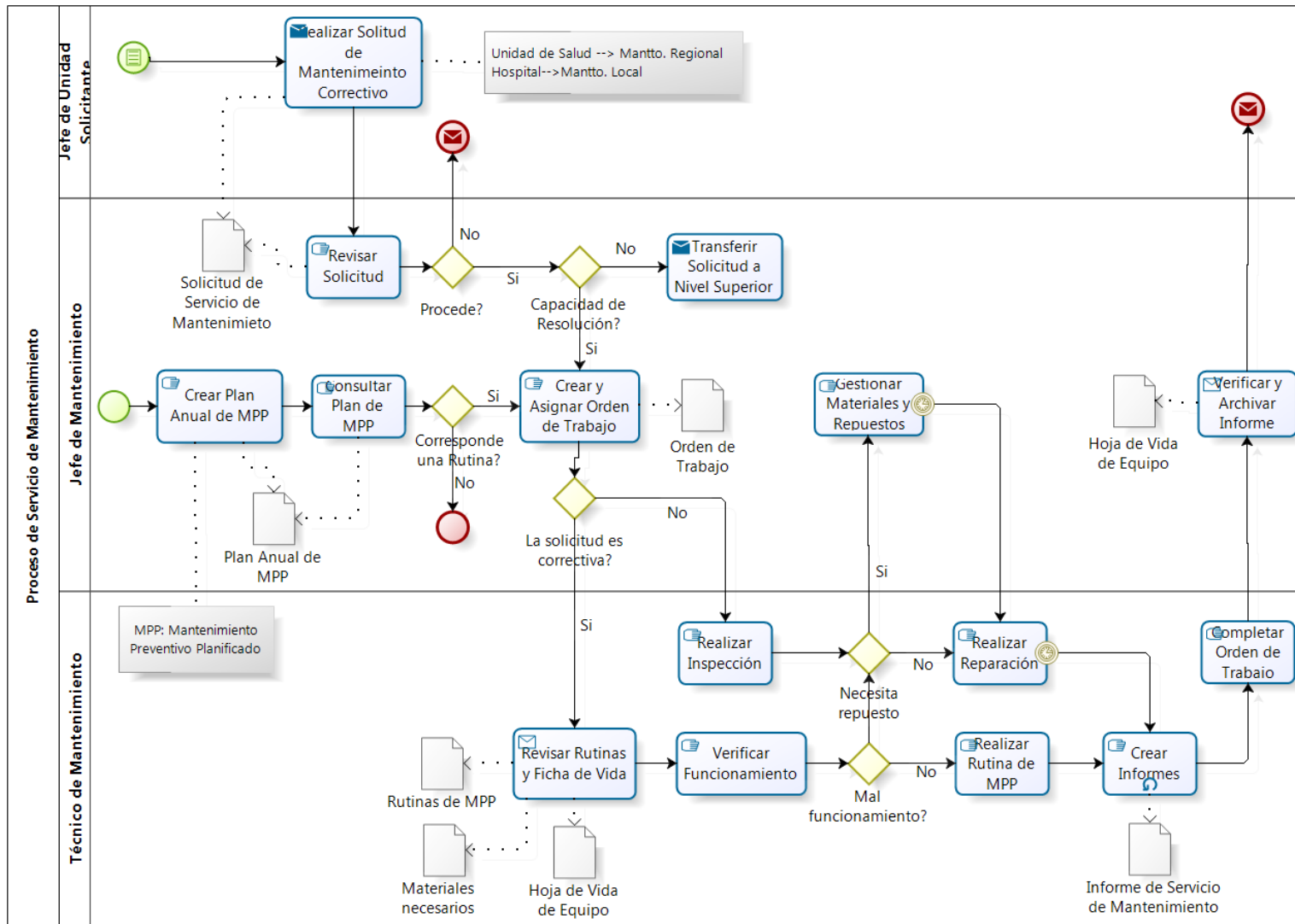


Figura 9-4. Proceso Principal de Servicio de Mantenimiento



9.1.3 Enfoque de Sistemas de la Situación Actual

La Figura 9-5 muestra el Diagrama del Enfoque de Sistemas que representa la situación actual del Área de Mantenimiento General y los Departamentos de Mantenimiento Regional y Local:

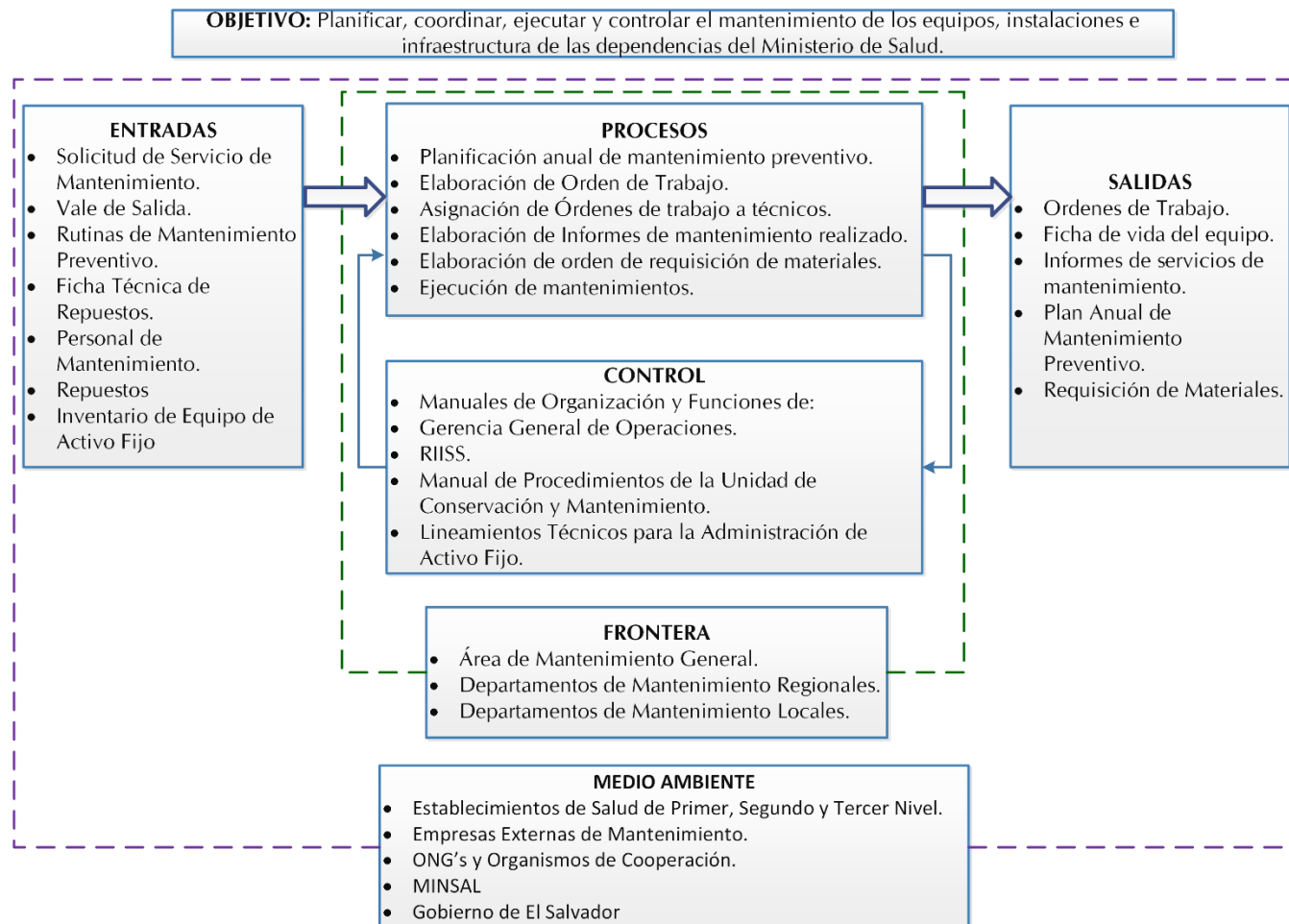


Figura 9-5 Enfoque de Sistemas para la situación actual.



El Diagrama de Enfoque de Sistemas presentado en la Figura 9-5, es una representación gráfica del comportamiento actual de los servicios de mantenimiento brindados por el Área de Mantenimiento General y los Departamentos de Mantenimiento Regional y Local. Los elementos del Enfoque de Sistema se describen a continuación:

- a) **Objetivo del sistema:** Planificar, coordinar, realizar y controlar el mantenimiento de los equipos, instalaciones e infraestructura de las dependencias del Nivel Superior y apoyar a los Niveles Regional y Local.
- b) **Salidas:** Son los resultados actuales que se obtienen de brindar los servicios de mantenimiento.
 - Ordenes de trabajo.
 - Ficha de vida del equipo.
 - Informes de servicios de mantenimiento.
 - Plan Anual de Mantenimiento Preventivo.
 - Requisición de Materiales.
- c) **Entradas:** Contiene todos los insumos necesarios para la prestación de servicios de mantenimiento.
 - Solicitud de servicio de mantenimiento.
 - Vale de salida.
 - Rutinas de Mantenimiento Preventivo.
 - Ficha técnica de repuestos.
 - Personal de mantenimiento.
 - Repuestos.
 - Inventario de Equipos de Activo Fijo.
- d) **Procesos:** Actividad que transforma los insumos (entradas) en salidas.
 - Planificación anual de mantenimiento preventivo.
 - Elaboración de orden de trabajo.
 - Descripción de mantenimiento realizado en orden de trabajo.
 - Elaboración de órdenes de requisición de materiales.
- e) **Control:** Conjunto de reglamentos, leyes, normas, estándares, entre otros que establecen la manera en que se debe de realizar los procesos de servicio de mantenimiento, y la forma en que se debe de actuar ante determinadas eventos.
 - Manuales de Organización y Funciones de:
 - Gerencia General de Operaciones.
 - RIISS.
 - Manual de Procedimientos de la Unidad de Conservación y Mantenimiento.
 - Lineamientos Técnicos para la Administración de Activo Fijo.
Para ver una descripción de los Manuales y Lineamientos mencionados vea el apartado 7.3.



- f) **Frontera:** Delimitador que permite clasificar que elementos son internos (propios del sistema), y que elementos son externos a éste (pertenecientes al medio ambiente).

Dado que el proceso de mantenimiento se realiza en el Nivel Superior, Regional y Local se tiene como frontera lo siguiente:

- Área de Mantenimiento General
- Departamento de Mantenimiento Regionales
- Departamentos de Mantenimiento Locales

- g) **Medio Ambiente:** Elementos externos con los que el sistema interactúa y que proporcionan insumos o entradas al sistema que son necesarios para procesamiento y generación de salidas del sistema.

El medio ambiente con el que interactúa el sistema actual está formado por:

- Establecimientos de Salud de primer, segundo y tercer nivel.
- Empresas Externas de Mantenimiento.
- Organismos no Gubernamentales (ONGs).
- Organismos de Cooperación.
- MINSAL.
- Gobierno de El Salvador.



9.2 Planteamiento del Problema

9.2.1 Diagnóstico

Por medio del diagnóstico se pretende conocer las causas que originan el problema actual, esto permite enfocar los esfuerzos en afrontar esas causas y cumplir con los objetivos del proyecto. Para ello se utilizan dos técnicas: El análisis Causa-Efecto y el Análisis FODA. Así mismo, este análisis se basa en la información obtenida de las entrevistas y las encuestas realizadas⁵¹.

9.2.1.1 Análisis Causa-Efecto

Las causas identificadas que inciden en el problema se clasifican en tres categorías, y se presentan a continuación:

Procedimientos

- Actualmente, las solicitudes de servicio de mantenimiento se realizan manualmente, es decir, a través de memorándum, correo electrónico o por medio de llamadas telefónicas, en las cuales se detalla lo que se necesita.
- La atención de las solicitudes de servicio, por medio de la asignación de órdenes de trabajo a los técnicos de mantenimiento, se realiza de forma manual, dependiendo del nivel de prioridad de la solicitud y de la carga de trabajo de los técnicos disponibles. Cuando se tienen varias solicitudes, y el recurso humano disponible es escaso, este trabajo se torna complejo.
- El encargado o jefe del Departamento de Mantenimiento a quien se le envió la solicitud, da seguimiento a las órdenes de trabajo asignadas lo que se realiza de forma manual; este seguimiento se puede perder o desviar al no hacerlo de manera controlada, más aun si son varias solicitudes que se atienden. De igual forma, no se posee el control y no se tiene una visión clara del avance de los técnicos en sus órdenes de trabajo.
- La falta de comunicación directa entre el área de mantenimiento y la dependencia que ha realizado la solicitud de servicio (Hospitales y Unidades Comunitarias de Salud Familiar), genera un desconocimiento del estado de la solicitud de mantenimiento.
- Los Departamentos de Mantenimiento Interno (Hospitales), en su mayoría no siguen un plan de mantenimiento, debido a que no posee el recurso adecuado (materiales, recurso humano) para poder brindar los servicios de mantenimiento, o no se realiza una planificación de mantenimiento preventivo.

⁵¹ Para más detalles sobre las entrevistas y encuestas ver Anexo 5 – Entrevistas y Anexo 6 – Encuestas sobre servicios de mantenimiento de equipos del Documento de Anteproyecto que se encuentra en el directorio /Documentos/1 – Anteproyecto/ del DVD anexo.



Organización

- El inventario de equipos (médico, eléctrico, mecánico y planta física) no posee características técnicas que ayuden al personal de mantenimiento a realizar sus actividades eficientemente.
- Los datos de los equipos no están completos o son erróneos. Esto hace que el servicio de mantenimiento no sea brindado de forma eficiente.
- La obtención manual de los costos de la reparación de los equipos es difícil, pues implican diversos cálculos (de materiales, de mano de obra, de transporte, entre otros) que pueden dar lugar a cifras incorrectas o inexactas.

Recurso Humano

- Existe en las Unidades Comunitarias de Salud Familiar (UCSF), falta de personal calificado, falta de recursos y también la falta de conocimiento técnico.
- El irrespeto a las líneas de mando, la falta de conocimiento de la estructura orgánica, y la falta de seguimiento de los manuales de procedimiento, generan dualidad de trabajo, gastos innecesarios, sobrecarga de trabajo a los departamentos de mantenimiento (Regional y Superior).

Estas causas o factores generan procesos de Servicio de Mantenimiento complicados e ineficientes. Así mismo, duplicidad de trabajo, con deficiencia en la atención de solicitudes de mantenimiento, en el control del trabajo de los técnicos y de los costos implicados. En la Figura 9-6 se muestra el Diagrama Causa Efecto.



9.2.1.2 Diagrama Causa-Efecto

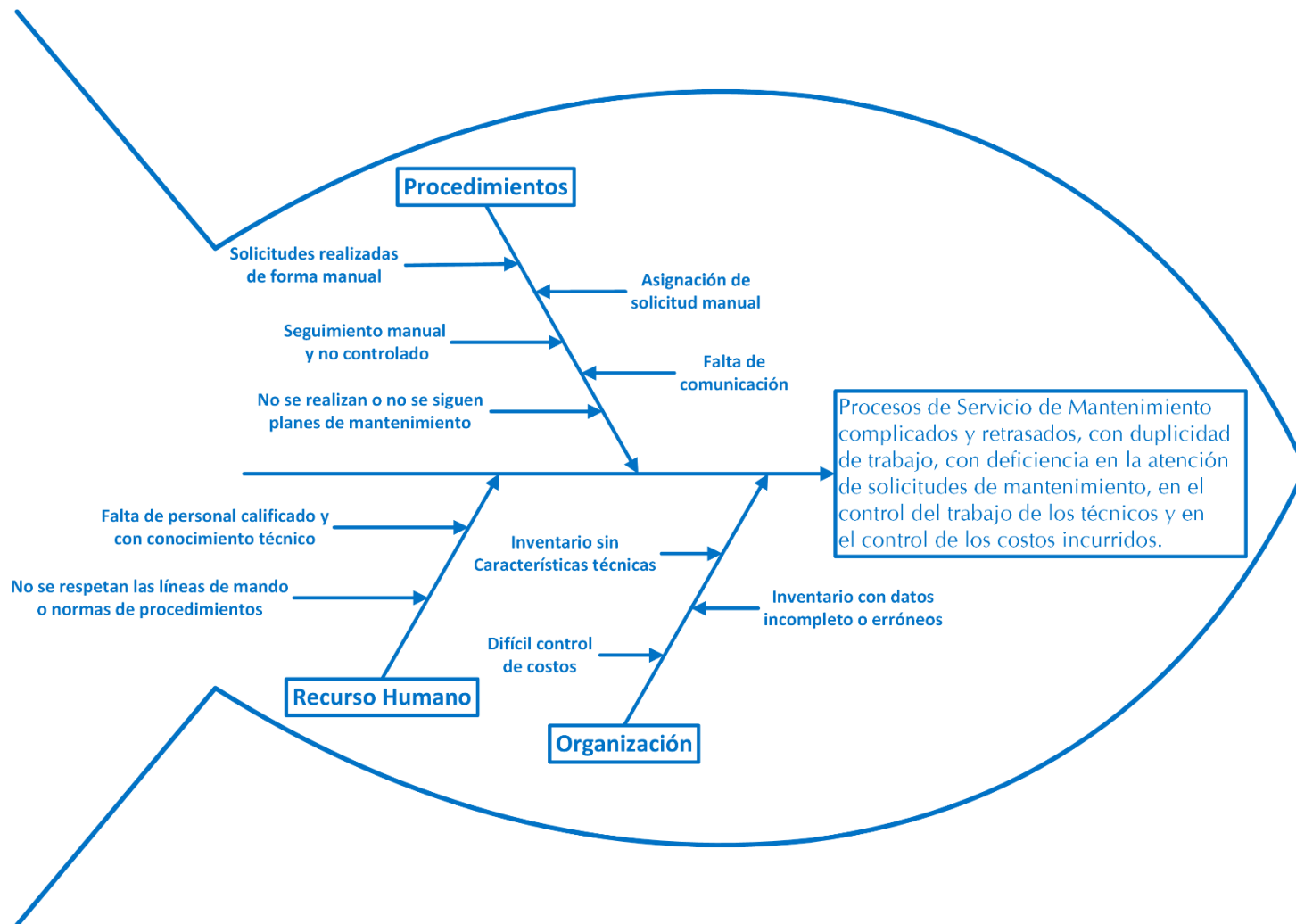


Figura 9-6. Diagrama Causa-Efecto



9.2.1.3 Análisis FODA

A continuación se presentan las Fortalezas, Debilidades, Oportunidades y Amenazas detectadas en el proceso general de servicios de mantenimiento y en las unidades o departamentos involucrados en el mismo a partir del análisis externo e interno (Ver Cuadro 9-1):

Matriz FODA

Fortalezas	Debilidades
<p>F1: Procesos de servicio de mantenimiento bien definidos y amparados mediante manuales de procedimientos, lineamientos técnicos y formatos de documentación.</p> <p>F2: Personal del Área de Mantenimiento General consiente de los beneficios de los sistemas de información, gracias a experiencias pasadas.</p>	<p>D1: La mayoría del personal de mantenimiento en Hospitales y Regionales no está familiarizado con el manejo de aplicaciones y uso del equipo de cómputo.</p> <p>D2: Los medios de comunicación, entre la unidad solicitante de servicios de mantenimiento y la unidad o departamento encargado de brindar este servicio, no son efectivos.</p> <p>D3: El recurso humano es escaso en los departamentos de mantenimiento de los diferentes niveles.</p> <p>D4: No todo el recurso humano tiene la capacidad técnica o especialización necesaria para brindar el servicio de mantenimiento requerido, más bien es empírico en muchos casos.</p> <p>D5: Falta de información técnica en el inventario de equipos a los que se les da mantenimiento. Al mismo tiempo la información que se presenta es errónea o incompleta en algunos casos.</p> <p>D6: La realización manual de la solicitud, expone a errores de especificación de algunos datos necesarios con lo cual no es posible dar un servicio eficaz.</p>
Oportunidades	Amenazas
<p>O1: Se posee equipo tecnológico necesario y a disposición de la unidad de negocios encargada de los sistemas de información.</p> <p>O2: Se tiene el apoyo de diversas dependencias del MINSAL involucradas en la situación y que se muestran a favor y con mucho interés de contribuir a una solución.</p> <p>O3: Existencia de convenios con Instituciones Educativas para formación de profesionales especializados en equipo médico hospitalario.</p>	<p>A1: El servicio de mantenimiento depende en cierta medida de otras unidades o áreas, por ejemplo se requiere de un inventario fiable con la información necesaria; y de la eficiente gestión de los materiales y repuestos necesarios para mantenimientos preventivos y correctivos del diferente equipo médico, reparaciones en planta física y mobiliario, mecánica general y electromecánica.</p> <p>A2: Poca o limitada asignación de recursos económicos.</p>

Cuadro 9-1. Matriz FODA



A través del análisis anterior, se han determinado cuatro conjuntos de alternativas estratégicas de solución, que permitan maximizar las fortalezas, aprovechar las oportunidades, contrarrestar las debilidades y minimizar los riesgos esto se puede observar en el Cuadro 9-2.

Matriz de Estrategias

		Factores Externos	
		Fortaleza F1, F2	Debilidades D1,...,D6
Factores Internos	Oportunidades O1,...,O3	Estrategia FO MAXI-MAXI	Estrategia DO MAXI-MINI
	Fortalecer, optimizar y automatizar los procesos de servicio de mantenimiento, que junto con la experiencia y conocimiento del personal, y el apoyo de la tecnología puedan orientar sus esfuerzos a la rápida atención y eficiente gestión de las solicitudes de servicio de mantenimiento, planificando y realizando los controles pertinentes. Que la respuesta a solicitudes sea oportuna, y que los recursos destinados a esta unidad sean empleados eficientemente, aprovechando o buscando todo lazo o alianza con instituciones que apoyen este proceso.	Concientizar al personal de los beneficios del uso de la tecnología en los procesos de negocio y capacitar en el uso de herramientas o sistemas informáticos, tomando en cuenta a los nuevos profesionales que se esperan con mejor conocimiento técnico. Crear una estructura o plataforma tecnológica, que provea el vínculo o medio de comunicación necesario entre las unidades solicitantes de servicios de mantenimiento y las áreas o departamentos encargados de brindarlos, que garantice la efectividad y eficiencia de dichos servicios; que a la vez provea toda aquella información de interés, útil y fiable para ambas partes; y que permita el seguimiento y control de los procesos para el aprovechamiento del recurso humano disponible.	
	Amenazas A1,A2	Estrategia FA MINI-MAXI	Estrategia DA MINI-MINI
	Lograr la integración y coordinación con las otras unidades o áreas, de las cuales el servicio de mantenimiento depende en cierta medida, garantizando que los elementos que se necesitan de estas unidades sean proveídos adecuadamente y de acuerdo con los manuales y lineamientos establecidos.	Proveer la información necesaria para brindar un servicio de mantenimiento eficiente (como características técnicas de equipos, ubicación, rutinas de mantenimiento correspondientes, entre otros) que ayude a agilizar dicho servicio; y que permita llevar el control y seguimiento de estos (como los planes de mantenimiento preventivo, solicitudes pendientes, entre otros), aprovechando el recurso disponible. Garantizando las necesidades de datos de fuentes externas.	

Cuadro 9-2 Matriz de Estrategias.



9.2.2 Definición del Problema

Un problema es aquella situación que se presenta como una discrepancia entre lo que se tiene y lo que se desea. Es importante tener en mente que, cuando existe un problema, lo que se aprecia son sus efectos y las reacciones que éstos provocan en las personas involucradas, por tanto en primera instancia se han dedicado esfuerzos a investigar las causas que lo originan.

En consecución con la definición anterior de “Problema”, si se define lo que se tiene (los efectos del problema), los factores o elementos que dan origen a este (las causas del problema) y lo que se desea, entonces se tendrá la Definición del Problema.

Efectos del Problema que se tienen actualmente

Un problema se manifiesta mediante un conjunto de efectos. Basado en el Análisis Causa-Efecto de la sección anterior (0) y en las entrevistas de investigación realizadas⁵², se han determinado diferentes efectos:

- Complejidad en la respuesta de las solicitudes de servicio de mantenimiento y la asignación de las ordenes de trabajo a los técnicos.
- No hay control o seguimiento de las órdenes de trabajo asignadas a los técnicos y del avance de los trabajos realizados.
- Desconocimiento del estado de la solicitud de servicio de mantenimiento por parte del solicitante.
- No existe o no se le da seguimiento a la planificación de mantenimiento preventivo para equipos.
- Falta de información técnicas de los diversos equipos.
- Duplicidad de trabajo.
- Calculo engorroso de los costos totales incurridos en las órdenes de trabajo, y en la creación de consolidados de costos.

Así mismo, otro de los efectos o manifestaciones del problema actual es que el servicio de mantenimiento presenta demora. Esto se puede observar en el Anexo 6⁵³, de forma específica en el gráfico de los resultados de la pregunta 7, en la cual el 46% de los encuestados opina que el tiempo de respuesta del servicio de mantenimiento es de muy larga duración y el 38% opina que es de duración intermedia, mientras que solo el 7% opina que es de corta duración y de respuesta inmediata respectivamente.

Causas que originan el problema actual

En el Análisis Causa-Efecto de la sección anterior (0) se han identificado las causas que inciden en el problema, y se pueden observar gráficamente en la Figura 9-6.

⁵² Para más información ver Anexo 5 – Entrevistas del Documento de Anteproyecto que se encuentra dentro del directorio /Documentos/1 – Anteproyecto/ del DVD anexo.

⁵³ Anexo 6 – Encuestas sobre servicios de mantenimiento de equipos del Documento de Anteproyecto que se encuentra dentro del directorio /Documentos/1 – Anteproyecto/ del DVD anexo.



Situación deseada

El Área de Mantenimiento General, tiene por objetivo planificar, coordinar y controlar el mantenimiento de los equipos, instalaciones e infraestructura de las dependencias del Nivel Superior y apoyar a los Niveles Regional y Local. Y entre las funciones que el Área de Mantenimiento General debe realizar, destacan las siguientes:

- Proporcionar asistencia técnica especializada a los establecimientos de salud del Nivel Regional y Local.
- Elaborar y supervisar programas de mantenimiento preventivo para el Nivel Superior y proporcionar apoyo en los Niveles Regional y Local.
- Dar seguimiento a los programas de mantenimiento preventivo de los Niveles Regional y Local.
- Asegurar el buen funcionamiento de los diferentes equipos (Médicos, Mecánicos, Electromecánicos, Planta Física y Mobiliario) a través de las Secciones respectivas que se encargan de estos equipos, mediante la prestación y ejecución de los servicios de mantenimiento preventivo y correctivo.

Esto entonces compone parte de la situación que se desea, cuyo fin es la gestión y provisión de servicios de conservación y mantenimiento, en forma oportuna y eficiente a los establecimientos del MINSAL, asegurando así la disponibilidad de estos recursos para ofrecer servicios de salud con calidad a la población.

Para definir completamente el problema, en la sección siguiente (9.2.3) se realiza la formulación de este.

9.2.3 Formulación del Problema

La formulación del problema se realiza mediante el método de la caja Negra (Ver Figura 9-7) en donde el Estado A representa la situación actual que es indeseada y el Estado B la situación que se espera o se desea, la cual se describe a continuación y está basada en las aseveraciones realizadas en la sección anterior (9.2.2):

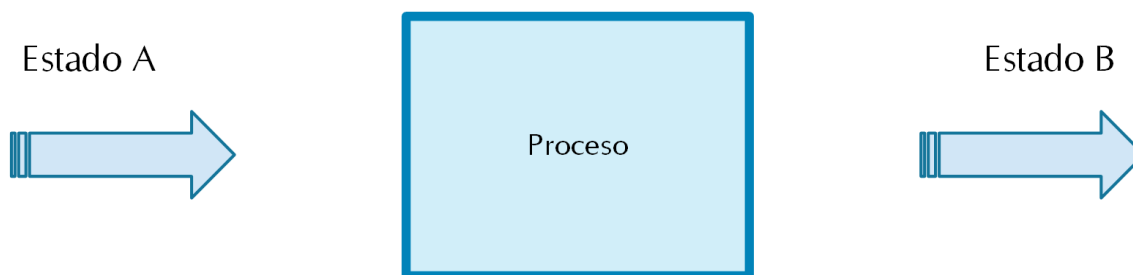


Figura 9-7. Formulación del Problema mediante el Diagrama de Caja Negra

El tiempo de respuesta de los procesos de servicios de mantenimiento es de muy larga duración, así mismo presenta duplicidad de trabajo, ineficiencia en la atención de solicitudes de mantenimiento, en el control del trabajo de los técnicos y en el registro y cálculo de los costos en los cuales se ha incurrido. Hay una mala comunicación entre las unidades solicitantes de servicio y los departamentos de mantenimiento. No existe o no se le da seguimiento a los planes de mantenimiento preventivo.

Tener procesos de servicio de mantenimiento eficientes, con rápida atención a las solicitudes de mantenimiento, con una comunicación efectiva por medio de información útil y oportuna. Planificación y seguimiento de las diferentes actividades y personal que involucran los procesos de servicios de mantenimiento.

9.3 Solución Propuesta

De forma general, la solución es la respuesta a un problema o situación indeseada. La situación que se desea en el Área de Mantenimiento General, los departamentos de mantenimiento de Nivel Regional y Local y las dependencias a las que se les brinda apoyo, es contar con procesos de servicio de mantenimiento eficientes, con rápida atención a las solicitudes de mantenimiento, con una comunicación efectiva por medio de información útil y oportuna; y la planificación y seguimiento de las diferentes actividades y personal que involucran los procesos de servicios de mantenimiento. Para llegar a esta situación deseada o Estado B del Diagrama de Caja Negra (Figura 9-7) presentado en la sección anterior, y tomando en cuenta las alternativas estratégicas de solución del Cuadro 9-2, se presenta la solución propuesta mediante un Diagrama de Enfoque de Sistemas en la Figura 9-9.

Para una mejor comprensión de la solución propuesta, en la Figura 9-8 se puede observar gráficamente la solución, el Proceso de Servicio de Mantenimiento desarrollado mediante el Sistema Informático de Mantenimiento (en adelante **SIM**), en ella se describe gráficamente cómo será la comunicación, entre las diferentes dependencias del MINSAL con el Área y los Departamentos de Mantenimiento, así como los diferentes actores encargados de realizar ciertas actividades necesarias para poder realizar el servicio de mantenimiento.



Descripción de la Figura 9-8

Los Jefes pueden ser de un Departamento, Área o Servicio que se encuentra en un determinado establecimiento o dependencia del Ministerio de Salud, entre los cuales están los Hospitales de Segundo y Tercer Nivel y las UCSF, sin embargo, la solución no se limita sólo a las unidades que brindan directamente los Servicios de Salud sino que incluye las Áreas o Departamentos de apoyo, es decir, de índole Administrativa que a la vez necesitan de Servicios de Mantenimiento.

Los Jefes de Mantenimiento, al igual que los Técnicos de Mantenimiento pueden pertenecer al Área de Mantenimiento General, Departamento de Mantenimiento Regional o Departamento de Mantenimiento Local.

Las personas mencionadas anteriormente serán usuarios del Sistema Informático y podrán interactuar con el mismo a través de un cliente (una computadora con acceso a internet) mediante el uso de un navegador web (por ejemplo, Mozilla Firefox), el cual presentara una interfaz gráfica que permita el ingreso de datos, selección de opciones, visualización de reportes, entre otros. Se refleja la forma en que los usuarios interactúan con el Sistema, realizando las acciones permitidas, representadas por flechas desde y hacia la interfaz gráfica que utiliza (Portal Web).

La comunicación del cliente, será con un Servidor Web que se encargará de responder a las peticiones realizadas por los usuarios, por ejemplo: recepción de datos ingresados por medio de formularios, envío de datos para ser presentados en pantalla, generación de reportes, realización de cálculos y validaciones, y la comunicación con el Servidor de Base de Datos, el cual permitirá el almacenamiento, modificación y extracción de datos de forma ordenada y controlada. Todas estas acciones dedicadas a un objetivo específico, permitirán cumplir con los requerimientos del usuario, es decir, lo que el Sistema Informático “deberá hacer” para apoyar los Procesos de Servicio de Mantenimiento y que soporta la lógica de negocios de la organización. La descripción del equipo donde funcionara la solución se hará en la Etapa de Determinación y Análisis de Requerimientos.

El Sistema Informático de Mantenimiento tendrá dos fuentes importantes de datos, que son el Sistema Informático de Activo Fijo del MINSAL (SIAF) y el Sistema Informático Nacional de Abastecimiento (SINAB), con los cuales tendrá que integrarse debido a que:

El SIAF: Es la herramienta de control administrativo, para cuantificar y clasificar los bienes que posee el MINSAL, y este se apega a los Lineamientos Técnicos para la Administración de Activo Fijo (descritos en la sección 7.3.4 de este documento). De aquí se obtendrá el código oficial de Activo Fijo de los diferentes Equipos que incorpore el Sistema Informático de Mantenimiento y algunas características básicas.

El SINAB: Es la herramienta para el registro y manejo de los bienes en los almacenes del MINSAL, en cuanto a recepción, almacenamiento, distribución y control de existencias de los suministros. De aquí se obtendrá la información de los vales de salida y los costos incurridos en materiales y repuestos, ya que según el Capítulo VI, numeral 2 inciso b, de los Lineamientos de Técnicos para el Manejo de Suministros en Almacenes del MINSAL, se establece que “En la preparación de los pedidos, deben emitirse oportunamente los vales de salida, con los campos del SINAB y otros sistemas informáticos autorizados por la Dirección de Tecnologías de Información y Comunicaciones (DTIC), completados correctamente”.



9.3.1 Propuesta del Proceso de Servicio de Mantenimiento para el SIM.

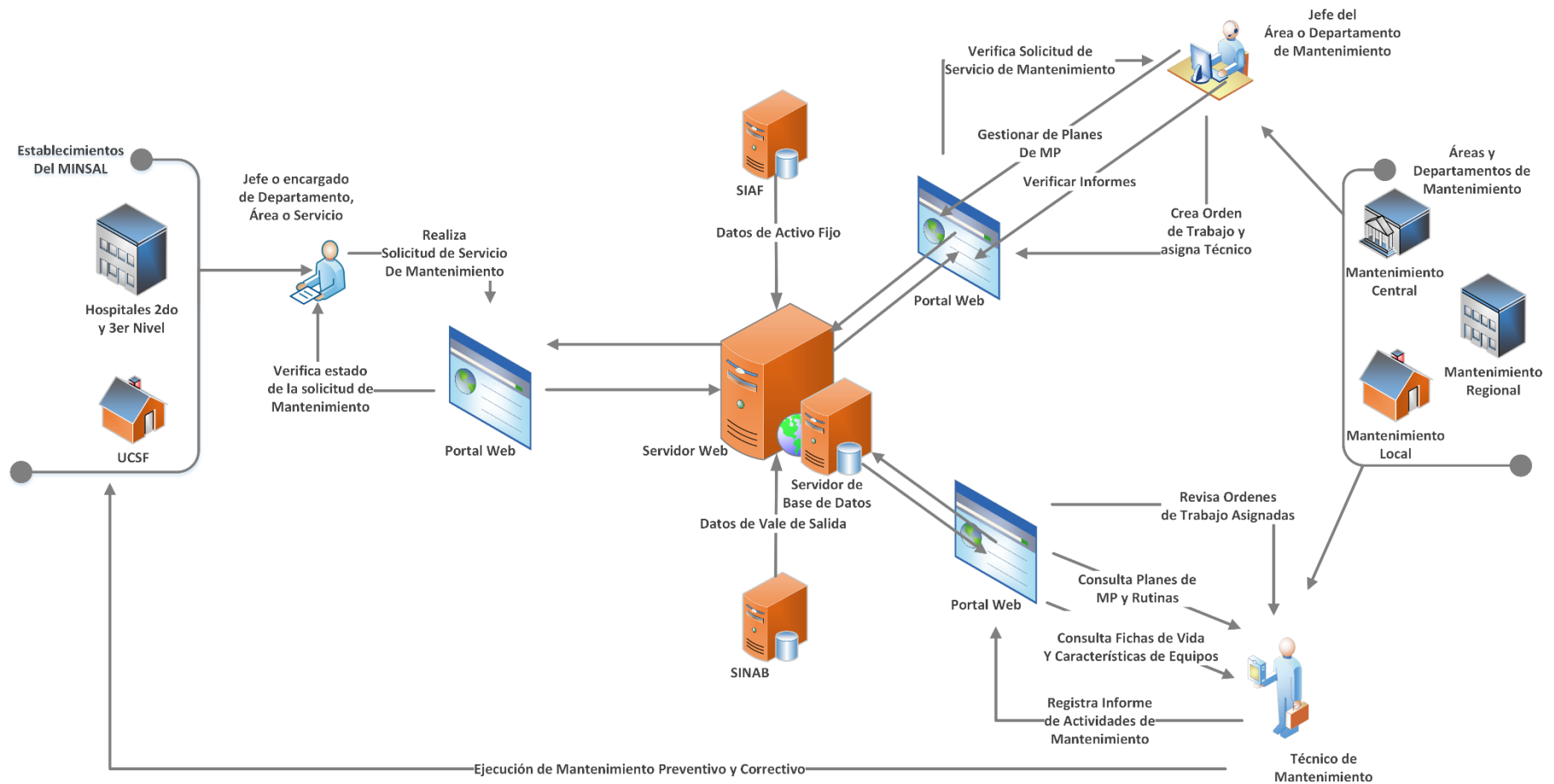


Figura 9-8 Propuesta del Proceso de Servicio de Mantenimiento para el SIM



9.3.2 Enfoque de Sistemas de la Solución Propuesta

La Figura 9-9 muestra el Diagrama del Enfoque de Sistemas para la solución propuesta del Área de Mantenimiento General y los Departamentos de Mantenimiento Regional y Local.

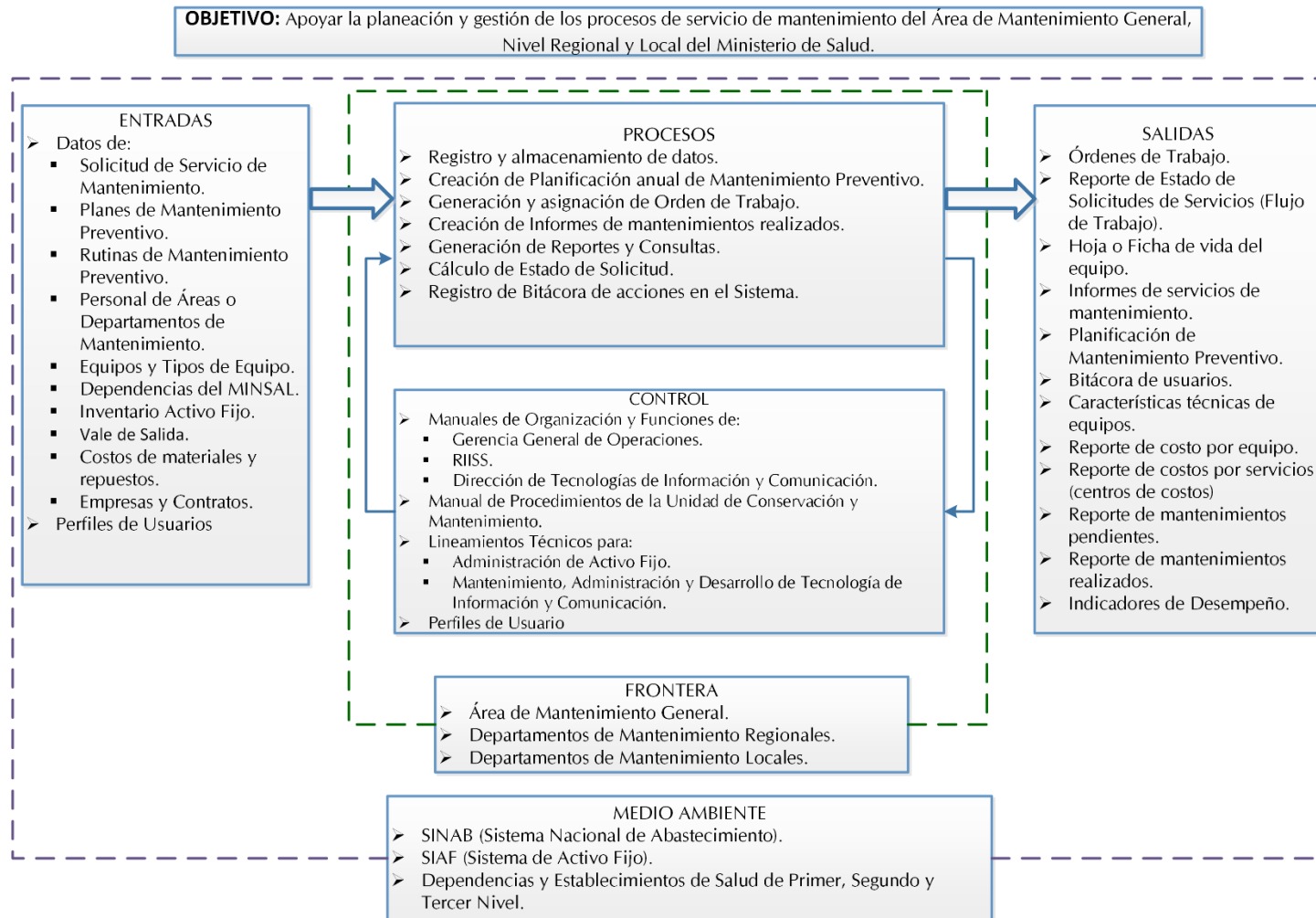


Figura 9-9. Diagrama de Enfoque de Sistemas de la Solución Propuesta.



El Diagrama de Enfoque de Sistemas de la Solución Propuesta presentado en la Figura 9-9, es una representación gráfica del comportamiento del sistema propuesto referente a los servicios de mantenimiento, a continuación se describen cada uno de sus elementos:

- a) **Objetivo del Sistema:** Apoyar la planeación y gestión del proceso de mantenimiento del Área de Mantenimiento General, Nivel Regional y Local del Ministerio de Salud.
- b) **Salidas:** Conjunto de elementos generados al realizar los procesos de mantenimiento por los diferentes Áreas o Departamentos de los Niveles Superior, Regional y Local.
- Órdenes de Trabajo.
 - Reporte de Estado de Solicitudes de Servicio.
 - Hoja o ficha de vida del equipo.
 - Informes de Servicios de Mantenimiento.
 - Planificación de Mantenimiento Preventivo.
 - Bitácora de Usuarios.
 - Características técnicas de equipos.
 - Reporte de Costo por equipo.
 - Reporte de costo por servicios (centros de costo).
 - Reporte de mantenimientos pendientes.
 - Reporte de mantenimientos realizados.
- c) **Entradas:** Insumos necesarios por el sistema para la ejecución de los procesos de mantenimiento entre los cuales se tiene:
Datos acerca de:
- Solicitud de Servicio de Mantenimiento.
 - Planes de Mantenimiento Preventivo.
 - Rutinas de Mantenimiento Preventivo.
 - Personal de Áreas o Departamentos de Mantenimiento.
 - Equipos y tipos de equipos.
 - Dependencias del MINSAL.
 - Inventario de Activo Fijo.
 - Vale de Salida.
 - Costo de materiales y repuestos.
 - Empresas y Contratos.
 - Perfiles de Usuarios.
- d) **Procesos:** Conjunto de actividades que son ejecutadas para poder transformar las entradas en salidas.
- Registro y almacenamiento de datos.
 - Creación de Planificación anual de Mantenimiento Preventivo.
 - Generación y asignación de Órdenes de Trabajo.
 - Creación de Informes de mantenimientos realizados.
 - Generación de Reportes y Consultas.



- Cálculo de Estado de Solicitud.
 - Registro de Bitácora de acciones en el Sistema.
- e) **Control:** Manuales, Lineamientos y otros elementos que contribuyen en el establecimiento de normas, reglas que debe de cumplir el sistema.
- Manuales de Organización y Funciones de:
 - Gerencia General de Operaciones.
 - RIIS
 - Dirección de Tecnología de Información y Comunicación.
 - Manual de Procedimientos de la Unidad de Conservación y Mantenimiento.
 - Lineamientos técnicos para:
 - Administración de Activo Fijo.
 - Mantenimiento, Administración y Desarrollo de Tecnologías de Información y Comunicación.
 - Perfiles de Usuarios.
- f) **Frontera.**
- Área de Mantenimiento General.
 - Departamentos de Mantenimiento Regional.
 - Departamentos de Mantenimientos Local.
- g) **Medio Ambiente:** Sistemas y dependencias externas con las cuales el sistema se relacionará, y brindarán entradas (insumos) al sistema.
- SINAB (Sistema Nacional de Abastecimiento).
 - SIAF (Sistema de Activo Fijo).
 - Dependencias y Establecimientos de Salud de Primer, Segundo y Tercer Nivel.



10 Capítulo IV

Especificación de Requerimientos (SRS)

La Especificación de Requerimientos de Software (en adelante SRS, por sus siglas en inglés), es una descripción completa del comportamiento del sistema que se va a desarrollar. Para realizar la Especificación de Requerimientos de Software se tomarán como base las prácticas recomendadas en el estándar IEEE 830-1998, Recommended Practice for Software Requirements Specifications.

Este capítulo presenta la implementación del Estándar IEEE 830 -1998 para la Especificación de Requerimientos del Sistema SIM, dentro del cual se especifica una descripción general que brinda a los usuarios de negocio, así como al personal técnico encargado del desarrollo, una perspectiva del producto que permite conocer las funciones que el Sistema brindará y las restricciones que se tendrá en el desarrollo del mismo.

Además de la perspectiva del producto, se presenta la especificación de los requerimientos categorizados en interfaces externas, requerimientos funcionales y otros tipos de requerimientos.

Las interfaces externas presentan un conjunto de requerimientos no funcionales que son necesarios tomar en cuenta en el desarrollo del Sistema, así también se tienen los requerimientos funcionales los cuales son plasmados por medio de Casos de Uso y Diagramas de Secuencias del Sistema.

La Especificación de Requerimientos de Software es el resultado de la Etapa de Análisis y Determinación de Requerimientos del Ciclo de Vida de Desarrollo de Software.



10.1 Introducción.

10.1.1 Propósito

El presente documento de SRS tiene por propósito especificar y clasificar de manera formal y oficial, los requerimientos funcionales, interfaces, restricciones de diseño y desarrollo del Sistema Informático que se ha de construir en conjunto con la Dirección de Tecnologías de Información y Comunicaciones, Área de Mantenimiento General del MINSAL y el Equipo de Trabajo de Graduación de la Escuela de Ingeniería de Sistemas Informáticos de la Universidad de El Salvador. Este documento ha sido elaborado en base al estándar del **IEEE 830-1998**⁵⁴, que define los lineamientos para la elaboración del documento de requerimientos de software. El documento de Especificación de Requerimientos está sujeto a revisiones periódicas por el equipo de trabajo, y las instituciones anteriormente mencionadas, generando nuevas versiones del documento hasta su aprobación, una vez aprobado, servirá de base para las etapas siguientes del desarrollo del Sistema Informático.

10.1.2 Definiciones, Acrónimos y Abreviaturas.

Las definiciones, acrónimos y abreviaturas son descripciones que permiten exponer de manera clara y exacta el significado de un término, con el fin de brindar al lector un conocimiento previo de los mismos, para una mejor comprensión del documento. Esta subsección se ha dividido en tres apartados: Definiciones, Acrónimos y Abreviaturas Generales, Definiciones Técnicas y Definiciones de Elementos de Casos de Uso.

10.1.2.1 Definiciones de Elementos de Casos de Uso

Termino	Definición
Actor	Representa a un usuario desempeñando un determinado papel (Rol), una organización o un sistema externo que interactúa con el sistema.
Caso de Uso	Es una descripción de los pasos o las actividades que deberán realizarse para llevar a cabo algún proceso o tarea por algún actor.
Descripción	Brinda una descripción general del caso de uso en cuestión.
Flujos Alternos	Son bifurcaciones del flujo básico (escenario principal de éxito), que representan la aparición de errores o excepciones y el manejo de estas, durante la ejecución de un caso de uso.
Flujo Básico	Las interacciones entre los actores y el sistema que son necesarios para lograr el objetivo y que satisface los intereses del personal involucrado, es decir, los pasos para realizar el caso de uso de forma exitosa (escenario principal de éxito).

⁵⁴ Institute of Electrical and Electronics Engineers. (1998). *IEEE Std 830-1998*. New York: IEEE-SA Standards Board.



Termino	Definición
Frecuencia Esperada	Número esperado de repeticiones expresado en período de tiempo, con la que se realizará el caso de uso en cuestión.
Importancia	Grado de trascendencia del caso de uso en el sistema. (Vital, Alta, Moderada o Baja)
Post-condiciones	Las post-condiciones (garantías de éxito) establecen qué debe cumplirse cuando el caso de uso se completa con éxito.
Precondiciones	Condiciones que deben ser verdaderas para que el caso de uso se realice con éxito. Las precondiciones no se prueban en el caso de uso, sino que son condiciones que se asume que son verdaderas.
Propósito	Es la meta u objetivo que persigue la realización de determinado caso de uso, que generalmente es un objetivo de negocio.
Referencia Cruzada	Indica los requerimientos funcionales con los que está relacionado o satisface el caso de uso.
Urgencia	Grado de necesidad de la elaboración del caso de uso dentro del desarrollo del sistema. (Alta, Moderada o Baja).

Cuadro 10-1. Definición de Elementos de la Descripción de Casos de Uso.

10.2 Resumen del Documento.

La segunda sección de este documento, Descripción General (ver apartado 10.3), brinda una descripción de la funcionabilidad del Sistema SIM, en la que se describe de forma general, cual es la perspectiva del producto a desarrollar, la funcionalidad que presentará, las características de los usuarios involucrados y todas aquellas restricciones que intervienen en el desarrollo del sistema. Esto presenta un contexto o marco de referencia para la especificación de requerimientos del sistema.

La tercera sección de este documento, Especificación de Requerimientos (ver apartado 10.4), describe en términos técnicos y con mayor profundidad de detalle los requerimientos del sistema.



10.3 Descripción General

10.3.1 Perspectiva del Producto.

La Perspectiva del Producto, consiste en presentar una visión y análisis del Producto a desarrollar, que permita al lector comprender los elementos de la Lógica del Negocio y su relación con los elementos del Producto, es decir, cómo el Producto da soporte a la Lógica del Negocio. A continuación se presenta la perspectiva del producto del SIM.

El SIM será un sistema multiplataforma que funcionará en un ambiente web, por lo tanto, los usuarios podrán acceder al sistema por medio de un navegador web que les permitirá interactuar con el sistema, desarrollado bajo el Modelo de Tres Capas, el cual se describe a continuación:

- Capa de presentación: Capa con la que el usuario interactúa, denominada también capa de usuario, debido a que presenta al usuario la información solicitada y captura los datos brindados por éste. Esta capa se comunica únicamente con la capa de negocio.
- Capa de negocio: Se denomina también Capa de Lógica del Negocio porque es aquí donde se establecen todas las reglas del negocio que deben cumplirse. Esta Capa se comunica con la Capa de Presentación, para recibir las solicitudes o peticiones de usuario y presentar los resultados o respuestas a dichas solicitudes, y con la Capa de Datos, para solicitar al gestor de base de datos almacenar o recuperar datos de él.
- Capa de datos: Es donde residen los datos, recibe solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

Según el Manual de Organización y Funciones de la DTIC, la función principal de la Unidad de Sistemas de Información, es “Construir los módulos que conforman el Sistema Único de Información en Salud (SUIS) e integrar los diferentes sistemas de forma armónica y alineada a los objetivos institucionales”. Por tanto, el SIM será uno de los sistemas que será parte del SUIS, y como tal, deberá integrarse con otros Sistemas Informáticos, obteniendo de estos datos útiles y necesarios para su funcionamiento. En la Figura 10-1 puede observarse un bosquejo de la interacción del SIM con los sistemas SIAF y SINAB.

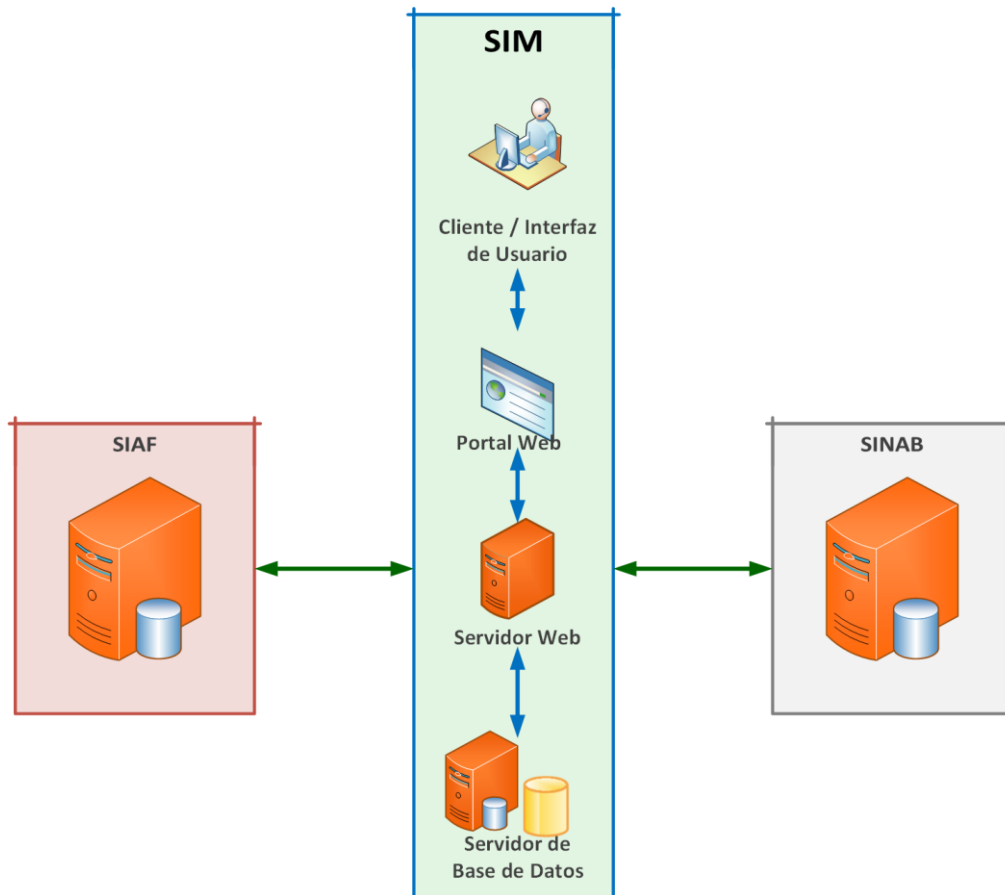


Figura 10-1. Interacción del SIM.

Los Sistemas con los que el SIM se integrará:

SIAF: Es un sistema de control administrativo, para cuantificar y clasificar los bienes que posee el MINSAL, basado en los Lineamientos Técnicos para la Administración de Activo Fijo. De aquí se obtendrá los datos del Activo Fijo requeridos por el Sistema Informático de Mantenimiento.

SINAB: Es un sistema para el registro y manejo de los bienes ingresados en los almacenes del MINSAL, en lo relacionado a recepción, almacenamiento, distribución y control de existencias de los suministros. De este sistema se obtendrá la información de los costos de materiales y repuestos incurridos en la realización de los servicios de mantenimiento.

En la Figura 9-8 se describe la funcionabilidad del sistema, su relación con los sistemas externos mencionados anteriormente y como cada uno de los usuarios descritos en el apartado 10.3.2 interactúa con el sistema.



10.3.1.1 Funciones del Producto.

Las Funciones del Producto, son el conjunto de características que permiten la realización de tareas que satisfacen las necesidades de los usuarios. El SIM es desarrollado con el objetivo de cubrir y dar soporte a ciertas necesidades de los usuarios en los procesos de mantenimiento, por lo cual brindará las siguientes funcionalidades:

- ✓ **Administración de roles y usuarios:** La aplicación tendrá la funcionalidad de crear, modificar, consultar y eliminar usuarios, así también, de cada uno de los roles definidos para cada usuario. Esta funcionalidad permite el otorgamiento y restricción de accesos, definiendo que es lo que puede realizar cada usuario en el sistema, formando así parte de la seguridad del sistema.
- ✓ **Administración de elementos del sistema:** Se conoce como elementos del sistema a cada uno de los elementos involucrados en los procesos de negocio, ya sea directa o indirectamente. Los elementos pueden ser: establecimientos del MINSAL, empresas, empleados, contratos, equipo médico, etc. El sistema proveerá de las interfaces de usuario necesarias para la gestión de cada uno de los elementos.
- ✓ **Gestión de solicitudes de servicio de mantenimiento:** Su principal función es la creación y seguimiento de las solicitudes de mantenimiento, es decir, mostrar al usuario la fase y estado en que se encuentra la solicitud realizada, así como también la consulta, cancelación, transferencia de esta, lo que ayudará al área de mantenimiento a una mejor organización de su trabajo, permitiendo con esto, dar una respuesta más rápida.
- ✓ **Gestión de Ordenes de trabajo:** Tiene como función la creación, iniciación, finalización, consulta, reasignación y cancelación de las órdenes de trabajo, las cuales junto con los informes asociados a estas, sirven de fuente de dato para la creación de las fichas de vida del inventario técnico al cual se le da mantenimiento.
- ✓ **Plan de Mantenimiento Preventivo:** El sistema brinda la función de realizar una programación anual de los mantenimientos preventivos del equipo que lo requiera, lo que permitirá la asignación del personal responsable de la ejecución del plan de mantenimiento preventivo, las rutinas de mantenimiento a realizar y los periodos de tiempo en que se han de realizar los mantenimientos.
- ✓ **Generación de Reportes:** El sistema permitirá la generación de reportes necesarios para los procedimientos administrativos que son realizados en el área de mantenimiento. También permitirá la impresión de: solicitudes de servicio de mantenimiento, órdenes de trabajo, informes de órdenes de trabajo, ficha de vida de equipos, datos técnicos de equipos, etc.



10.3.2 Características de los Usuarios.

10.3.2.1 Clasificación de Usuarios.

La clasificación, es la categorización en determinados grupos de las diferentes personas o actores que intervienen en el Sistema, lo que permite una mejor comprensión de las diferentes responsabilidades (roles) que tendrá cada usuario dentro del sistema.

Es importante tener clara la diferencia entre un Usuario y un Actor. Un Actor es un Rol o Grupo de Roles, mientras que un Usuario, es una persona que dentro del sistema asume un Rol o pertenece a un Grupo de Roles, esto significa que, un Usuario puede acceder al sistema realizando el papel de distintos Actores.

Los Usuarios entonces se pueden clasificar en Actores. Dicha clasificación se puede apreciar en el Cuadro 10-2:

Código	Actor	Descripción
adminSist	Administrador General del Sistema	Será el encargado de administrar los diferentes elementos del Sistema tales como; usuarios, roles, privilegios, dependencias, servicios (ambientes), departamentos de mantenimiento, y personal de cualquier dependencia del Ministerio de Salud, así como también tendrá acceso a la bitácora del sistema.
adminEstb	Administrador de Establecimiento	Usuario que forma parte del área o departamento de mantenimiento, puede ser un jefe de mantenimiento o un técnico. Este usuario será el encargado de realizar la gestión de las empresas que brindarán servicios de mantenimiento, modelo de bien, la administración del bien, gestión del personal, contratos, servicios y departamentos de mantenimiento, este usuario será de apoyo a las tareas que realiza el jefe de mantenimiento.
jefeManto	Jefe de Mantenimiento	Usuario que es jefe del área o departamento de mantenimiento, tiene a su cargo, personal técnico para la realización de servicios de mantenimiento correctivo y preventivo, tiene a su cargo un conjunto de dependencias a las que les brinda los servicios antes mencionados (Nivel Superior y Regional). Este usuario podrá realizar la gestión de planes anuales de mantenimiento preventivo, períodos de rutinas, órdenes de trabajo, administración del bien, y coordinar las tareas administrativas relacionadas con los servicios de mantenimiento.



Código	Actor	Descripción
tecnManto	Técnico de Mantenimiento	Usuario que forma parte del área o departamento de mantenimiento, que tiene a su cargo un conjunto de equipos, instalaciones e infraestructuras que pertenecen a las áreas o sub-unidades, de la cual es responsable. Este usuario podrá realizar la consulta de los planes de mantenimiento que le han sido asignados, rutinas de mantenimiento, órdenes de trabajo y características de los equipos, además de poder gestionar los informes de las órdenes de trabajo.
jefeUni	Jefe de Unidad	Usuario que tiene a su cargo la jefatura de un departamento o área médica, administrativa o de apoyo, tendrá la capacidad de realizar solicitudes de mantenimiento de cualquier coordinación de las sub-unidades bajo su cargo, así como también de aprobar solicitudes de mantenimiento que las sub-unidades hayan realizado, además de consultar el estado de las solicitudes realizadas.
solicit	Solicitante	Usuario que es jefe de una sub-unidad o ambiente, tiene a su cargo equipos, instalaciones e infraestructuras pertenecientes a la sub-unidad o ambiente. Este usuario podrá realizar solicitudes de mantenimiento de los recursos bajo su cargo, para lo cual necesitan la aprobación del jefe superior del departamento o área al que pertenece (Jefe de Unidad), además podrá consultar el estado de la solicitud.

Cuadro 10-2 Clasificación de Usuarios en Actores.

10.3.2.2 Perfil de los Usuarios.

Las siguientes tablas (Tabla 10-1, Tabla 10-2, Tabla 10-3, Tabla 10-4, Tabla 10-5 y Tabla 10-6) presentan un conjunto de características ideales que los usuarios deben poseer, para asumir y desempeñar de la mejor manera un determinado Rol (Actor) dentro del sistema:



Perfil - Administrador General del Sistema.

Usuario/Actor:	Administrador General del Sistema.	Código:	AdminSist
Nivel Académico			
Ingeniero de Sistemas Informáticos, Computación o Áreas afines.			
Conocimientos			
<ul style="list-style-type: none"> • Conocimientos en Administración de Sistemas Informáticos. • Conocimientos en Sistemas Webs basados en el Framework Symfony2. • Manejo del Framework Sonata para la administración y la seguridad del sistema. • Conocimientos de los Gestores de Base de Datos PostgreSQL y SQL Server 2008 R2. • Manejo del Sistema Operativo Linux, Distribución Debian 7.0 Wheezy. 			
Experiencia			
Dos años en actividades de Administración y Desarrollo de Sistemas Informáticos.			
Cargo Sugerido			
Analista – Programador de la DTIC – MINSAL.			

Tabla 10-1 Perfil de Administrador General del Sistema.

Perfil - Administrador del Establecimiento.

Usuario/Actor:	Administrador del Establecimiento.	Código:	adminEstb
Nivel Académico			
Ingeniero o Técnico Mecánico, Eléctrico o Industrial.			
Conocimientos			
<ul style="list-style-type: none"> • Conocimientos de Biomédica. • Conocimientos de Equipo Médico Hospitalario. • Conocimientos de Administración General. • Conocimientos Medios en el manejo de computadoras. • Conocimientos en el manejo del Sistema Operativo Linux, en cualquiera de sus distribuciones, Debian, Ubuntu, Linux Mint, etc. Sistema Operativo Windows en sus versiones XP, 7, 8. 			
Experiencia			
<ul style="list-style-type: none"> • Dos años en labores de mantenimiento, de preferencia hospitalario y administración general con énfasis en recursos humanos. • Experiencia en el uso de sistemas web. 			
Cargo Sugerido			
Jefe de Área o Departamento de Mantenimiento del Nivel Superior, Regional o Local del MINSAL.			

Tabla 10-2 Perfil del Administrador del Establecimiento.



Perfil - Jefe de Mantenimiento.

Usuario/Actor:	Jefe de Mantenimiento.	Código:	jefeMantto
Nivel Académico			
Ingeniero o Técnico Mecánico, Eléctrico o Industrial.			
Conocimientos			
<ul style="list-style-type: none"> • Conocimientos de Biomédica. • Conocimientos de Equipo Médico Hospitalario. • Conocimientos de Administración General. • Conocimientos Medios en el manejo de computadoras. • Conocimientos en el manejo del Sistema Operativo Linux, en cualquiera de sus distribuciones, Debian, Ubuntu, Linux Mint, etc. Sistema Operativo Windows en sus versiones XP, 7, 8. 			
Experiencia			
<ul style="list-style-type: none"> • Dos años en labores de mantenimiento, de preferencia hospitalario y administración general con énfasis en recursos humanos. • Experiencia en el uso de sistemas web. 			
Cargo Sugerido			
Jefe de Área o Departamento de Mantenimiento del Nivel Superior, Regional o Local del MINSAL.			

Tabla 10-3 Perfil del Jefe de Mantenimiento.

Técnico de Mantenimiento

Usuario/Actor:	Técnico de Mantenimiento	Código:	tecMantto
Nivel Académico			
Técnico en Mecánica, Electricidad, Electrónica, Refrigeración y Aire Acondicionado.			
Conocimientos			
<ul style="list-style-type: none"> • Conocimientos en electricidad básica y avanzada, maquinaria de vapor, instalaciones eléctricas y potencia, aire acondicionado, sistemas de control automático, reparación y mantenimiento de cocinas, lavadoras, esterilizadores y calderas, mantenimiento de infraestructura hospitalaria, operación y mantenimiento de equipo de laboratorio, mecánica avanzada, tecnología biomédica, equipo odontológico, equipo de rayos X, plantas de emergencia, conocimientos de albañilería y fontanería. • Conocimientos básicos de computadoras. • Conocimientos básicos en el Manejo de Sistemas Operativos Linux en sus diferentes Distribuciones: Debian, Ubuntu, Linux Mint, etc. Sistema Operativo Windows en sus versiones XP, 7, 8. 			
Experiencia			
<ul style="list-style-type: none"> • Dos años en labores de mantenimiento de preferencia hospitalario. 			
Cargo Sugerido			
Técnico del Área o Departamento de Mantenimiento del Nivel Superior, Regional o Local del MINSAL.			

Tabla 10-4 Perfil del Técnico de Mantenimiento.



Perfil - Jefe de Unidad

Usuario/Actor:	Jefe de Unidad.	Código:	jefeUni
Nivel Académico			
Doctorado en Medicina. El nivel académico requerido para el Jefe de Unidad queda establecido según requerimientos del Área de Recursos Humanos de cada establecimiento del MINSAL.			
Conocimientos			
<ul style="list-style-type: none"> • Conocimientos básicos en el Manejo de Sistemas Operativos Linux en sus diferentes Distribuciones: Debian, Ubuntu, Linux Mint, etc. Sistema Operativo Windows en sus versiones XP, 7, 8. 			
Experiencia			
La experiencia requerida puede variar, según los requerimientos del Área de Recursos Humanos de cada establecimiento del MINSAL.			
Cargo Sugerido			
Jefes y Directores de Establecimientos, Unidades, Departamentos, Áreas.			

Tabla 10-5 Perfil del Jefe de Unidad.

Perfil - Solicitante

Usuario/Actor:	Solicitante	Código:	Solict
Nivel Académico			
El nivel académico del solicitante puede variar según el nivel académico del personal de cada establecimiento del MINSAL el cual tendrá acceso al sistema informático.			
Conocimientos			
<ul style="list-style-type: none"> • Conocimientos básicos de computadoras. • Conocimientos básicos en el Manejo de Sistemas Operativos Linux en sus diferentes Distribuciones: Debian, Ubuntu, Linux Mint, etc. Sistema Operativo Windows en sus versiones XP, 7, 8. 			
Experiencia			
No existe experiencia requerida para este Usuario/Actor.			
Cargo Sugerido			
No existe cargo sugerido, este puede variar según el personal de los establecimientos del MINSAL que tendrán acceso al sistema informático.			

Tabla 10-6 Perfil del Solicitante.



10.3.3 Restricciones.

Las restricciones son un conjunto de limitaciones basadas en políticas organizacionales, lineamientos técnicos, estándares establecidos, etc., que rigen o sirven de guía en el desarrollo de un Sistema. A continuación se presentan las restricciones o consideraciones que el Equipo de Trabajo debe tener en cuenta para el desarrollo de proyecto, las cuales han sido dadas por la DTIC.

10.3.3.1 Políticas Regulatoras.

Las Políticas Regulatoras son el conjunto de directrices que sirven para normalizar la realización de ciertas tareas o actividades. Las siguientes políticas están basadas en los lineamientos técnicos que la DTIC posee para el desarrollo y mantenimiento de Sistemas de Información, las cuales están orientadas, en su mayoría, a la utilización sistemas operativos, software y herramientas de licenciamiento libre y a los estándares que ya se utilizan en su ambiente laboral de desarrollo y producción.

Uso de Software Bajo Licenciamiento Libre.

El software requerido para el desarrollo del sistema y su posterior puesta en marcha, es necesario que se encuentre bajo licenciamiento libre y gratuito, exceptuándose el caso de sistemas de información anteriores a la aprobación del documento de Lineamientos técnicos⁵⁵ de la DTIC y circunstancias muy específicas que requieran de funcionalidades no existentes, en plataformas libres y gratuitas, las cuales deberán justificarse oportunamente en el documento de análisis.

Sistema Operativo.

Para el desarrollo y producción del SIM se utilizará el Sistema Operativo Linux, en su distribución Debian 7.0 Wheezy, como alternativa se permitirá la utilización de Ubuntu únicamente para desarrollo del Sistema.

Framework de Programación.

El Framework de programación que se utilizará para el desarrollo del sistema es Symfony2, en conjunto con el motor de plantillas de PHP llamado Twig, y para el mapeo objeto-relacional de la base de datos, se utilizara el Framework ORM Doctrine2. Adicionalmente, para la administración de la aplicación (operaciones CRUD y generación automática de formularios), se debe utilizar el Bundle Sonata, el cual está basado en Symfony.

⁵⁵ Ministerio de Salud. (2011). *Lineamientos Técnicos para el Mantenimiento, Administración y Desarrollo de Tecnologías de Información y Comunicaciones*. San Salvador, El Salvador.



Sistema Gestor de Base de Datos.

El SGBD que se utilizará, es PostgreSQL en su versión 9.1. Como herramienta para el manejo de la base de datos puede utilizarse PgAdmin III o phpPgadmin; sin embargo, queda a disposición del grupo de trabajo la utilización de la consola del sistema operativo con la herramienta embebida PSQL.

Control de Versiones.

Para el control de versiones, se utilizará Git, que permitirá el registro de los cambios realizados sobre un archivo o conjunto de archivos a lo largo del tiempo, de modo que se pueda recuperar a versiones anteriores.

10.3.3.2 Contenido Mínimo de Manuales Adjuntos al Sistema Informático.

Los manuales requeridos por la DTIC y que se entregarán junto con el Sistema Informático desarrollado son:

- **Manual de Instalación:** Contendrá el detalle completo de la instalación del sistema desarrollado, además de las configuraciones necesarias para que el sistema pueda funcionar.
- **Manual de Administración:** Será una guía para realizar la administración del sistema desarrollado.
- **Manual de Usuario:** Este manual debe reunir la información adecuada para que el usuario pueda conocer y utilizar adecuadamente el software desarrollado. Debe estar redactado en lenguaje entendible para el usuario final no para un usuario informático.
- **Manual Técnico:** Este documento debe contener todo el análisis y diseño realizado para el desarrollo del Sistema de Información. Debe contener como mínimo los siguientes apartados:
 - Descripción de la problemática a resolver y propuesta de solución.
 - Alcance detallado del sistema.
 - Requerimientos de software y hardware (lado cliente y servidor).
 - Diagramas UML, especialmente de:
 - Casos de uso.
 - Actividades o procesos.
 - Clases.
 - Estados (opcional).
 - Esquema conceptual de la base de datos (EER) y traducción al modelo relacional (SQL).
 - Diccionario de datos del modelo relacional.
 - Cronograma estimado de entregas.
 - Estimación de necesidades para la sostenibilidad del sistema (recursos humanos y otros).
 - Diseño de las interfaces junto con la explicación técnica de cada una de ellas.
 - Descripción de los roles y privilegios de los usuarios del sistema.



10.3.3.3 Restricciones de Diseño.

Las Restricciones de Diseño son conjunto de consideraciones basadas en lineamientos, políticas y estándares de índole obligatoria, que restringen o limitan en cierta medida el libre Diseño del Sistema. Sin embargo, su definición y conocimiento previo, permite una correcta concepción de los elementos involucrados en el Sistema a Desarrollar.

Restricciones de Base de Datos.

Conjunto de consideraciones a tomar en cuenta en el Diseño de la Base de Datos, estas consideraciones involucran los formatos de nombres de objetos de la Base de Datos, y la estandarización de la definición y estructura de los mismos. Estos estándares se resumen en la Tabla 10-7.

Restricciones y Estándares para el Diseño de la Base de Datos
Consideraciones Generales
Deben tomarse en cuenta las siguientes consideraciones: <ul style="list-style-type: none"> • No usar espacios en blanco para los nombres de los objetos. • No utilizar letras mayúsculas, ni tildes tampoco caracteres especiales. • Usar guion bajo (_) para separar las palabras.
Nombre de la Base de Datos
El nombre usado para la base de datos debe ser definido usando un nombre corto y se recomienda usar el nombre del proyecto todo en minúsculas.
Nombres de Tablas
Durante el nombramiento de las tablas de la base de datos se deben tomar en cuenta las siguientes recomendaciones: <ul style="list-style-type: none"> • Nombres auto-descriptivos: Los nombres utilizados para definir las tablas deben de ser en lo posible auto-descriptivos, y evitar el uso de abreviaciones. Se recomienda ampliamente el uso de nombres cortos. • Nombres en singular: Los nombres de las tablas deben de ser definidos en singular. En el caso de nombres de tablas con más de una palabra ambas deben estar en singular separadas por un guion bajo. • Prefijos: Dentro de la base de datos pueden existir ciertas tablas que necesiten agruparse según la lógica del negocio. En estos casos puede colocarse un prefijo de entre dos a tres caracteres para identificar el grupo o modulo al que pertenecen. • Tablas de referencia: Los nombres de las tablas que son usadas para guardar referencias (many to many) deben de definirse en lo posible usando el nombre de ambas tablas. Ejemplo: ctl_especialidad_ctl_establecimiento • Tablas catálogos: el nombre estará compuesto de la siguiente forma: ctl_nombre_catalogo. Ejemplos: ctl_pais



<p>Nombres de Campos</p> <p>Las columnas son atributos de la entidad, es decir, las columnas describen las propiedades de la entidad por tanto los nombres de las columnas deben ser representativos y naturales.</p> <p>El campo que será llave primaria debe llamarse id.</p> <p>Si es una tabla de referencia colocar las primeras tres letras de la primera tabla, guion bajo (_), las primeras tres letras de la segunda tabla y el nombre del campo.</p> <p>El nombre de la columna que contenga una llave foránea hacia otra tabla dentro de la misma base de datos, deber de ser la siguiente: id_nombre_tabla_padre sin el uso de prefijo.</p>
<p>Usuario de la Base de Datos</p> <p>Se debe crear un usuario dueño de la base de datos sin permisos de creación de base de datos, ni roles, ni ser súper usuario con permisos solo sobre la base y sus objetos relaciones.</p>
<p>Llaves Primarias</p> <p>La llave primaria es la columna que identifica cada fila en la tabla. Para nombrarlas se utilizará:</p> <p style="text-align: center;">pk_nombre_tabla</p> <p>Nota: Si se trabajara con Symfony2 y Doctrine2 no se pueden realizar llaves primarias compuestas; por ende crear un campo auto-numérico como llave primaria y se debe crear un índice para que los dos campos sean únicos.</p>
<p>Llaves Foráneas</p> <p>Por convención estas pueden nombrarse de la siguiente forma:</p> <p style="text-align: center;">fk_nombre_tabla_padre_nombre_tabla_hija</p> <p>Si las tablas padre e hija poseen prefijo, cuando se nombre la llave foránea debe ir sin el prefijo que posee.</p>
<p>Vistas</p> <p>Se nombran con un nombre significativo de la información que contienen utilizando el prefijo "vw_"</p>
<p>Procedimientos</p> <p>Los procedimientos pueden nombrarse de acuerdo a lo que ellos realizan dentro de la base de datos; puede utilizarse un verbo en infinitivo para describir dicha acción. Deben comenzar con el prefijo "proc_".</p>
<p>Triggers o Disparadores</p> <p>Son procedimientos especiales que se ejecutan luego de una determinada acción en una tabla. Durante el nombrado de los "triggers" se deben tomar en cuenta las siguientes recomendaciones:</p> <ul style="list-style-type: none"> • Prefijos: se usará prefijo "trg", que es ampliamente recomendado con el propósito de hacer saber que el objeto en la base de datos es un trigger. • Nomenclatura: Para su nombramiento, al igual que los procedimientos se puede utilizar un verbo en infinitivo precedido del nombre de la tabla a la que pertenece. Por ejemplo: <p style="text-align: center;">trg_accion_tabla</p> <p>Excepciones: las funciones a las que hacen referencias los triggers se deben llamar con el mismo nombre del trigger agregando el prefijo fn_nombre_trigger.</p>



<p>Índices</p> <p>Los índices solo existen para una determinada tabla, para uno o más campos separados por guion bajo (_). Su nombramiento puede ser de la siguiente manera:</p> <p style="padding-left: 40px;">idx_nombre_campo(s)</p> <p>Para el nombre del índice se deben colocar máximo 2 campos.</p>
<p>Documentación Interna</p> <p>Se debe realizar un comentario para cada campo, tablas, procedimientos, triggers o cualquier elemento que contenga la base de datos; esto con el fin de tener claros las funcionalidades de los elementos que constituyen la base de datos. La documentación debe de ser descriptiva y no redundante.</p>
<p>Catálogos Generales</p> <p>Dentro del MINSAL se utilizan una serie de catálogos y tablas generales, los que han sido proporcionados al Equipo de Trabajo de Graduación por medio de archivos de imagen⁵⁶.</p> <p>Para la integración y compatibilidad futura con otros sistemas informáticos del MINSAL, el diseño de la Base de Datos realizado por el Equipo de Trabajo, debe apegarse al diseño de estos catálogos y tablas generales en lo referente al manejo de:</p> <p>Instituciones, Establecimientos, Dependencias, Unidades, Empleados, Experiencia Laboral, Nivel Educativo, Domicilio, Departamentos y Municipios.</p> <p>En cuanto a los demás catálogos queda a decisión del grupo, que tablas y campos de los catálogos proporcionados son requeridos por el sistema para su desarrollo y funcionamiento.</p>

Tabla 10-7. Restricciones de Diseño de la Base de Datos.

⁵⁶ Ver imagen en Anexo 1 – Catálogo General DTIC, del Documento de Requerimientos (SRS) que se encuentra dentro del directorio /Documentos/2 – Documento de Requerimientos – SRS/ del DVD anexo.



10.3.3.4 Consideraciones de Seguridad

La seguridad en los Sistemas Informáticos es un área importante que se enfoca en la infraestructura computacional pero especialmente en la información almacenada en ellos o la que circula a través de las redes de comunicación. Algunas de las consideraciones de seguridad involucra la gestión de roles y usuarios, inicios de sesión y registro de acciones realizadas en el sistema las cuales son básicas para mantener la información segura y confiable.

Bitácora (Registro de Acciones del Sistema)

El sistema deberá contar con un historial o registro de las acciones que realicen los usuarios dentro del sistema, con el objetivo de poder consultar posteriormente por motivo de seguridad o auditoría lo que realizó un usuario en un determinado momento.

El registro de la bitácora, contará con los siguientes datos:

- La acción realizada, ya sea de creación, actualización o eliminación de registros en la base de datos.
- La fecha y hora de la realización.
- El Usuario que realizó la acción.

Consideraciones a incluir en la bitácora, según análisis del Grupo de Trabajo:

- a) Guardar la descripción de la acción realizada, para facilitar auditorías futuras, por ejemplo:
"Se creó una nueva solicitud de mantenimiento correctivo".
"Inicio de sesión exitosa".
"Fallo en inicio de sesión, ingreso de contraseña no válida".
- b) Las acciones a registrar deberán ser las más importantes o aquellas que son vitales en los procesos de mantenimiento, para evitar el crecimiento excesivo de la base de datos. Las acciones que se propone registrar en la bitácora son (Propuesta realizada por el equipo de desarrollo):
 - Creación, modificación y eliminación de Usuarios.
 - Actualización de datos de Empleados.
 - Creación y modificación de Modelos de equipos.

10.3.3.5 Suposiciones y Dependencias.

Las Suposiciones y Dependencias a continuación, son un conjunto de elementos que se asume que existen, que serán facilitados por el ente o persona responsable y de los cuales depende correcto el funcionamiento del Sistema a desarrollar. Es una de las partes vitales del presente documento, ya que el cambio o la falta de alguna de estas suposiciones generará un impacto en el proyecto puede afectar el cumplimiento exitoso de las actividades del mismo.



Costo de Materiales y Repuestos.

- Para un correcto cálculo de los costos de materiales y repuestos incurridos en la realización de los órdenes de trabajo de los servicios de mantenimiento, el Sistema a desarrollar requiere que en las requisiciones de materiales y repuestos que se realizan a través del SINAB, uno o más vales de salidas sean utilizados sola y únicamente en una orden de trabajo, es decir, que los materiales o repuestos descritos en el vale de salida no sean utilizados en dos más ordenes de trabajo, sino que puedan ser cargados a una única orden.

Datos del Inventario de Activo Fijo.

- Se tiene como supuesto que los datos que se encuentran en el SIAF, y que serán requeridos por el SIM son consistentes.
- El Bien (equipo, mobiliario, maquinaria, etc. – ver definición en apartado 7.1.2) que no se encuentre inventariado en el SIAF, y al cual se requiera brindar servicios de mantenimiento, es necesario realizar un registro de este dentro del Sistema a desarrollar. A diferencia del Bien que se encuentre en el SIAF, es necesario el ingreso de los datos que normalmente serian registrados dentro del SIAF y que son requeridos por el Sistema a desarrollar.

Solicitud de Mantenimiento

- El Jefe y Coordinadores de Unidad, son los únicos que podrán realizar solicitudes de mantenimiento. Sin embargo, los Coordinadores de Unidad deberán esperar aprobación de las solicitudes de servicio por parte del Jefe de Unidad, para ser enviadas al Departamento de Mantenimiento.

Inventario Técnico de Bien

- El Jefe de Mantenimiento (o un designado por este) será el encargado, de registrar los datos técnicos de los equipos, maquinaria y bienes que ingresen en las dependencias del Ministerio de Salud, luego de haber sido registrados en el SIAF o cuándo el contrato de mantenimiento por garantía se halla vencido.
- El código principal a utilizar dentro del sistema, será el código asignado al momento de la creación o registro del Bien dentro del inventario técnico del SIM, como código alternativo, también se podrá utilizar el Código del Inventario de Activo Fijo, esto solamente para equipos, maquinaria y mobiliario que se encuentre registrado en el SIAF. En la Figura 10-2 se muestra la estructura del Código del Inventario técnico, el cual se describe en la Tabla 10-8.

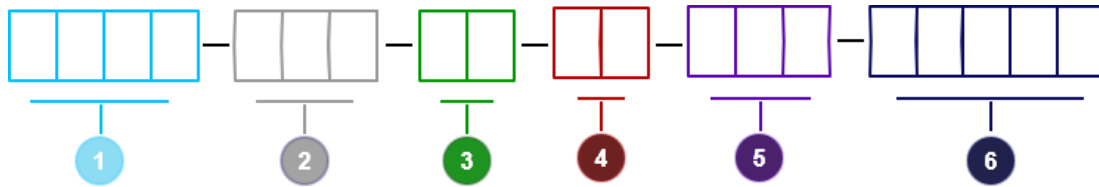


Figura 10-2 Estructura del Código de Inventario Técnico.

Elemento	Nombre	Descripción
1	Código de Establecimiento.	Código de cuatro dígitos que indica a que establecimiento pertenece el inventario técnico.
2	Código de Unidad.	Código de dos dígitos, que indica en que Unidad, Departamento, Servicio o Ambiente dentro del establecimiento se encuentra ubicado el Bien.
3	Grupo	Código de dos dígitos, que permite clasificar el Bien basado en tres criterios: <ul style="list-style-type: none"> • Área de aplicación del Bien. • Función del Bien. • Afinidad.
4	Clasificación Técnica.	Código de dos dígitos, Clasificación más detallada del Bien, que brinda detalles más precisos sobre el bien.
5	Código de Modelo	Esta sección puede estar compuesta por el código del Modelo, el cual es código formado por 3 dígitos.
6	ID del Bien	Código de cinco dígitos que representa un identificador único para cada Bien del Inventario Técnico.

Tabla 10-8 Descripción de la Estructura del Código de Inventario Técnico.

Funcionamiento del SIM

- Para la puesta en marcha del SIM, se capacitará a los usuarios en el manejo de equipo informático y del uso del Sistema, y se dotará a las unidades y departamentos de este equipo, para que se realice una efectiva utilización del Sistema y se obtengan los resultados esperados.

Mantenimiento Preventivo Planificado

- Las Rutinas de Mantenimiento de los Equipos, se manejarán en el Sistema como archivos en formato PDF.



10.4 Especificación de Requerimientos.

La especificación de requerimientos de software, es la descripción detallada de las necesidades y/o características que el Sistema ha de cubrir o soportar y lo que ha de realizar. De forma general, se define el comportamiento del mismo; así como, aquellos elementos con los que debe cumplir.

10.4.1 Interfaces externas.

Interfaces externas comprende el conjunto de conexiones físicas o funcionales entre el SIM y otros sistemas, que van desde interfaces de usuario, hasta protocolos de comunicación, entre otros.

10.4.1.1 Interfaces del Sistema.

Las Interfaces de Sistema son aquellas que permiten la interconexión con sistemas externos y el consumo de servicios requeridos de estos sistemas. Las Interfaces del Sistema se presentan en el Cuadro 10-3:

Código	Descripción
ISI-01	El sistema permitirá la integración con el Sistema Informático de Activo Fijo del Ministerio de Salud (SIAF). El propósito es obtener el código oficial de Activo Fijo de los diferentes Equipos que incorpore el SIM y algunas características básicas de estos bienes.
ISI-02	El sistema permitirá la integración con el Sistema Informático Nacional de Abastecimiento (SINAB). El propósito es obtener la información de los vales de salida y los costos incurridos en materiales y repuestos, que pertenecen a una orden de trabajo de mantenimiento determinada.

Cuadro 10-3. Interfaces del Sistema.

10.4.1.2 Interfaces de Usuario.

Su principal objetivo, consiste en proporcionar un entorno visual para permitir una fácil interacción entre el Sistema y el usuario. Las Interfaces de Usuario se presentan en el Cuadro 10-4.

Código	Descripción
IUS-01	Permitir el acceso al sistema mediante ambiente web.
IUS-02	El sistema estará optimizado para los navegadores web Iceweasel, Mozilla Firefox y Google Chrome.
IUS-03	La interfaz debe de ser adaptable con un ancho mínimo de 979px, y un ancho máximo del 85% de la resolución de la pantalla.
IUS-04	La interfaz del sistema debe presentar un banner con el nombre del sistema y el logo del Ministerio de Salud, con una resolución de 750px de ancho por 73px de alto.
IUS-05	El menú del sistema debe de ser de tipo horizontal con sub-menús desplegables hasta un nivel 3 de profundidad, con una altura de 40px.
IUS-06	Se debe de utilizar colores institucionales entre los que se encuentran, gris, blanco, azul y sus derivados.

Cuadro 10-4. Interfaces de Usuario.



10.4.1.3 Interfaces de Software.

Las Interfaces de Software especifican el uso de otros productos de software necesarios para el desarrollo, pruebas y puesta en marcha del SIM. Las Interfaces de Software se presentan en el Cuadro 10-5:

Código	Descripción
ISW-01	El sistema deberá ser desarrollado sobre el Sistema Operativo Debian 7.0 stable versión (Debian Wheezy) ya que también será puesto en producción sobre este Sistema Operativo.
ISW-02	Para el desarrollo y administración de la base de datos, el sistema deberá utilizar el Sistema Gestor de Base de Datos PostgreSQL versión 9.1.
ISW-03	Para el mapeo objeto-relacional de la base de datos, se utilizara el Framework ORM Doctrine2.
ISW-04	Se deberá utilizar el Framework de desarrollo Symfony 2, integrado con el Bundle basado en Symfony, "Sonata Bundle".

Cuadro 10-5. Interfaces de Software.

10.4.1.4 Interfaces de Comunicación.

Las Interfaces de Comunicación son las conexiones físicas y funcionales que permite la interacción entre diferentes dispositivos o sistemas. Su objetivo es definir los requerimientos de comunicación en relación a protocolos, puertos, etc. Las Interfaces de Comunicación se presentan en el Cuadro 10-6:

Código	Descripción
ICO-01	Se requiere que la comunicación del sistema se realice por medio del protocolo TCP.
ICO-02	Se utilizará el puerto 80 para el acceso web al sistema, por medio del protocolo HTTP.
ICO-03	Para la comunicación con la base de datos PostgreSQL se utilizará el puerto 5432.
ICO-04	El sistema será accesible solamente en la red interna del MINSAL (INTRANET), es decir, que no se podrá acceder a él por medio de Internet.
ICO-05	Para el control de versiones del software, en su desarrollo se utilizará Git con protocolo SSH, que utiliza el puerto TCP 22, esto se realizará por medio de una llave pública que se deberá registrar en el servidor de Git.
ICO-06	Para la comunicación por medio de correo electrónico se utilizará el protocolo SMTP (Simple Mail Transfer Protocol), puerto TCP 25.

Cuadro 10-6 Interfaces de Comunicación.



10.4.1.5 Requerimientos de Adaptación del Sitio.

Los requerimientos de adaptación del sitio comprenden las necesidades de software, hardware (ver apartado 10.4.3.1) y la configuración requeridas para que se cumpla una correcta funcionalidad del sitio. Los Requerimientos de Adaptación del Sitio se presentan en el Cuadro 10-7:

Código	Descripción
RAS-01	Instalación y configuración correcta de Apache 2.2 como servidor HTTP (Servidor Web).
RAS-02	Instalación y configuración de PHP 5, que será lenguaje de desarrollo a utilizar para el código fuente del sistema.
RAS-03	Instalación de librerías php-apc y php5-intl.
RAS-04	Implementación de Symfony2 como Framework de Desarrollo Web de PHP, para la construcción del sistema.
RAS-05	Implementación y configuración del Bundle Sonata, para la administración de usuarios del sistema.
RAS-06	Implementación del ORM Doctrine2 como motor de persistencia para la conexión entre la base de datos y symfony2.
RAS-07	Instalación de los Drivers php5-sybase y Freetds para la conexión con las bases de datos: PostgreSQL y SQL Server, configuración del Sistema y de Doctrine para la correcta utilización de los Drivers.

Cuadro 10-7. Requerimientos de Adaptación del Sitio.



10.4.2 Requerimientos funcionales.

Los requerimientos funcionales definen las acciones específicas que debe proporcionar el Sistema y su comportamiento ante determinados escenarios, que van desde la captura de datos, el procesamiento de estos, y las respectivas salidas que debe entregar como resultado.

10.4.2.1 Requerimientos Funcionales Generales.

Teniendo en cuenta la definición anterior, a continuación, se describe de manera general los requerimientos funcionales que el Sistema debe cumplir, así como los actores que podrán realizar las acciones descritas.

Código	Descripción	Actor/es
RFU-01	El sistema deberá permitir el inicio y cierre de sesión.	Todos
RFU-02	El sistema deberá permitir el cambio de contraseña.	Todos
RFU-03	El sistema deberá permitir el cambio de datos de usuario.	Todos
RFU-04	El sistema deberá permitir la administración de grupos.	adminSist
RFU-05	El sistema deberá permitir la asignación de roles.	adminSist
RFU-06	El sistema deberá permitir la administración de usuarios.	adminSist
RFU-07	El sistema deberá permitir la consulta de la bitácora del sistema.	adminSist
RFU-08	El sistema deberá permitir la gestión de los establecimientos.	adminSist
RFU-09	El sistema deberá permitir la administración de tipos de establecimientos.	adminSist
RFU-10	El sistema deberá permitir la gestión de las unidades.	adminSist, adminEstb
RFU-11	El sistema deberá permitir la administración de tipo de unidades. Las cuales pueden ser: servicios, ambientes, áreas, secciones, jefaturas y departamentos.	adminSist
RFU-12	El sistema deberá permitir la gestión de empleados.	adminSist, adminEstb
RFU-13	El sistema deberá permitir la administración de puestos de empleado	adminSist, adminEstb
RFU-14	El sistema deberá permitir la realización de solicitudes de servicio de mantenimiento.	jefeManto,jefeUni, solict
RFU-15	El sistema deberá permitir verificar el estado de la solicitud de servicio.	jefeUni, solict
RFU-16	El sistema deberá permitir la aprobación de las solicitudes de mantenimiento.	jefeManto,jefeUni
RFU-17	El sistema deberá permitir la administración de planes de mantenimiento.	jefeManto
RFU-18	El sistema deberá permitir la gestión de períodos de planes de mantenimiento.	jefeManto
RFU-19	El sistema deberá permitir crear órdenes de trabajo.	jefeManto



Código	Descripción	Actor/es
RFU-20	El sistema deberá permitir la reasignación de una orden de trabajo	jefeManto
RFU-21	El sistema deberá permitir la cancelación de órdenes de trabajo.	jefeManto
RFU-22	El sistema deberá permitir llenar el informe de orden de trabajo.	jefeManto, tecnManto
RFU-23	El sistema deberá permitir la gestión de los Modelos de los diferentes bienes del inventario técnico propiedad del Ministerio de Salud.	adminEstb, jefeManto
RFU-24	El sistema deberá permitir la clasificación de bienes.	adminEstb
RFU-25	El sistema deberá permitir la gestión de grupos y subgrupos de bienes.	adminEstb
RFU-26	El sistema deberá permitir la administración de bienes y equipos.	adminEstb,jefeManto
RFU-27	El sistema deberá permitir la consulta de planes de Mantenimiento Asignado.	tecnManto
RFU-28	El sistema deberá permitir la consulta de la orden de trabajo.	jefeManto, tecnManto
RFU-29	El sistema deberá permitir Iniciar la orden de trabajo.	jefeManto, tecnManto
RFU-30	El sistema deberá permitir Finalizar la orden de trabajo.	jefeManto, tecnManto
RFU-31	El sistema deberá permitir la consulta de las características del equipo.	jefeManto, tecnManto
RFU-32	El sistema deberá permitir la consulta de la ficha de vida del equipo.	jefeManto, tecnManto
RFU-33	El sistema deberá permitir la gestión de contratos.	adminEstb
RFU-34	El sistema deberá permitir la gestión de empresas.	adminEstb
RFU-35	El sistema deberá permitir la generación del reporte de Bitácora de Usuario.	adminSist
RFU-36	El sistema deberá permitir la generación de reporte de usuarios.	adminSist
RFU-37	El sistema deberá permitir la generación de reportes de los empleados registrados.	adminSist
RFU-38	El sistema deberá permitir la generación del reporte de estado de solicitud de servicio.	jefeManto, jefeUni, solict
RFU-39	El sistema deberá permitir la generación del plan de mantenimiento anual.	jefeManto
RFU-40	El sistema deberá permitir la generación del reporte de la orden de trabajo.	jefeManto, tecnManto
RFU-41	El sistema deberá permitir la generación de listado de órdenes de trabajo asignadas.	jefeManto
RFU-42	El sistema deberá permitir la generación del reporte de la hoja o ficha de vida del equipo.	jefeManto, tecnManto
RFU-43	El sistema deberá permitir la generación del reporte de informe de mantenimiento de una orden de trabajo.	jefeManto, tecnManto



Código	Descripción	Actor/es
RFU-44	El sistema deberá permitir la generación del reporte de costo por servicio (o depto., o área, etc.).	jefeManto
RFU-45	El sistema deberá permitir la generación del reporte de mantenimientos pendientes.	jefeManto, tecnManto
RFU-46	El sistema deberá permitir la generación del reporte de mantenimientos realizados.	jefeManto
RFU-47	El sistema deberá permitir la administración de la orden de trabajo.	jefeManto
RFU-48	El sistema deberá permitir la administración del informe de la orden de trabajo.	jefeManto, tecnManto
RFU-49	El sistema deberá permitir la consulta del informe de la orden.	jefeManto, tecnManto
RFU-50	El sistema deberá permitir la actualización del informe de la orden.	jefeManto, tecnManto
RFU-51	El sistema deberá permitir consultar las solicitudes de servicio.	jefeUni, solict

Cuadro 10-8 Requerimientos Funcionales Generales.

A partir de los requerimientos funcionales descritos en la Cuadro 10-8 se ha determinado la siguiente lista actor-objetivo.

Actor/es	Objetivo.
jefeUni, solict	Crear Solicitud de Servicio
	Consultar Solicitud de Servicio
	Consultar Estado de Servicio
jefeManto, jefeUni	Autorizar Solicitud de Servicio
jefeManto, tecnManto	Iniciar Orden de Trabajo
	Finalizar Orden de trabajo
	Consultar plan Asignado
	Administrar Informe de Orden
jefeManto	Administrar Orden de Trabajo
	Administrar Asignación de Plan
	Gestionar Plan Preventivo
	Consultar Ficha de vida
	Gestionar Medida
adminSist, adminEstb, jefeManto	Transferir Solicitud de Servicio
	Gestionar puestos de empleados
	Administrar Unidad
	Administrar Empleado



Actor/es	Objetivo.
adminEstb, jefeManto	Gestionar Empresa
	Administrar Modelo
	Gestionar Grupo de Bien
	Gestionar Clasificación Técnica
	Administrar contrato
	Administrar Inventario Técnico.
	Administrar Clasificación de Bien
	Administrar condiciones de contrato
adminSist	Administrar Establecimiento
	Consultar Bitácora
	Administrar Usuario
	Administrar Grupo
	Gestionar Tipo de establecimiento
	Gestionar tipo de unidad.

Cuadro 10-9 Lista Actor - Objetivo.



10.4.2.2 Requerimientos Funcionales Específicos.

Los requerimientos funcionales específicos describen el comportamiento del sistema a un mayor nivel de detalle que en los requerimientos funcionales generales, los cuales se especifican por medio de la Descripción de los Casos de Uso (Para ver la definición de Caso de Uso y los elementos que intervienen en su Descripción, vea la sección 10.1.2.1 del presente documento):

Casos de Uso - Tipo de Establecimiento

Código:	CUS-01-01	Nombre:	Consultar Tipo Establecimiento.
Actores:	Administrador General del Sistema.		
Propósito:	Consultar datos de los Tipos de Establecimiento.		
Descripción:	Se presenta los tipos de establecimientos registrados.		
Precondiciones:	<ol style="list-style-type: none"> 1. El usuario debe haberse autenticado. 2. Existe al menos un registro de Tipo Establecimiento. 		
Post-Condiciones:	Se muestra los datos de un registro de Tipo de Establecimiento.		
Flujo Básico			
Paso	Acción		
1	El sistema muestra una lista que corresponde a los diferentes registros de Tipos de Establecimiento, y pide que seleccione uno de ellos para mostrar su información.		
2	El usuario selecciona de la lista el Tipo de Establecimiento deseado.		
3	El sistema muestra los datos del Tipo de Establecimiento seleccionado.		
4	Fin del caso de uso.		
Flujos Alternos			
Paso en el que fue invocado	Acción		
1	A. El sistema no muestra la lista. <ol style="list-style-type: none"> 1. Se muestra un mensaje de que no existen tipos de establecimientos registrados. 2. El sistema continúa con el paso 4 del flujo básico. 		
2	A. Búsqueda por filtro de información. <ol style="list-style-type: none"> 1. El usuario ingresa los datos correspondiente a los campos del filtro por los cuales desea realizar la búsqueda. 2. El usuario ejecuta la opción Filtrar. 3. El sistema continúa con el paso 2 del flujo básico. 		
Frecuencia Esperada:	Anual.		
Importancia:	Moderada		
Urgencia:	Moderada		
Referencias Cruzadas:	RFU-09		



Código:	CUS-01-02	Nombre:	Agregar Tipo Establecimiento.
Actores:	Administrador General del Sistema.		
Propósito:	Agregar un nuevo Tipo de Establecimiento.		
Descripción:	El usuario realiza el registro de un nuevo tipo de establecimiento.		
Precondiciones:	1. El usuario debe haberse autenticado.		
Post-Condicion:	Se creó un nuevo registro de Tipo de establecimiento.		
Flujo Básico			
Paso	Acción		
1	El sistema solicita el ingreso de los siguientes datos: <ul style="list-style-type: none"> • Nombre del Tipo Establecimiento. • Código. 		
2	El usuario ingresa los datos que son requeridos y ejecuta la opción "Crear".		
3	El sistema valida los datos ingresados.		
4	Se registra el nuevo Tipo de Establecimiento y se muestra un mensaje de éxito al usuario.		
5	Fin del caso de uso.		
Flujos Alternos			
Paso en el que fue invocado	Acción		
3	A. Error de validación. <ol style="list-style-type: none"> 1. El sistema muestra un mensaje de error debido a que alguno de los datos requeridos se ha ingresado incorrectamente o no se ha especificado. 2. El sistema continúa con el paso 1 del flujo básico. 		
Frecuencia Esperada:	Anual		
Importancia:	Alta		
Urgencia:	Moderada		
Referencias Cruzadas:	RFU-09		



Código:	CUS-01-03	Nombre:	Modificar Tipo Establecimiento.
Actores:	Administrador General del Sistema.		
Propósito:	Modificar datos de los Tipos de Establecimiento que han sido registrados.		
Descripción:	El usuario podrá realizar la Modificación de un Tipo de Establecimiento.		
Precondiciones:	<ol style="list-style-type: none"> 1. El usuario debe haberse autenticado. 2. Existe al menos un registro de Tipo de Establecimiento. 		
Post-Condiciones:	Se actualizó correctamente los datos de un registro de Tipo de Establecimiento.		
Flujo Básico			
Paso	Acción		
1	Se realiza el caso de uso Consultar Tipo Establecimiento.		
2	El usuario selecciona la opción Editar.		
3	El sistema muestra el formulario correspondiente a la edición del Tipo de Establecimiento.		
4	El usuario modifica los datos del Tipo de Establecimiento.		
5	El usuario selecciona la opción Actualizar.		
6	El sistema valida los datos modificados.		
7	El sistema actualiza correctamente los datos del Tipo de Establecimiento seleccionado y muestra un mensaje de éxito.		
8	Fin del caso de uso.		
Flujos Alternos			
Paso en el que fue invocado	Acción		
6	A. Error de validación. <ol style="list-style-type: none"> 1. El sistema muestra un mensaje de error debido a que alguno de los datos que han sido modificados es incorrecto o no ha sido especificado. 2. El sistema continúa con el paso 3 del flujo básico. 		
Frecuencia Esperada:	Anual.		
Importancia:	Moderada.		
Urgencia:	Moderada.		
Referencias Cruzadas:	RFU-09		



Código:	CUS-01-04	Nombre:	Eliminar Tipo Establecimiento.
Actores:	Administrador General del Sistema.		
Propósito:	Eliminar completamente uno de los Tipos de Establecimiento que han sido registrados.		
Descripción:	El usuario tendrá la opción de eliminar un Tipo de Establecimiento determinado, luego de verificar su información.		
Precondiciones:	<ol style="list-style-type: none"> 1. El usuario debe haberse autenticado. 2. Existe al menos un registro de Tipo de Establecimiento. 		
Post-Condiciones:	Se eliminó correctamente un registro completo de Tipo de Establecimiento.		
Flujo Básico			
Paso	Acción		
1	Se realiza el caso de uso Consultar Tipo Establecimiento.		
2	El usuario ejecuta la opción Editar		
3	El sistema muestra el formulario correspondiente a la edición del Tipo de Dependencia.		
4	El usuario ejecuta la opción Eliminar.		
5	El sistema pide confirmar la acción a realizar.		
6	El usuario confirma la acción.		
7	El sistema verifica que no existan otros registros (Establecimientos) que hacen referencia al registro de Tipo de Establecimiento a eliminar y realiza la eliminación.		
8	El sistema muestra al usuario un mensaje de éxito.		
9	Fin del caso de uso.		
Flujos Alternos			
Paso en el que fue invocado	Acción		
6	A. Cancelación de Operación. <ol style="list-style-type: none"> 1. El usuario cancela la acción. 2. El sistema continúa con el paso 3 del flujo básico. 		
7	A. Error de eliminación. <ol style="list-style-type: none"> 1. El sistema muestra un mensaje de error debido a que existen otros registros (Establecimientos) que hacen referencia al registro de Tipo de Establecimiento a eliminar. 2. El sistema continúa con el paso 3 del flujo básico. 		
Frecuencia Esperada:	Anual.		
Importancia:	Moderada.		
Urgencia:	Moderada.		
Referencias Cruzadas:	RFU-09		

En este apartado solamente se mostrarán una fracción a medida de ejemplo de los casos de usos que pertenecen al SRS, para ver todos los casos de usos del SRS ver Sección 3.2.2 del Documento de Documento de Requerimientos (SRS)⁵⁷.

⁵⁷ Ver Documento de Requerimientos (SRS) que se encuentra en el directorio /Documentos/2 – Documento de Requerimientos – SRS/ del DVD anexo.



10.4.2.3 Diagramas de Secuencia del Sistema.

Antes de realizar el Diseño de cómo funcionará el software, es conveniente estudiar y definir su comportamiento como una “caja negra”. El comportamiento del sistema es una descripción de qué hace el sistema, sin explicar cómo lo hace. Esto puede ilustrarse mediante un Diagrama de Secuencia del Sistema.

Un Diagrama de Secuencia del Sistema es un gráfico que muestra, para un escenario específico de un Caso de Uso, los eventos que generan los actores externos (como usuarios, sistemas externos, entre otros.) relacionados con el sistema que se está estudiando y la respuesta del mismo:

En este apartado solamente se mostrarán una fracción de los Diagramas de Secuencias del Sistema a medida de ejemplo, para ver todos los Diagramas de Secuencias del Sistema del SRS ver Sección 3.2.2 del Documento de Documento de Requerimientos (SRS)⁵⁸.

Diagrama de Secuencia del Sistema – Consultar Tipo de Establecimiento.

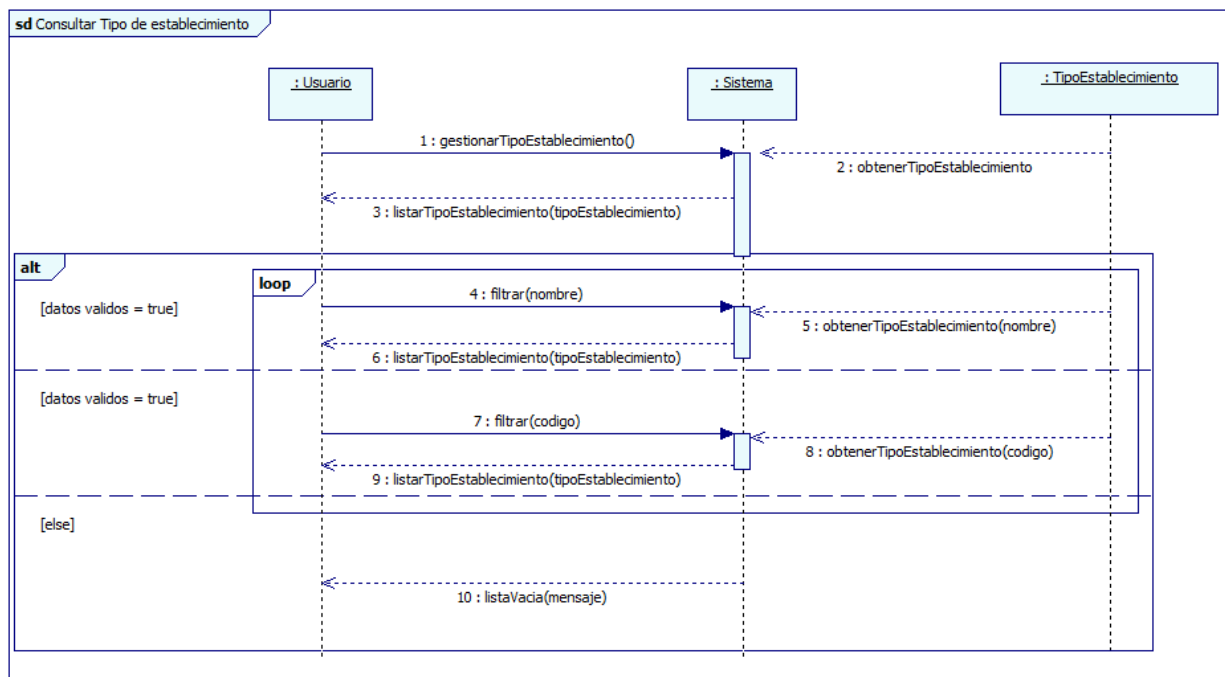


Figura 10-3. DSS Consultar Tipo de Establecimiento

⁵⁸ Ver Documento de Requerimientos (SRS) que se encuentra en el directorio /Documentos/2 – Documento de Requerimientos – SRS/ del DVD anexo.



Diagrama de Secuencia del Sistema – Crear Tipo de Establecimiento.

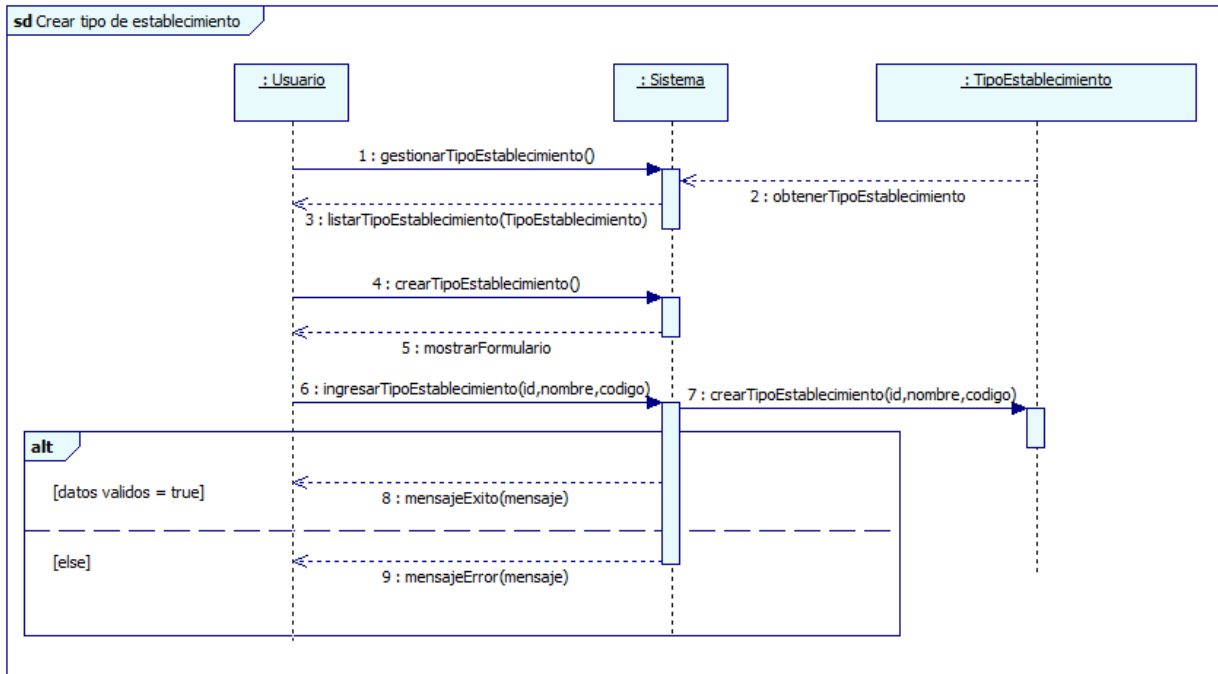


Figura 10-4. DSS Crear Tipo de Establecimiento.

Diagrama de Secuencia del Sistema – Modificar Tipo de Establecimiento.

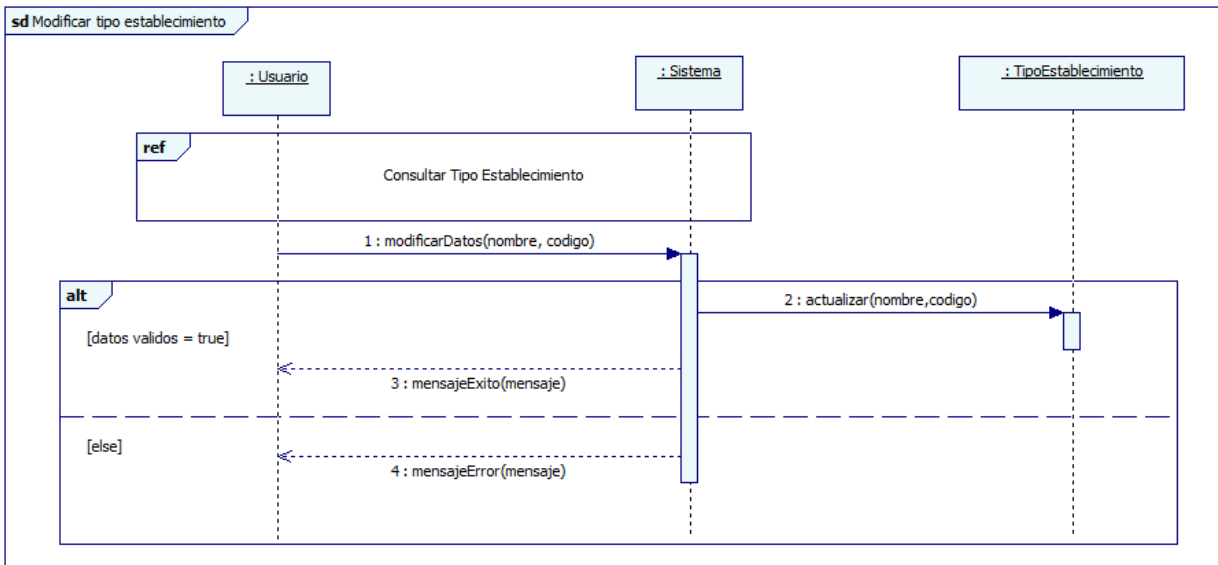


Figura 10-5. DSS Modificar Tipo de Establecimiento



Diagrama de Secuencia del Sistema – Eliminar Tipo de Establecimiento.

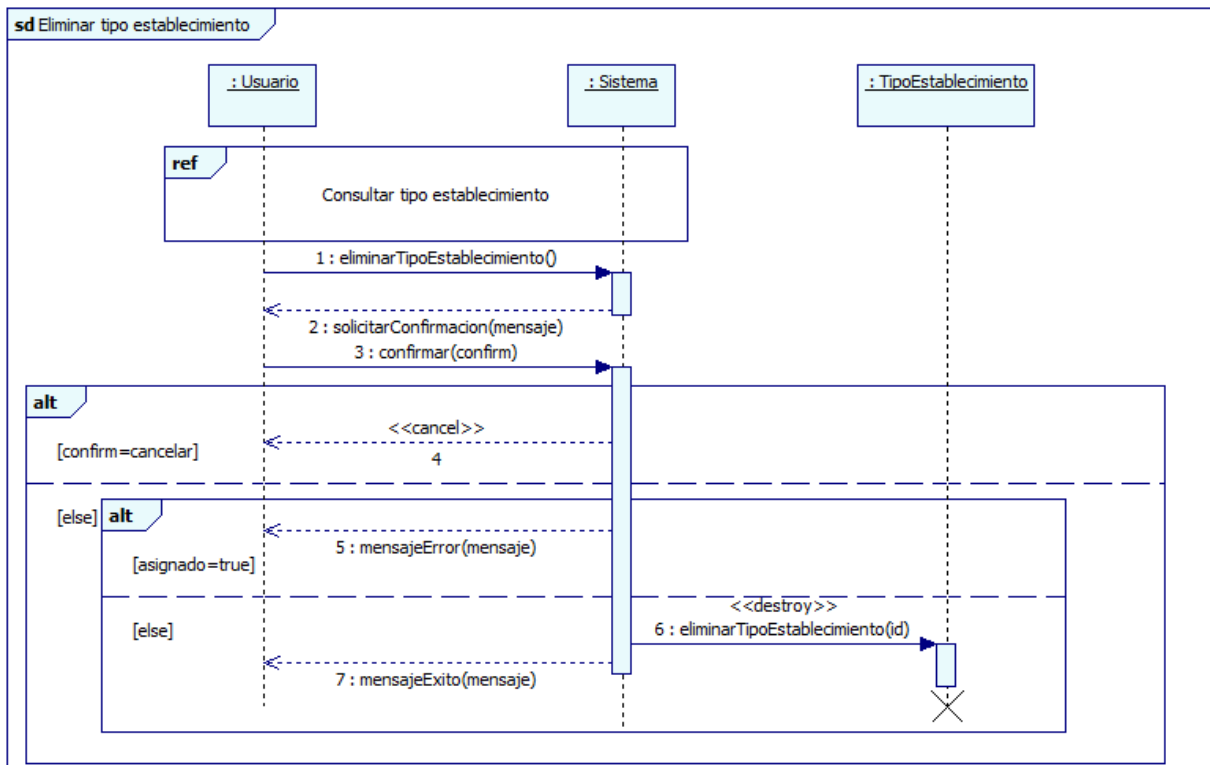


Figura 10-6. DSS Eliminar Tipo de Establecimiento.



10.4.3 Otros atributos y requerimientos.

Atributo de calidad	Descripción	Actividad	Responsable
Seguridad	Es la medida sobre la habilidad del sistema para resistir usos no autorizados mientras sigue proveyendo sus servicios a los usuarios legítimos.	Administración de accesos.	Administrador General del sistema.
		Creación de respaldos y restauración.	DTIC
		Inicio y cierre de sesión.	Todos los usuarios del sistema.
		Acceso físico y virtual al servidor.	DTIC
Facilidad de uso	Es la facilidad de aprender a utilizar el sistema por parte de los usuarios e interactuar con sus diferentes funcionalidades.	Capacitación.	DTIC.
		Creación de plan de implementación.	Desarrolladores del proyecto.
		Creación de manuales.	Desarrolladores del proyecto.
Facilidad de mantenimiento	Es la facilidad de agregar nuevas funcionalidades al sistema, corregir errores encontrados o modificar las funciones ya existentes. Así como la facilidad de modificar o la interfaz de usuario del sistema.	Manual Técnico del Sistema	Desarrolladores del proyecto.
		Documentación interna del software	Desarrolladores del proyecto.

10.4.3.1 Requerimientos de Hardware.

Los requerimientos de hardware especifican las características técnicas necesarias del equipo cliente y servidor que permitirán la implementación del producto a entregar.

Especificaciones Técnicas de Servidor

A continuación en la Tabla 10-9 se presentan las características y especificaciones técnicas del Hardware del Servidor donde estará alojado el servidor web y el sistema gestor de base de datos que contendrán el código fuente y base de datos del sistema:



Característica	Especificación Técnica
Modelo	Dell PowerEdge R815
Factor de Forma	2U rack
Procesador	AMD Opteron 6100 (48 Núcleos)
Sockets de Procesadores	4
Front Side Bus / HyperTransport	HyperTransport-3 Links
Cache	L2: 512K / core L3: 12MB
Chipset	AMD
Memoria RAM	64 GB (32 DIMM Slots) 1333MHz
I/O Slots	6 PCIe G2 slots + storage slot: 5 x8 slots 1 x4 slot 1 x4 storage slot
Disco Duro	3 TB
Conexiones de Red	Broadcom BMC57710 10Base-T Copper Single Port NIC, PCI-E x8.
Disponibilidad	Hot-plug hard drives, hot-plug redundant power, dual SD modules, ECC memory, interactive LCD screen
Fuente de Alimentación	<ul style="list-style-type: none"> ▪ Una fuente de alimentación de 1100W no redundante conectable en caliente ▪ Dos fuentes de alimentación 1100W redundantes conectables en caliente
Video	Matrox G200eW w/ 8MB memory
Soporte de Rack	ReadyRail sliding rails 4-post racks
Sistema Operativo	Debian 7.0 Wheezy

Tabla 10-9 Especificaciones Técnicas de Servidor

Requerimientos Mínimos de Máquina Cliente

A continuación en la Tabla 10-10 se presentan los requerimientos mínimos necesarios de la máquina cliente, con los cuáles el sistema podrá funcionar adecuadamente:

Característica	Especificación Técnica
Procesador	Intel® Pentium 4, 1.0GHz
Memoria RAM Libre Necesaria	128 MB
Espacio Disco Duro Libre Necesario	1 GB
Conexiones de Red	Tarjeta de Red Ethernet a 10 Gbit/s Tarjeta Inalámbrica 802.11b
Tipo de Equipo	Desktop

Tabla 10-10 Requerimientos Mínimos de Máquina Cliente



11 Capítulo V

Diseño

El diseño, es un proceso creativo en el que se define la arquitectura de hardware, software, componentes, módulos y datos de un Sistema Informático, que permiten satisfacer los requerimientos. La realización de un buen diseño brinda el beneficio de una mejor comprensión del sistema por parte de los desarrolladores, permitiendo una mayor eficiencia al tener un plano de lo que se realizará, reduciendo así tiempo y costo de desarrollo.

En este capítulo, se presenta el diseño de los diferentes elementos del Sistema Informático de Mantenimiento (SIM). El Diseño de Estándares, define los estándares a seguir para la presentación de contenido, de salida, de documentación, de base de datos, de mensajes, de código, entre otros. El Contexto del Sistema, presenta las relaciones entre el Sistema y su entorno desde diferentes perspectivas complementarias entre sí. El Diseño Arquitectónico es un elemento importante que permite la organización del sistema con sus subsistemas o componentes, de manera que satisfaga los requisitos funcionales y no funcionales. Y finalmente, además de los elementos antes mencionados, la elaboración del diseño también involucra el diseño de elementos de seguridad, establecimiento pautas que garanticen la fiabilidad del sistema.



11.1 Diseño de Estándares.

11.1.1 Estándar de Estructura de Página

La estructura de una página web se refiere a la ubicación de los elementos que la componen, tales como: menús de navegación, enlaces a otras páginas web y dónde se posicionará el contenido que podrán visualizar los usuarios. Para facilitar al equipo de trabajo el desarrollo de las páginas web se ha definido el estándar que se observa en la Figura 11-1, lo que a la vez permitirá una interfaz de usuario amigable y fácil de utilizar, manteniendo esta estructura y diseño durante toda la navegación por el sitio web.



Figura 11-1 Estándar de Estructura de Página

En el Cuadro 11-1 se describen los elementos del Estándar de Estructura de Página (Plantilla) presentado en la Figura 11-1:



No.	Elemento	Descripción
1	Banner del Sitio Web.	El Banner que estará presente en todas las páginas web del sistema, el cual estará compuesto por: <ul style="list-style-type: none"> ▪ Logo del escudo nacional ubicado en el extremo izquierdo superior de la página web. ▪ Nombre de la Institución, en este caso MINISTERIO DE SALUD, ubicado en el centro superior de la página web. ▪ Logo del MINSAL ubicado en el extremo derecho superior de la página web. ▪ El tamaño del Banner será 750px x 92px. ▪ El formato de la imagen del Banner debe ser PNG.
2	Sesión	Presenta la información del usuario que ha iniciado sesión en el sistema y estará compuesto por los siguientes elementos: <ul style="list-style-type: none"> ▪ Nombre de usuario con el cual se ha accedido al sistema. ▪ Fecha actual en la que se ha accedido al sistema ▪ Opción de cerrar la sesión. Estos datos estarán ubicados en el extremo derecho superior de la página web contiguo al Banner del sitio web.
3	Menú Principal	El menú principal del sistema poseerá las siguientes características: <ul style="list-style-type: none"> ▪ Las macro opciones del sistema, las cuales podrán variar en función del usuario que haya iniciado sesión. ▪ Se desplegará verticalmente en dirección descendente.
4	Árbol de Navegación	Elemento que permite al usuario conocer en que sección del sitio web se encuentra ubicado, brindándole la facilidad de retornar a una sección anterior si se desea. Este elemento se ubica en la parte inferior del menú principal.
5	Acciones Rápidas	Sección que contendrá diversos botones y parámetros que permitirán realizar acciones rápidas en función de la opción en la que se encuentre el usuario.
6	Área de Trabajo	El área de trabajo estará ubicada en el centro de la página web. En esta área estarán los formularios de captura de datos y las consultas al sistema o reportes de información que se presentarán al usuario.
7	Botones de Acción	Los botones de acción pueden ser: Agregar o crear, modificar, eliminar y confirmar. Estarán ubicados en la parte central inferior de la página web debajo del área de trabajo.
8	Pie de Página	Contendrá el nombre las partes que apoyaron el desarrollo del Sistema Informático. <ul style="list-style-type: none"> ▪ DTIC ▪ EISI – UES. Estará ubicado en la parte central inferior de la página web.

Cuadro 11-1 Descripción de los Elementos de la Estructura de Página



11.1.2 Estándares de Presentación de Contenido.

Los estándares de presentación de contenido permite el establecimiento de referencias de diseño y desarrollo para los diferentes elementos que conforman una vista (interfaz de usuario) dentro del Sitio Web.

Estándar Tipográfico.

La tipografía hace referencia al tipo de letra que se utilizará y sus propiedades. Las especificaciones que se deberán seguir en todas las páginas web durante su desarrollo, se han establecido mediante el siguiente estándar tipográfico, que se observa en el Cuadro 11-2:

Elemento	Estándar
Tipo de letra	Helvética
Tamaño	13px
Color	Negro RGB(0,0,0)
Background-color:	#FFFFFF

Cuadro 11-2 Estándar Tipográfico

Estándar de formularios web.

Los formularios web se utilizan para interactuar con el usuario de un modo más personalizado y obtener datos proporcionados por él. Para los elementos que componen un formulario se han definido los siguientes estándares:

- **Campo de Texto**

Campo de texto

Es un cuadro que permite introducir texto, al cual se le puede especificar su longitud de caracteres a presentar, longitud máxima de caracteres a introducir y texto que aparece al inicio (cuando el cuadro de texto se encuentra vacío), además puede ser definido como cuadro de texto para contraseña. Además, poseerá las propiedades descritas en la Tabla 11-1:



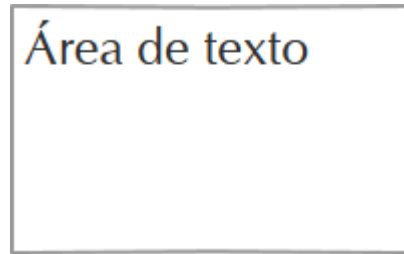
Elemento	Sub-elemento	Descripción
Campo de Texto No Requerido	Borde	Grosor: 1px. Estilo: Sólido. Color: HTML: # CCCCCC RGB(204,204,204) 
Campo de Texto Requerido	Borde	Grosor: 1px. Estilo: Sólido. Color: HTML: # E9322D RGB(233,50,45) 

Tabla 11-1 Estándar de Campo de Texto



- **Área de Texto**



Cuadro que permite introducir una cantidad mayor de texto comparado con el campo de texto. Se puede especificar longitud en caracteres de la línea, número de caracteres por línea, texto o valor inicial. Además, poseerá las propiedades descritas en la Tabla 11-2:


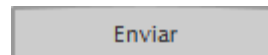
Elemento	Sub-elemento	Descripción
Campo de Texto No Requerido	Borde	Grosor: 1px. Estilo: Sólido. Color: HTML: # CCCCCC RGB(204,204,204) 

Tabla 11-2 Estándar de Área de Texto

- **Botón**



Permite la ejecución de ciertas acciones dentro del Sitio, entre las cuales se encuentra: crear, actualizar, filtrar información, comprobar datos como paso previo, entre otros. Los botones utilizados poseerán alguna de las propiedades descritas en la Tabla 11-3:







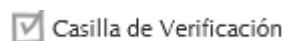
Elemento	Sub-elemento	Descripción
Botón : Crear, Actualizar o Filtrar.	Color de fondo	Color: HTML: #006DCC RGB(0,109,204) 
	Color de letra	Color: HTML: #FFFFFF RGB(255,255,255) 
Botón: Eliminar o que denota una acción crítica.	Color de fondo	Color: HTML: # BD362F RGB(189,54,47) 
	Color de letra	Color: HTML: #FFFFFF RGB(255,255,255) 
Botón: Botón de éxito.	Color de fondo	Color: HTML: # 51A351 RGB(98,196,98) 
	Color de letra	Color: HTML: #FFFFFF RGB(255,255,255) 

Tabla 11-3 Estándar de Botón

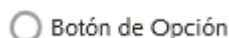


- **Casilla de Verificación**



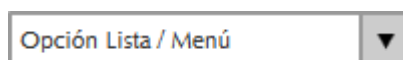
Pequeño recuadro que se puede marcar o desmarcar para seleccionar alguna característica o propiedad.

- **Botón de Opción**



Pequeño círculo que se puede marcar o desmarcar (siempre tiene que existir un círculo de la lista marcado) para seleccionar una determinada opción.

- **Lista Desplegable**



Listado de opciones, del cual puede ser seleccionada una opción. El usuario puede desplegar las opciones haciendo clic sobre el listado. Deberá poseer las propiedades descritas en la Tabla 11-4:


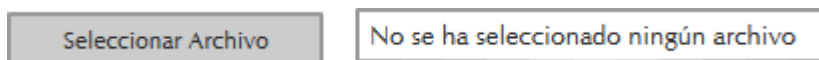
Elemento	Sub-elemento	Descripción
Lista / Menú	Borde	Grosor: 1px. Estilo: Sólido. Color: HTML: # CCCCCC RGB(204,204,204) 

Tabla 11-4 Estándar de Lista / Menú

- **Campo de Archivo o Imagen.**



Permite la funcionalidad de seleccionar un archivo o imagen que se encuentre en la computadora para ser subido al servidor y ser utilizado en el sitio web. Deberá tener las propiedades especificadas en la Tabla 11-5:


Elemento	Sub-elemento	Descripción
Campo de Archivo o Imagen	Borde	Grosor: 1px. Estilo: Sólido. Color: HTML: # CCCCCC RGB(204,204,204) 

Tabla 11-5 Estándar de Campo de Archivo o Imagen



11.1.3 Estándares de Salida.

Son el conjunto de normas que se establecen para garantizar la homogeneidad y acoplamiento de los documentos de salida, que permita el establecimiento de un nivel de calidad en la elaboración de éstos.

Para el desarrollo de los documentos de salida que brindará el SIM, se han establecido los siguientes estándares de salida (ver Cuadro 11-3), que comprende la elaboración de reportes, así como también la elaboración de documentos legales.

Elemento	Sub-elemento	Descripción
Estándares de Página.	Tipo	Papel Bond tamaño Carta
	Ancho	21.59 cm
	Alto	27.94 cm
	Margen	Superior: 0.64 cm Derecho: 0.64 cm Inferior: 0.64 cm Izquierdo: 0.64 cm
	Orientación	Vertical, Horizontal.
Estándares de Letra.	Titulo 1	Tipo de Letra: Sans Serif Tamaño: 16px Color: RGB (0,0,0). Negrita.
	Titulo 2	Tipo de Letra: Sans Serif Tamaño: 14px Color: RGB (0,0,0). Negrita.
	Titulo 3	Tipo de Letra: Sans Serif Tamaño: 12px Color: RGB (0,0,0).
	Texto Independiente	Tipo de Letra: Sans Serif Tamaño: 10px Color: RGB (0,0,0).
Estándares de Logo.	Posición	Superior: 0 cm Derecho: 0 cm
	Tamaño	Ancho: 4.37 cm Alto: 2.99 cm 124px 85px
	Formato de Imagen.	Portable Network Graphics (PNG)
Estándares de Tabla	Borde	Grosor de línea: 1pto. Color de Línea: RGB (153,153,153). Estilo de Línea: Sólido.
	Color de Relleno	RGB (219,229,241) Solamente para Encabezados de Tablas.

Cuadro 11-3 Estándares de Documentos de Salida.



Los siguientes documentos son considerados como “documentos de salida”:

- Documento de Reporte: Documento que contiene datos tabulados, y muestra datos detallados del reporte generado.
- Documento de propósito legal: Documento que contiene información formal. Algunos ejemplos de estos documentos son: ordenes de trabajo, solicitudes de servicios, etc.

En la figuras Figura 11-2, Figura 11-3 y Figura 11-4 se muestran un ejemplo de la estructura de un reporte y documento de propósito legal en el cual se ha aplicado los estándares establecidos en el Cuadro 11-3.



Estructura de un Documento Legal.

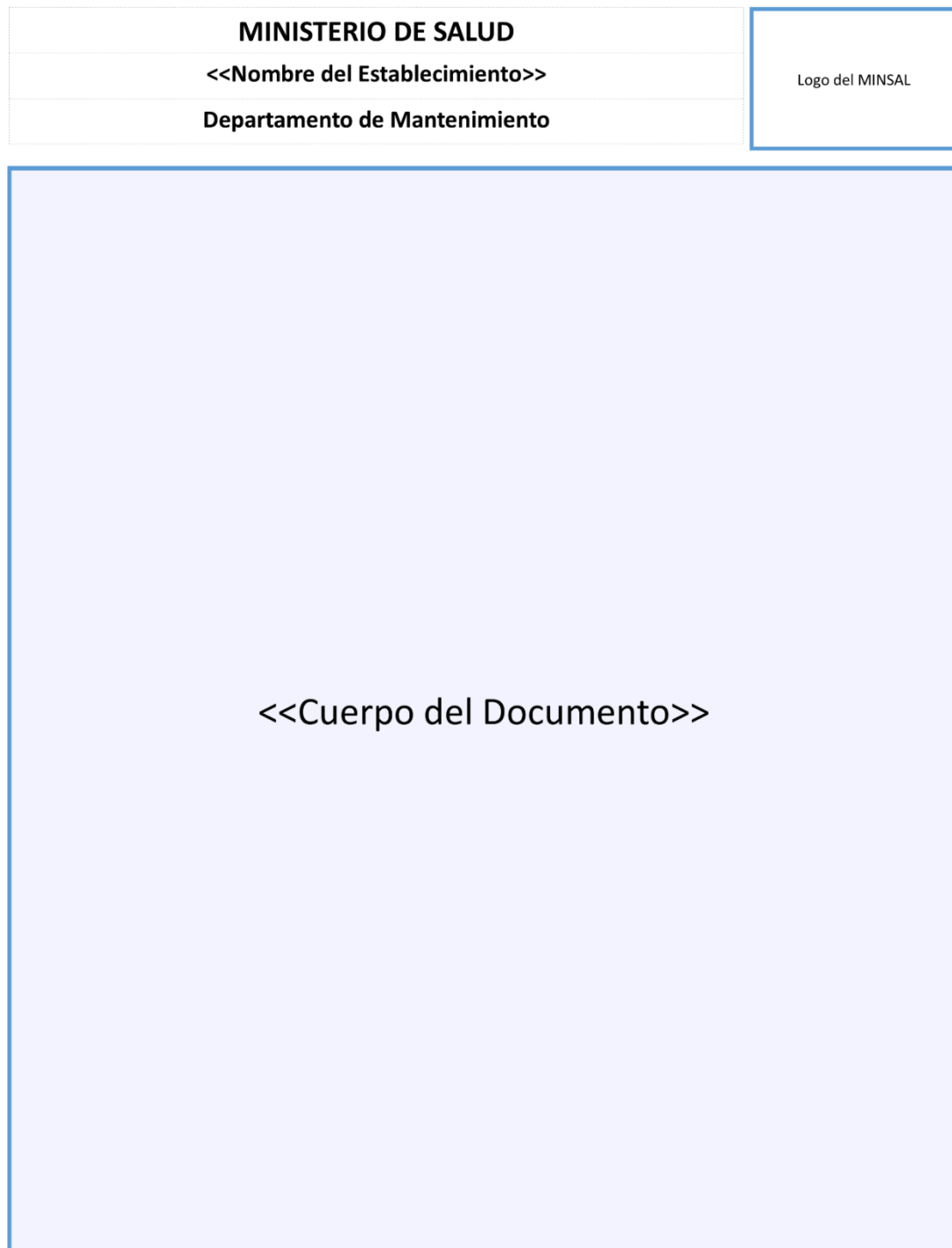


Figura 11-4 Estructura de un Documento Legal.



11.1.4 Estándares de Documentación Interna.

11.1.4.1 Estándares de Documentación Interna del Código Fuente

Todo sistema informático posee documentación interna la cual facilita el desarrollo y el mantenimiento de los sistemas. La documentación interna de código se especifica mediante comentarios que se encuentran junto con el código fuente y esta se incluye en archivos de: hojas de estilo, archivos JavaScript, vistas, plantillas, controladores, modelos y clases. La información que se debe documentar es la siguiente:

- a. **Encabezado:** Cada archivo de código fuente deberá poseer un encabezado (especialmente aquellos denominados "controladores") en donde se debe especificar la siguiente información:
 - Nombre del modulo
 - Dirección física del archivo.
 - Objetivo.

En el Cuadro 11-4 se presenta un ejemplo de la estructura del encabezado anteriormente descrito:

Estructura	Ejemplo
<pre>/* *Nombre del módulo: *Dirección física: *Objetivo: **/</pre>	<pre>/* * Nombre del módulo: Gestión de departamentos. * Dirección física: src/Minsal/SimBundle/Entity/CtlDepartamento.php * Objetivo: Clase que permite realizar la persistencia con la tabla * ctl_departamento en la base de datos. **/</pre>

Cuadro 11-4. Ejemplo de encabezado para documentación de archivos.

- b. **Límites de sección:** Se especificarán los límites de cada sección importante dentro del código fuente de cada archivo, estas secciones incluyen: módulos, funciones, definición de variables o estructuras de datos, iteraciones (loops), algoritmos especiales y otras que el desarrollador considere necesarias. En el Cuadro 11-5 se presentan ejemplos de este estándar:

Lenguaje	Ejemplo
HTML	<pre><!--Inicio de Modulo o Función--> Línea de Código 1 //... Línea de Código N <!--Fin de Modulo o Función--></pre>
PHP	<pre>/*Inicio de Modulo o función*/ Línea de Código 1 //... Línea de Código N /*Fin de Modulo o Función */</pre>

Cuadro 11-5. Ejemplo documentación de límites de sección.



- c. **Objetivos de Funciones o Módulos:** En el caso de las funciones, módulos, iteraciones (loops) u otros algoritmos especiales, se deberá especificar el objetivo o una breve descripción de la funcionalidad de la misma, como se muestra en el Cuadro 11-6:

Lenguaje	Ejemplo
PHP	<pre> /* Inicio de Modulo o función *Objetivo: calcular la próxima fecha de mantenimiento*/ public function calcularFecha() { Línea de Código 1 ... Línea de Código N } /*Fin de Modulo o Función*/ </pre>

Cuadro 11-6. Ejemplo de documentación de objetivos de funciones o módulos.

- d. **Otros comentarios:** Comentarios que el desarrollador considere necesarios para el correcto entendimiento del código fuente para otros desarrolladores, los cuales puede incluir: comentarios de variables, parámetros de funciones (recibidos y retornados), entre otros. Se puede observar un ejemplo de ello, en el Cuadro 11-7:

Lenguaje	Ejemplo
PHP	<pre> \$variable = valor; //almacena el valor de... Línea de Código 1 ... Línea de Código N //registra el nuevo cambio </pre>

Cuadro 11-7. Ejemplo de documentación de otros comentarios.

Para cada lenguaje, la sintaxis de comentarios o documentación es diferente, la cual se especifica en el Cuadro 11-8:

Lenguaje de programación	Comentario
PHP	<pre> /* Este es un comentario * de más de una línea para php*/ // Este es un comentario de una línea para php. </pre>
vistas y plantillas Twig	{# Este es un comentario para las vistas y plantillas Twig #}
CSS	/* Este es un comentario para hojas de estilo */
Javascript	<pre> /* Este es un comentario de varias líneas para javascript */ // Este es un comentario de línea para javascript </pre>
XML	<!-- Este es un comentario en lenguaje de etiquetado -->
HMTL	<!-- Este es un comentario en HTML -->
PostgreSQL (Scripts SQL)	<pre> -- Este es el comentario estándar SQL. /* Este es un comentario de varias líneas SQL */ </pre>

Cuadro 11-8. Sintaxis de comentarios según lenguajes de programación.



11.1.4.2 Estándares de Documentación Interna de los Objetos de la Base de Datos.

La base de datos, específicamente los objetos dentro de ella, no son ajenos a la documentación que facilite la comprensión de los mismos o conocer su objetivo, funcionalidad y utilidad, y tal como se mencionó en la sección anterior, facilita el mantenimiento futuro por parte de otros desarrolladores.

Se documentaran los objetos de la base de datos, realizando un comentario para los mismos, por ejemplo para tablas, campos, procedimientos, triggers o cualquier elemento que contenga la base de datos. La documentación debe de ser descriptiva y no redundante, mediante el comando “COMMENT ON”, tal como se muestra en el *Cuadro 11-9*:

Comando SQL	Ejemplo
COMMENT ON Tipo_de_Objeto Nombre_del_Objeto IS 'Comentario'	COMMENT ON COLUMN my_table.my_column IS 'Representa el ID oCodigo del Empleado'; COMMENT ON SEQUENCE my_sequence IS 'Secuencia utilizada para generar el Codigo de los Empleados';

Cuadro 11-9. Ejemplo de Documentación de Objetos de la Base de Datos.

11.1.5 Estándares de Documentación Externa.

Para tener un marco de referencia y control sobre la edición de cada uno de los documentos correspondientes, se ha establecido el siguiente estándar de documentos, descrito en el Cuadro 11-10 el cual establece los estándares correspondientes a los diferentes manuales que se han de elaborar.

Elemento	Subelemento	Tipo	Valor
Página	Tamaño	Ancho	21.59 cm
		Alto	27.94 cm
	Margen	Inferior, Superior	2.0 cm
		Izquierdo	3.0 cm
	Derecho	1.8 cm	
Capítulo	Titulo 1	Tipo de Letra	Century Gothic
		Tamaño	36
		Color	RGB(23,54,93)
	Titulo 2	Tipo de Letra	CG Omega
		Tamaño	26
		Color	RGB(0,0,0)
	Texto	Tipo de Letra	CG Omega
		Tamaño	10
		Color	RGB(0,0,0)



Elemento	Subelemento	Tipo	Valor
Documento	Titulo 1	Tipo de letra	Century Gothic
		Tamaño	18
		Color	RGB(0,0,0)
		Interlineado	Múltiple 1.15
		Espaciado	Superior - 24, Inferior - 0
	Titulo 2	Tipo de letra	CG Omega
		Tamaño	15
		Color	RGB(0,0,0)
		Interlineado	Múltiple 1.15
		Espaciado	Superior - 2, Inferior - 0
	Titulo 3	Tipo de letra	CG Omega
		Tamaño	13
		Color	RGB(0,0,0)
		Interlineado	Múltiple 1.15
		Espaciado	Superior - 10, Inferior - 0
	Titulo 4	Tipo de letra	CG Omega
		Tamaño	12
		Color	RGB(0,0,0)
		Interlineado	Múltiple 1.15
		Espaciado	Superior - 2, Inferior - 0
Texto	Tipo de letra	CG Omega	
	Tamaño	11	
	Color	RGB(0,0,0)	
	Interlineado	Múltiple 1.15	
	Espaciado	Superior - 0, Inferior - 10	

Cuadro 11-10. Estándares de documentación externa.

Estándares de tablas.

Los estándares para el formato de las tablas ayudaran a tener una mejor visualización de los datos. Así también, en mantener un mismo formato para toda la documentación externa que ha de elaborarse. Este se describe en el Cuadro 11-11:

Elemento	Detalle
Encabezado	Negrita
Tamaño de letra	11 pto
Tipo de Letra	CG Omega
Relleno encabezado	RGB(219,229,241)
Color de Línea	RGB(149,179,215)
Grosor de Línea	1 pto

Cuadro 11-11. Estándares para tablas



11.1.6 Estándares de Base de Datos.

Al realizar un diseño de base de datos es necesario establecer estándares, con los cuales se pretende formalizar la estructura para cada uno de los objetos que forman parte de la base de datos. Los siguientes estándares, son también requerimientos de desarrollo del sistema SIM, por lo cual deben ser respetados a cabalidad (ver Cuadro 11-12):

Estándares para el Diseño de la Base de Datos	
Consideraciones Generales	Ejemplo
<p>Deben tomarse en cuenta las siguientes consideraciones:</p> <ul style="list-style-type: none"> • No usar espacios en blanco para los nombres de los objetos. • No utilizar letras mayúsculas, ni tildes tampoco caracteres especiales. • Usar guion bajo (_) para separar las palabras. 	
Nombre de la Base de Datos	
<p>El nombre usado para la base de datos debe ser definido usando un nombre corto y se recomienda usar el nombre del proyecto todo en minúsculas.</p>	sim
Nombres de Tablas	
<p>Durante el nombramiento de las tablas de la base de datos se deben tomar en cuenta las siguientes recomendaciones:</p> <ul style="list-style-type: none"> • Nombres auto-descriptivos: Los nombres utilizados para definir las tablas deben de ser en lo posible auto-descriptivos, y evitar el uso de abreviaciones. Se recomienda ampliamente el uso de nombres cortos. • Nombres en singular: Los nombres de las tablas deben de ser definidos en singular. En el caso de nombres de tablas con más de una palabra ambas deben estar en singular separadas por un guion bajo. • Prefijos: Dentro de la base de datos pueden existir ciertas tablas que necesiten agruparse según la lógica del negocio. En estos casos puede ponerse un prefijo de entre dos a tres caracteres para identificar el grupo o modulo al que pertenecen. • Tablas de referencia: Los nombres de las tablas que son usadas para guardar referencias (many to many) deben de definirse en lo posible usando el nombre de ambas tablas. • Tablas catálogos: el nombre estará compuesto de la siguiente forma: ctl_nombre_catalogo. 	<p>Nombre de tabla: orden_trabajo, ctl_establecimiento ctl_pais.</p> <p>Nombre de tabla de referencia: ctl_especialidad_ctl_establecimiento</p>



Nombres de Campos	
<p>Las columnas son atributos de la entidad, es decir, las columnas describen las propiedades de la entidad por tanto los nombres de las columnas deben ser representativos y naturales.</p> <p>El campo que será llave primaria debe llamarse id y debe de ser autoincrementable. Dicho estándar es necesario ya que el Paquete de Doctrine lo requiere de esta forma como llave primaria.</p> <p>Si es una tabla de referencia colocar las primeras tres letras de la primera tabla, guion bajo (_), las primeras tres letras de la segunda tabla y el nombre del campo.</p> <p>El nombre de la columna que contenga una llave foránea hacia otra tabla dentro de la misma base de datos, deber de ser la siguiente: id_nombre_tabla_padre sin el uso de prefijo.</p>	
Usuario de la Base de Datos	
<p>Se debe crear un usuario dueño de la base de datos sin permisos de creación de base de datos, ni roles, ni ser súper usuario con permisos solo sobre la base y sus objetos relaciones.</p>	<pre>createuser -DRSP nombre_usuario</pre> <p>D: No puede crear base de datos R: No puede crear roles S: No es superusuario P: Para que muestre un mensaje en donde se pida la contraseña del nuevo usuario.</p> <pre>createdb nombre_base -O nombre_usuario</pre> <p>O: Dueño de la base de datos</p>
Llaves Primarias	
<p>Las llaves primarias es la columna que identifica cada columna en cada tabla. Para nombrarlas se utilizará:</p> <p style="text-align: center;">pk_nombre_tabla</p> <p>Nota: Si se trabajara con Symfony2 y Doctrine2 no se pueden realizar llaves primarias compuestas; por ende crear un campo auto-numérico como llave primaria y se debe crear un índice para que los dos campos sean únicos.</p>	<p>pk_pais</p>
Llaves Foráneas	
<p>Por convención estas pueden nombrarse de la siguiente forma:</p> <p style="text-align: center;">fk_nombre_tabla_padre_nombre_tabla_hija</p> <p>Si las tablas padre e hija poseen prefijo, cuando se nombre la llave foránea debe ir sin el prefijo que posee.</p>	<p>fk_asignatura_estudiante</p>
Vistas	
<p>Se nombran con un nombre significativo de la información que contienen utilizando el prefijo “vw_”</p>	<p>vw_estudiante</p>



Procedimientos	
Los procedimientos pueden nombrarse de acuerdo a lo que ellos realizan dentro de la base de datos; puede utilizarse un verbo en infinitivo para describir dicha acción. Deben comenzar con el prefijo “proc_”.	proc_calcularnota_asignatura
Triggers o Disparadores	
<p>Durante el nombrado de los “triggers” se deben tomar en cuenta las siguientes recomendaciones:</p> <ul style="list-style-type: none"> • Prefijos: se usará prefijo “trg”, que es ampliamente recomendado con el propósito de hacer saber que el objeto en la base de datos es un trigger. • Nomenclatura: son procedimientos especiales que se ejecutan luego de una determinada acción en una determinada tabla. Para nombrarlos, al igual que los procedimientos se puede utilizar un verbo en infinitivo precedido del nombre de la tabla a la que pertenece. Por ejemplo: <p style="text-align: center;">trg_accion_tabla</p> <p>Excepciones: las funciones a las que hacen referencias los triggers se deben llamar con el mismo nombre del trigger agregando el prefijo fn nombre_trigger.</p>	tgr_actualizar_transaccion
Índices	
<p>Los índices solo existen para una determinada tabla, para uno o más campos separados por guion bajo (_). Para nombrarse puede ser de la siguiente manera:</p> <p style="text-align: center;">idx_nombre_campo(s)</p> <p>Para el nombre del índice se deben colocar máximo 2 campos.</p>	idx_numero_orden
Documentación Interna	
Se debe realizar un comentario para cada campo, tablas, procedimientos, triggers o cualquier elemento que contenga la base de datos; esto con el fin de tener claros las funcionalidades de los elementos que constituyen la base de datos. La documentación debe de ser descriptiva y no redundante.	/* ctl_departamentos, este es un catálogo de los departamentos de El Salvador */
Catálogos Generales	
Dentro del MINSAL se utilizan una serie de catálogos generales los cuales deben de tener la estructura que se presentara en un archivo adjunto denominado catalogo.png	

Cuadro 11-12. Estándares de Base de Datos.



11.1.7 Estándares de Mensajes.

Los estándares de mensajes son un conjunto de reglas establecidas que permitan la facilidad de visualización y comprensión de los mensajes que brindará el sistema, facilitando la asociación de estos con su significado.

Para la presentación de mensajes que serán mostrados en el SIM, se han establecido los siguientes estándares, los cuales comprenden los mensajes de éxito, error, y confirmación.

Mensajes de Error.

Los mensajes de error son aquellos que se generan cuando se produce un evento incorrecto en el sistema. En la Figura 11-5 se muestra un ejemplo de estándar establecido para los mensajes de error que han de ser implementado en el SIM.

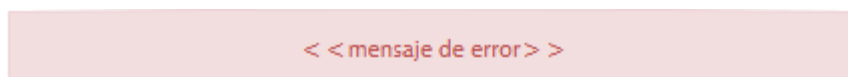


Figura 11-5 Ejemplo de Mensaje de Error.

El ejemplo de la Figura 11-5 ha sido creada siguiendo los estándares mostrados en la Tabla 11-6.




Elemento	Descripción
Color de Fondo	HTML: #F2DEDE. RGB (242,222,222) 
Borde	Grosor: 1px. Estilo: Sólido. Color: HTML: #EED3D7 RGB(238,211,215) 
Letra	Alineación: Centrada. Tipo: "Helvetica Neue",Helvetica,Arial,sans-serif. Tamaño: 13px. Color: HTML: #B94A48 RGB(185,74,72) 

Tabla 11-6 Estándar de Mensaje de Error

Mensajes de Éxito.

Los mensajes de éxitos son generados a partir de eventos que se han realizado correctamente y que no han generado algún error en su proceso, como se muestra e la Figura 11-6.

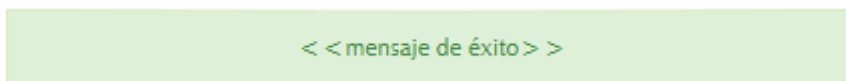


Figura 11-6 Ejemplo de Mensaje de Éxito.

El ejemplo de la Figura 11-6 ha sido creado en base al estándar establecido en la Tabla 11-7.




Elemento	Descripción
Color de Fondo	HTML: # DFF0D8. RGB (223,240,216) 
Borde	Grosor: 1px. Estilo: Sólido. Color: HTML: # D6E9C6 RGB(214,233,198) 
Letra	Alineación: Centrada. Tipo: "Helvetica Neue",Helvetica,Arial,sans-serif. Tamaño: 13px. Color: HTML: # 468847 RGB(70,136,71) 

Tabla 11-7 Estándar de Mensaje de Éxito.

Mensajes de Confirmación.

Los mensajes de confirmación son generados a partir de eventos que requieren que el usuario ratifique la acción a realizar, normalmente este tipo de mensajes son generados por eventos de actualización o eliminación de datos en el sistema.

Titulo Mensaje Confirmación

Descripción del mensaje de confirmación - Cuerpo del Mensaje



Figura 11-7 Ejemplo de Mensaje de Confirmación.

El mensaje mostrado en la Figura 11-7, está estructurado de la siguiente manera:

- Título del mensaje de confirmación.
- Cuerpo del mensaje de confirmación, en el que se presenta un mensaje más descriptivo de la acción a realizar.
- Botones de acción, en donde el usuario puede continuar o cancelar la acción.

El mensaje presentado anteriormente ha sido estructurado en base al estándar establecido en la Tabla 11-8.









Elemento	Sub-elemento	Descripción
Título del mensaje	Letra	Alineación: Centrado. Tipo: "Helvetica Neue",Helvetica,Arial,sans-serif. Tamaño: 36px. Color: HTML: # 000000.  RGB (0,0,0) Negrita.
Descripción del mensaje	Letra	Alineación: Centrado. Tipo: "Helvetica Neue",Helvetica,Arial,sans-serif. Tamaño: 13px. Color: HTML: # 404040.  RGB (64,64,64)
Botón de confirmación.	Letra	Alineación: Centrado. Tipo: "Helvetica Neue",Helvetica,Arial,sans-serif. Tamaño: 14px. Color: HTML: # FFFFFFF  RGB(255,255,255)
	Fondo	Color: HTML: # DA4F49  RGB(218,79,73)
Botón de Cancelación	Letra	Alineación: Centrado. Tipo: "Helvetica Neue",Helvetica,Arial,sans-serif. Tamaño: 14px. Color: HTML: # FFFFFFF  RGB(255,255,255)
	Fondo	Color: HTML: # 5BB75B  RGB(91,183,91)

Tabla 11-8 Estándar de Mensaje de Confirmación.



11.1.8 Estándares de Código.

Para una mejor comprensión y mantenimiento del código fuente se establecen los siguientes estándares de código.

Estándar de Código Symfony 2.

Los estándares seguidos en la Tabla 11-9, han sido tomados de los documentos PSR-0⁵⁹, PSR-1⁶⁰ y PRS-2⁶¹, los cuales son estándares de código PHP, que symfony 2 ha adaptado en su código.

Estándar de Codificación	Código de Ejemplo
Archivos	
Los archivos deben usar únicamente las etiquetas <code><?php</code> o las etiquetas <code><?=.</code> La etiqueta de cierre <code>?></code> debe de ser omitida en archivos que contienen solo código PHP.	<pre><?php namespace Acme\DemoBundle\Controller; use Symfony\Bundle\FrameworkBundle\... class WelcomeController extends Controller { public function indexAction() { } }</pre>
Los archivos sólo deben utilizar una codificación UTF-8.	<pre><!DOCTYPE html> <html> <head> <meta charset="UTF-8" /></pre>
Definir una clase por archivo.	
Utilizar caracteres alfanuméricos y guiones bajos para nombres de archivos.	DemoController.php SecuredController.php AcmeDemoExtension.php
Todos los archivos deben de tener extensión .php.	
Nombre de Clases y Namespace	
Los nombres de clases deben ser escritas utilizando la técnica UpperCamelCase .	<pre><?php namespace Vendor\Model; class FooBar { }</pre>
Las clases controladores siempre llevarán el sufijo Controller .	<pre><?php namespace Acme\DemoBundle\Controller; class WelcomeController extends Controller { }</pre>

⁵⁹ PHP Framework Interop Group (25 de junio del 2013). *Autoloading Standard*. Obtenido de: <http://www.php-fig.org/psr/0/>

⁶⁰ PHP Framework Interop Group (25 de junio del 2013). *Basic Coding Standard*. Obtenido de: <http://www.php-fig.org/psr/1/>

⁶¹ PHP Framework Interop Group (25 de junio del 2013). *Coding Style Guide*. Obtenido de: <http://www.php-fig.org/psr/2/>



<p>Debe haber una línea en blanco después de la declaración de namespace.</p>	<pre><?php namespace Acme\DemoBundle\Controller; use Symfony\Bundle\FrameworkBundle\Controller\Controller; use Symfony\Component\HttpFoundation\RedirectResponse; use Acme\DemoBundle\Form\ContactType; // Importa "@Route" y "@Template" annotations use Sensio\Bundle\FrameworkExtraBundle\Configuration\Route; use Sensio\Bundle\FrameworkExtraBundle\Configuration\Template; class DemoController extends Controller { /** * @Route("/", name="_demo") * @Template() */ }</pre>
<p>Los namespace y las clases deben tener la siguiente estructura: \<VendorName>\(<Namespace>\)* <Class Name></p>	
<p>Los nombres de los namespaces o clases deben ser ordenadas alfabéticamente.</p>	
<p>Debe usarse la palabra clave use por cada declaración y todas las declaraciones de use deben ir después de la declaración de namespace.</p>	
<p>Constantes, Métodos y Propiedades.</p>	
<p>Las constantes deben ser definidas en MAYÚSCULAS y utilizando guion bajo (_) cómo separador.</p>	<pre><?php namespace Vendor\Model; class Foo { const VERSION = '1.0'; const DATE_APPROVED = '2012-06-01'; }</pre>
<p>Métodos deben ser escritos utilizando la técnica camelCase.</p>	<pre>public function indexAction() { return array(); } public function helloAction(\$name) { return array('name' => \$name); }</pre>
<p>Las funciones o métodos de los controladores siempre llevarán el sufijo Action.</p>	
<p>Declarar las propiedades de clase antes que los métodos.</p>	<pre>class DemoController extends Controller { public atributo1; protected atributo2; private atributo3;</pre>
<p>No utilizar la palabra reservada var para declarar una propiedad, en su lugar utilizar public, private, static o protected.</p>	<pre>public function helloAction(\$name) { return array('name' => \$name); } }</pre>
<p>Declarar las propiedades y métodos públicos en primer lugar, a continuación, los protegidos y finalmente los privados.</p>	
<p>Líneas e Identación.</p>	
<p>La indentación debe ser con un tabulador establecido a 4 espacios.</p>	
<p>El número de caracteres por línea deben ser de 80 columnas aunque también esta aceptado que sean hasta 120.</p>	



Utiliza el caracter de salto de línea para finalizar las líneas.	
--	--

Tabla 11-9 Estándar de Código de Symfony 2.

Estándar de Código Twig.

En el desarrollo de plantillas con Twig, se tomarán los estándares de codificación oficiales de la página de Twig⁶². Para mejor comprensión de los ejemplos mostrados en la Tabla 11-11 se seguirá un estándar de colores el cual se muestra en la Tabla 11-10.

	Delimitadores
	Elementos dentro Delimitadores
	Comentarios
	Estructuras de Control
	Signos de Agrupación
	Cadenas entre comillas.

Tabla 11-10 Estándar de Color para Ejemplos

Existen 3 tipos de delimitadores en twig:

- `{{ }}`: Utilizada para imprimir el resultado de la evaluación de la expresión.
- `{% %}`: Utilizada para ejecutar declaraciones.
- `{# #}`: Comentarios

Estándar de Codificación	Código de Ejemplo
Delimitadores	
Dejar un espacio (y sólo uno) después de un delimitador inicial (<code>{{</code> , <code>{%</code> , y <code>{#</code>) y antes del final de un delimitador (<code>}}</code> , <code>%}</code> , y <code>#}</code>).	<pre>{{ foo }} {# comentario #} {% if foo %}{% endif %}</pre>
Cuando se utilice el caracter de guión junto con el espacio en blanco, no dejar ningún espacio entre este y el delimitador.	<pre>{{- foo -}} {#- comentario -#} {%- if foo -%}{%- endif -%}</pre>
Operadores	
Dejar un espacio (y sólo uno) antes y después de los siguientes operadores: <ul style="list-style-type: none"> • operadores de comparación (<code>==</code>, <code>!=</code>, <code><</code>, <code>></code>, <code>>=</code>, <code><=</code>) • operadores matemáticos (<code>+</code>, <code>-</code>, <code>/</code>, <code>*</code>, <code>%</code>, <code>//</code>, <code>**</code>) 	<pre>{{ 1 + 2 }} {{ foo ~ bar }} {{ true ? true : false }}</pre>

⁶² SensioLabs (26 de junio del 2013). *Coding Standards*. Obtenido de: http://twig.sensiolabs.org/doc/coding_standards.html



<ul style="list-style-type: none"> operadores lógicos (not, and, or), ~, is, in operador ternario (?:) 	
Plantilla Base	
<p>Se creará una plantilla base, con el layout o estructura estándar que se utilizará en todas las páginas web. La plantilla se reutilizará utilizando la capacidad de herencia que posee Twig.</p>	<pre>{% extends "base.html" %} {% block title %}Index{% endblock %} {% block head %} {{ parent() }} <style type="text/css"> .important { color: #336699; } </style> {% endblock %} {% block content %} <h1>Index</h1> <p class="important"> Welcome on my awesome homepage. </p> {% endblock %}</pre>
Nombre de variables	
<p>Utilizar letras minúsculas y guiones bajos en nombres de variables.</p>	<pre>{% set foo = 'foo' %} {% set foo_bar = 'foo' %}</pre>
Indentación y espacios en blanco	
<p>Indentar el código dentro de las etiquetas, utilizando la misma profundidad que la utilizada en el lenguaje principal del archivo.</p>	<pre>{% block foo %} {% if true %} true {% endif %} {% endblock %}</pre>
<p>Dejar un espacio (y sólo uno) después del signo dos puntos (:) en hashes e igual con la coma (,) en arreglos y hashes.</p>	<pre>{{ [1, 2, 3] }} {{ {'foo': 'bar'} }}</pre>
<p>No dejar ningún espacio después de un paréntesis de apertura y antes de un paréntesis de cierre en expresiones.</p>	<pre>{{ 1 + (2 * 3) }}</pre>
<p>No dejar ningún espacio en blanco antes y después de los delimitadores de cadena</p>	<pre>{{ 'foo' }} {{ "foo" }}</pre>
Indentación y espacios en blanco	
<p>No dejar ningún espacio en blanco antes y después de los siguientes operadores: , ., .., []</p>	<pre>{{ foo upper lower }} {{ user.name }} {{ user[name] }} {% for i in 1..12 %}{% endfor %}</pre>
<p>No dejar ningún espacio en blanco antes y después de los paréntesis utilizados en filtros y llamadas a funciones.</p>	<pre>{{ foo default('foo') }} {{ range(1..10) }}</pre>
<p>No dejar ningún espacio en blanco antes y después de la apertura de arreglos y hashes.</p>	<pre>{{ [1, 2, 3] }} {{ {'foo': 'bar'} }}</pre>

Tabla 11-11 Estándar de Código Twig



11.1.9 Estándares de Estructura de Navegación.

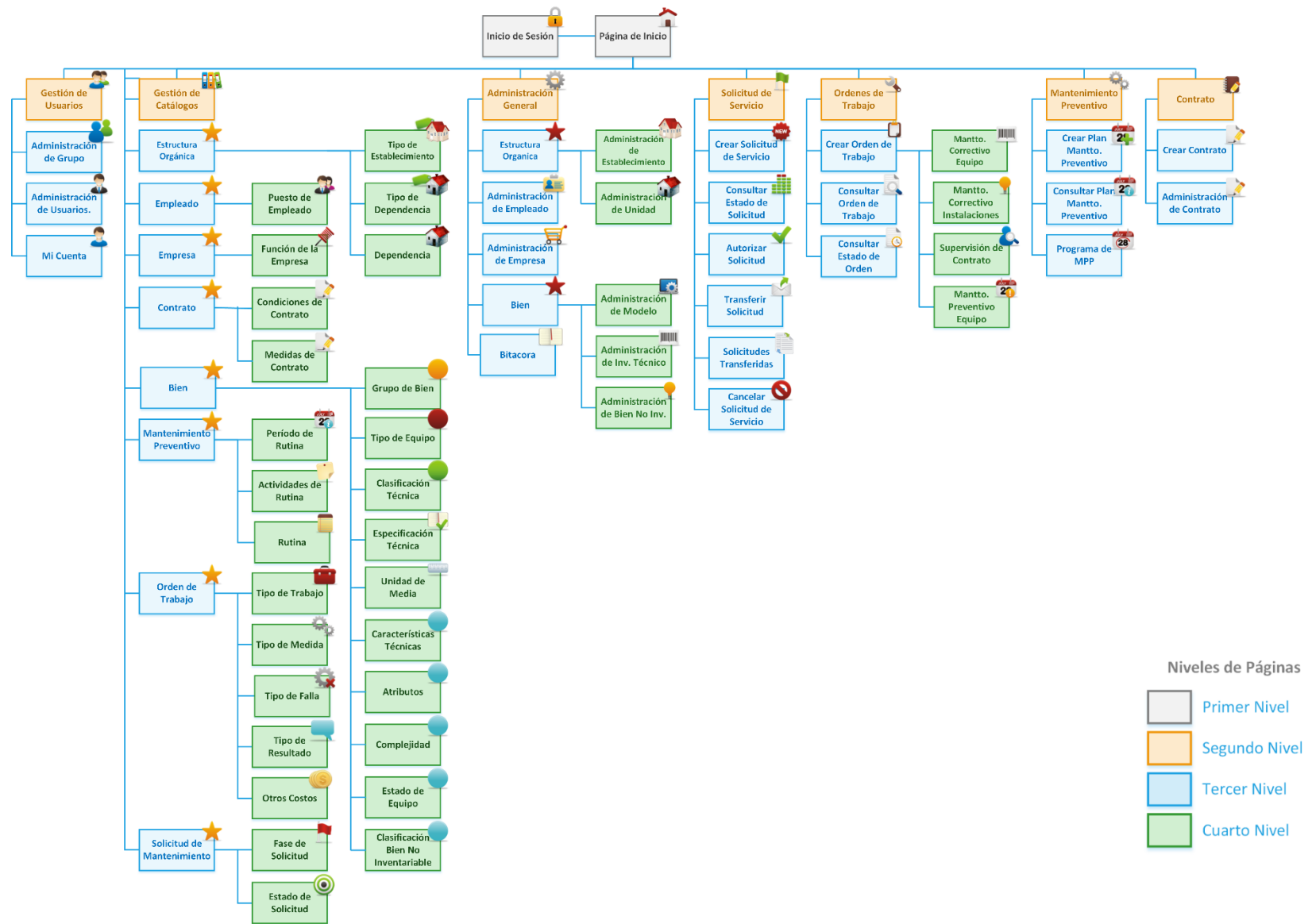


Figura 11-8 Mapa de Navegación del Sitio.



11.1.10 Estándares de Estructura de Archivos.

Estructura General de Archivos del Proyecto

De forma predeterminada, cada aplicación basada en Symfony tiene la misma estructura de archivos básica, de igual forma este será el estándar para la Estructura de Archivos del Sistema a desarrollar. La estructura básica se muestra en la Figura 11-9:

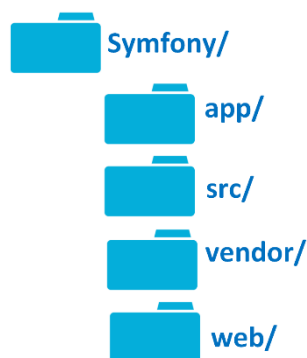


Figura 11-9. Estructura básica de archivos de Symfony.

- Directorio “**app**”: Contiene los archivos necesarios para configuración del Sitio Web, además contiene un directorio caché que permite que el acceso a la aplicación se realice de manera más rápida, un directorio logs en el que se registra los diferentes eventos que suceden dentro del sistema, y un directorio resources, que contiene los diferentes archivos (recursos) que pueden ser utilizados en todo el Sitio Web.
- Directorio “**src**”: Contiene los diferentes Bundles (ver definición en Glosario Técnico) que se hayan creado dentro del sistema, cada Bundle contiene el conjunto de directorios y archivos que almacenan la lógica del sistema.
- Directorio “**vendor**”: dentro de este directorio, se almacena las librerías a utilizar, que pertenecen a terceros.
- Directorio “**web**”: Contiene archivos que necesitan ser accedidos de forma pública, como por ejemplo: imágenes, hojas de estilo y archivos JavaScript. También es el lugar donde reside cada Controlador Frontal (ver definición en Glosario Técnico).

Estructura General de Bundles (Paquetes)

Un Bundle o paquete es un directorio que tiene una estructura bien definida y puede contener desde clases hasta controladores y recursos web. Los estándares a seguir para la creación y administración de Bundles son los siguientes:



Estándares de Nombres de Bundles:

Los estándares de nombres de Bundles se rigen según los estándares de espacio de nombres (namespace), el cual se puede entender como la forma lógica de encapsular los diferentes elementos de una clase (funciones, constantes), brinda la facilidad de evitar conflictos cuando se definen clases que poseen el mismo nombre de las clases internas de PHP, además sirven como enlaces (link) evitando colocar toda la dirección de la ruta física de la clase. El nombre de la clase Bundle debe seguir los siguientes estándares:

- Solamente se debe utilizar caracteres alfanuméricos y guion bajo.
- Utilizar nombres con UpperCase.
- Utilizar un nombre corto y descriptivo.
- El nombre tendrá el sufijo “Bundle”.

Ejemplo:

La Figura 11-10 presenta algunos espacios de nombres y nombres de clase Bundle válidos:

Espacio de nombres	Nombre de clase <code>Bundle</code>
<code>Acme\Bundle\BlogBundle</code>	<code>AcmeBlogBundle</code>
<code>Acme\Bundle\Social\BlogBundle</code>	<code>AcmeSocialBlogBundle</code>
<code>Acme\BlogBundle</code>	<code>AcmeBlogBundle</code>

Figura 11-10. Espacio de nombres y nombres de Bundles.

Estándares de Estructura de Archivos Bundle.

La estructura de directorios de un Bundle es simple y flexible. Por defecto, el sistema de Bundles sigue un conjunto de convenciones que ayudan a mantener el código consistente entre todos los Bundles de Symfony2. La Figura 11-11 presenta la estructura básica de un Bundle:

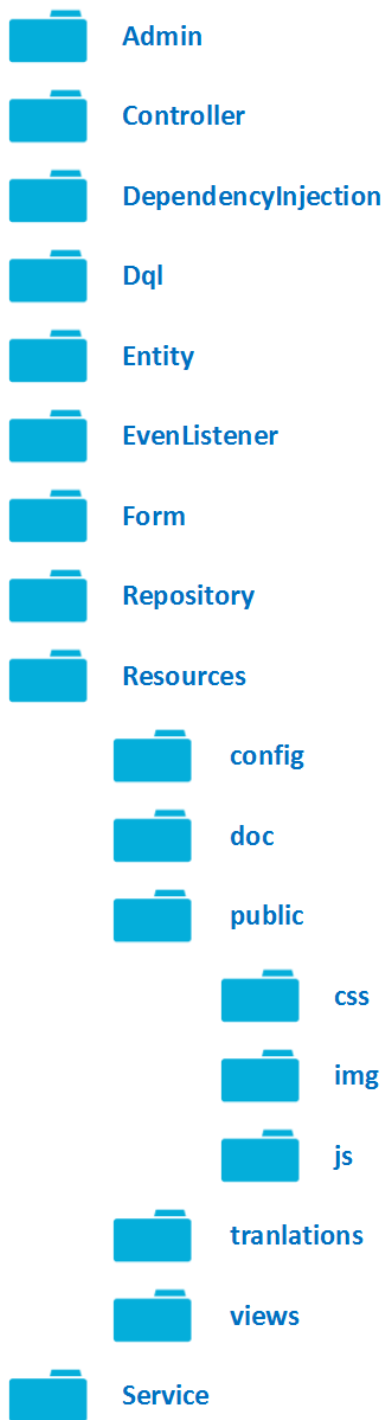


Figura 11-11 Estructura básica de directorios de un Bundle.

- **Admin:** Contiene las clases Admins, perteneciente a los “services⁶³” (objetos creados con un propósito específico) que extienden del Framework SonataAdmin.
- **Controller:** Contiene los controladores del Bundle.
- **DepenencyInjection:** Contiene los archivos que permiten estandarizar y centralizar la construcción de objetos dentro de la aplicación.
- **Dql:** Directorio que contiene los archivos de clases que de las funciones personalizadas que permitirán agregar mayor funcionalidad a Doctrine.
- **Entity:** En este directorio se almacenaran todas aquellas Clases que son el resultado del mapeo objeto-relacional, mediante Doctrine y que ayudan a realizar la persistencia con la base de datos.
- **EventListener:** Contiene los archivos de clases que permiten verificar si un evento ha ocurrido en la base de datos, y realizar determinadas acciones en base a ello.
- **Form:** Contienen las clases que permiten la creación de los formularios en la aplicación.
- **Repository:** Directorio que contiene los archivos de clases que permiten realizar consultas personalizadas a la base de datos.
- **Resource:** Contiene diferentes recursos que brindan funcionalidad a la aplicación.
- **config:** Se encuentran los archivos de configuraciones, incluyendo la configuración de enrutamiento.
- **doc:** Directorio que contiene archivos de ayuda (documentación) del sistema.
- **public:** En este directorio se encuentran elementos web (imágenes, hojas de estilo, etc.), los cuales pueden ser accedidos públicamente.
- **translation:** Contiene los archivos necesarios para la traducción de los mensajes, al idioma Español.
- **views:** Contiene plantillas para las vistas, organizadas por el nombre del controlador.
- **Service:** Contiene las clases tipo “services” personalizadas.

⁶³ SenioLabsNetworks. (5 de Septiembre de 2013). *Service Container*. Obtenido de Symfony: http://symfony.com/doc/current/book/service_container.html.



11.2 Contexto del Sistema

La primera actividad en el proceso de diseño, es comprender las relaciones entre el software que se diseñara y los elementos que interactúan con el mismo. Esto es esencial para definir cómo el sistema proporcionará la funcionalidad que se requiere y cómo será estructurado para que se comunique con su entorno.

Para lograr este objetivo, se hace uso del Modelo de Contexto del Sistema y el Modelo de Interacción del Sistema, que presentan vistas complementarias entre sí, de las relaciones entre un sistema y su entorno.

11.2.1 Modelo de Contexto del Sistema

Un modelo de contexto del sistema, es un modelo estructural, que muestra el sistema a desarrollar, otras entidades en el entorno y las relaciones entre los mismos.

El modelo de contexto de un sistema puede representarse mediante asociaciones de entidades, las cuales muestran simplemente, que existe una relación entre las entidades que intervienen en la asociación. Por lo tanto, es posible documentar el Contexto del Sistema mediante un Diagrama de Bloques que muestre las entidades y sus asociaciones⁶⁴, el cual se presenta a en la Figura 11-12.

El Diagrama de la Figura 11-12, indica que las principales entidades en el entorno del Sistema a desarrollar, y que se relacionan con el mismo son:

- Las Áreas/Departamentos de Mantenimiento.
- Las Dependencias del MINSAL.
- El Sistema Informático de Activo Fijo del MINSAL (SIAF).
- El Sistema Informático Nacional de Abastecimiento del MINSAL (SINAB).

Las líneas que unen estas entidades, indican que existe una relación o alguna forma de interacción entre estas entidades. Y la cardinalidad (número al final de cada línea) indica, por ejemplo, que hay un Sistema de Activo Fijo, pero existen una o muchas Dependencias del MINSAL que se relacionan con él.

⁶⁴ Sommerville, I. (2011). Ingeniería de Software (9a ed.), 7.1.1 Contexto e Interacciones del Sistema. Pearson Education.

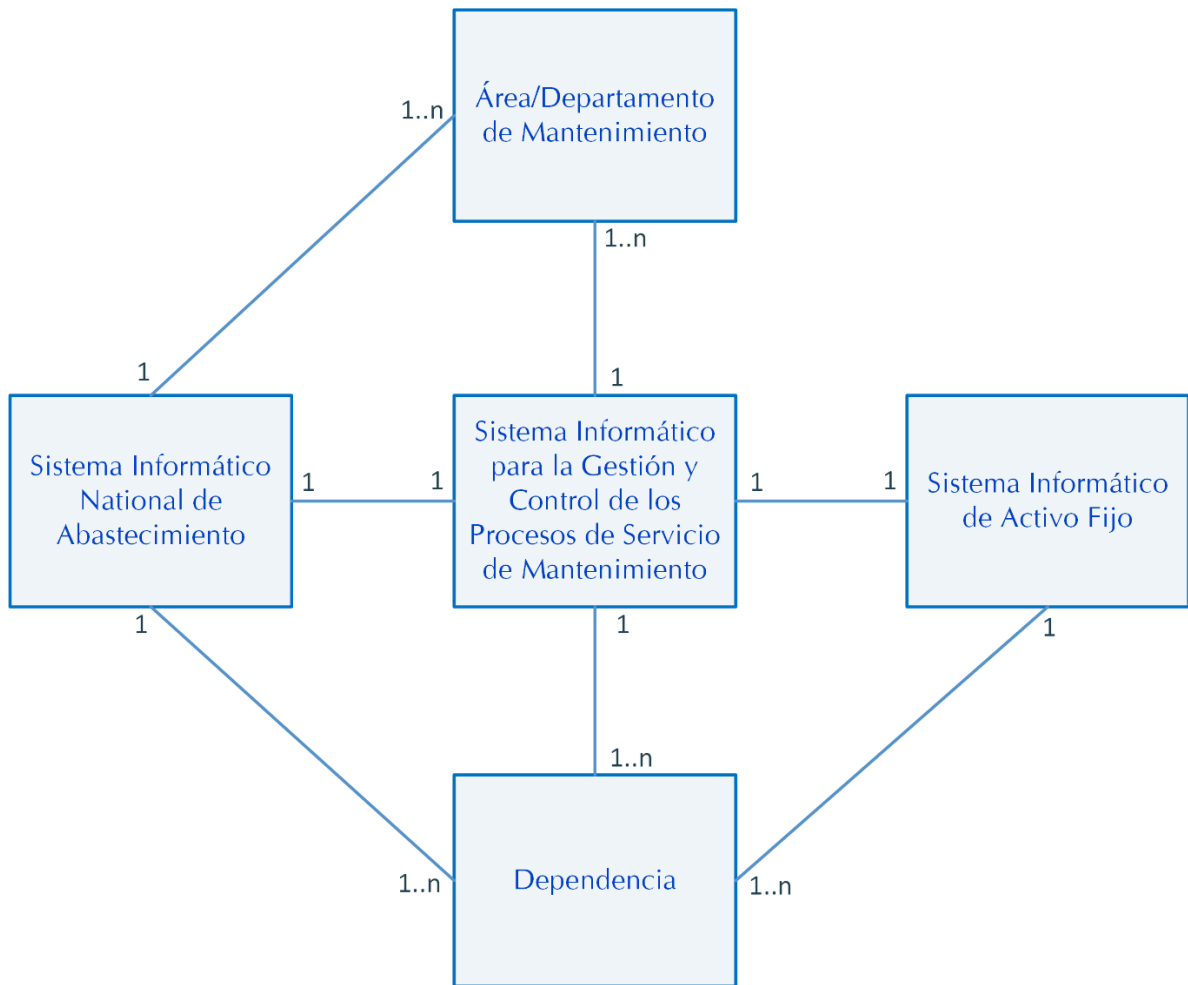


Figura 11-12. Modelo de Contexto del Sistema representado mediante Diagrama de Bloques.

11.2.2 Modelo de Interacción del Sistema

Un modelo de interacción del sistema, es un modelo dinámico, que indica la forma en que el sistema interactúa con su entorno conforme se utiliza⁶⁵. Al modelar las interacciones de un sistema con su entorno, se debe utilizar un enfoque abstracto, es decir, sin entrar en demasiados detalles. Por tanto, las interacciones del Sistema Informático de Mantenimiento se presentan mediante un Diagrama de Casos de Uso de nivel contextual (Para conocer la nomenclatura del Diagrama de Casos de Uso, ver sección 8.6.2 del presente documento), ya que cada caso de uso representa una interacción con el Sistema.

El Modelo de Interacción del Sistema se presenta en el Diagrama de la Figura 11-13, en el cual se muestran las interacciones principales existentes:

⁶⁵ Sommerville, I. (2011). Ingeniería de Software (9a ed.), 7.1.1 Contexto e Interacciones del Sistema. Pearson Education.

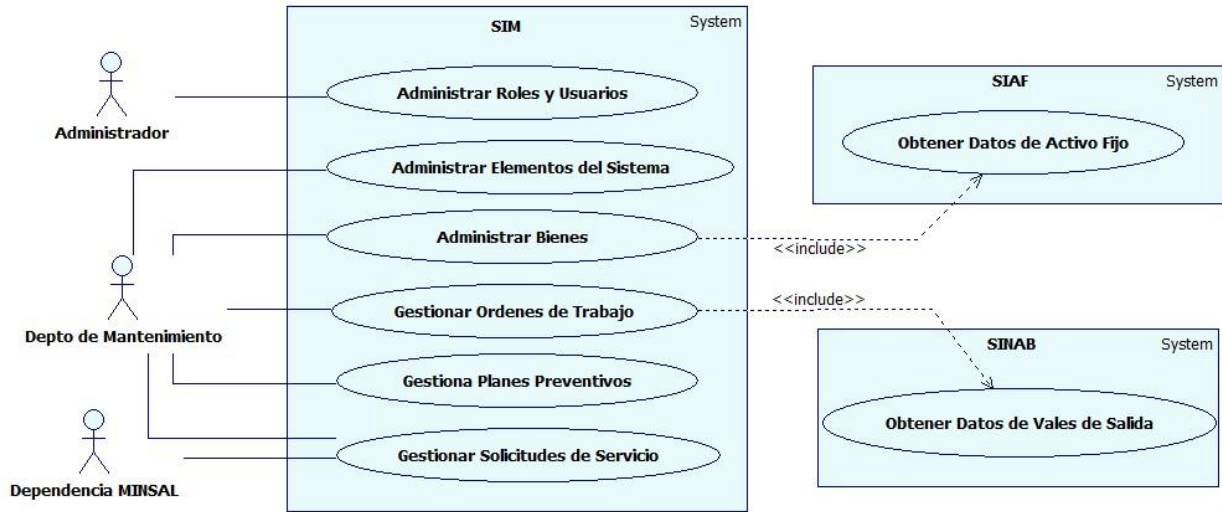


Figura 11-13. Modelo de Interacción del Sistema representado mediante Diagrama de Casos de Uso.

En la figura anterior (Figura 11-13), se muestra como los Administradores, Departamentos de Mantenimiento y Dependencias del Ministerio de Salud interactúan con el Sistema Informático a desarrollar. De forma específica, quienes interactuaran con el sistema son los usuarios que pertenecen a cada una de estas entidades (por ejemplo, los Jefes de los Departamentos de Mantenimiento), sin embargo, se ha presentado de esa forma ya que, como se mencionó anteriormente, se debe usar un enfoque abstracto, que es fundamental para una mejor comprensión del mismo.

Obsérvese como el Sistema Informático de Mantenimiento interactúa con los Administradores, Departamentos de Mantenimiento y Dependencias del MINSAL, proporcionando servicios de Administración de Roles y Usuarios, Administración de Bienes, Gestión de Solicitudes, entre otros.

De igual forma el SIAF y SINAB, interactúan con el Sistema Informático de Mantenimiento proporcionando acceso y transferencia de datos (Datos de Activo Fijo y Datos de Vales de Salida) que son necesarios para un correcto funcionamiento del SIM. Por esta razón, el Sistema Informático de Mantenimiento se puede observar como otro actor en relación con el SIAF y SINAB, tal como se muestra en la Figura 11-14:

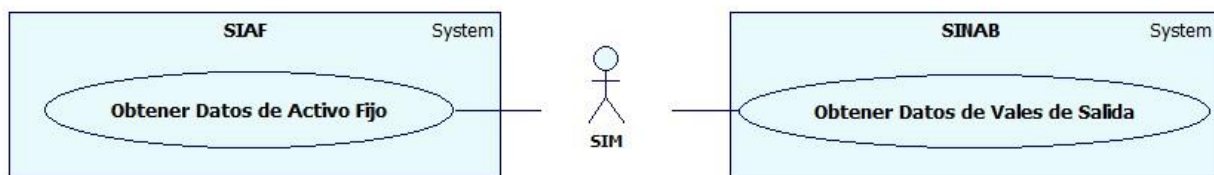


Figura 11-14. SIM y su Interacción con Sistemas Externos.



11.2.3 Diagrama de Contexto Arquitectónico

A nivel de diseño arquitectónico, la manera de representar que el software interactúa con otras entidades ubicadas más allá de sus límites, es mediante un Diagrama de Contexto Arquitectónico⁶⁶. Por este motivo, se presenta el siguiente Diagrama de Contexto Arquitectónico del Sistema Informático para Gestión y Control de los Procesos de Mantenimiento (Figura 11-15):

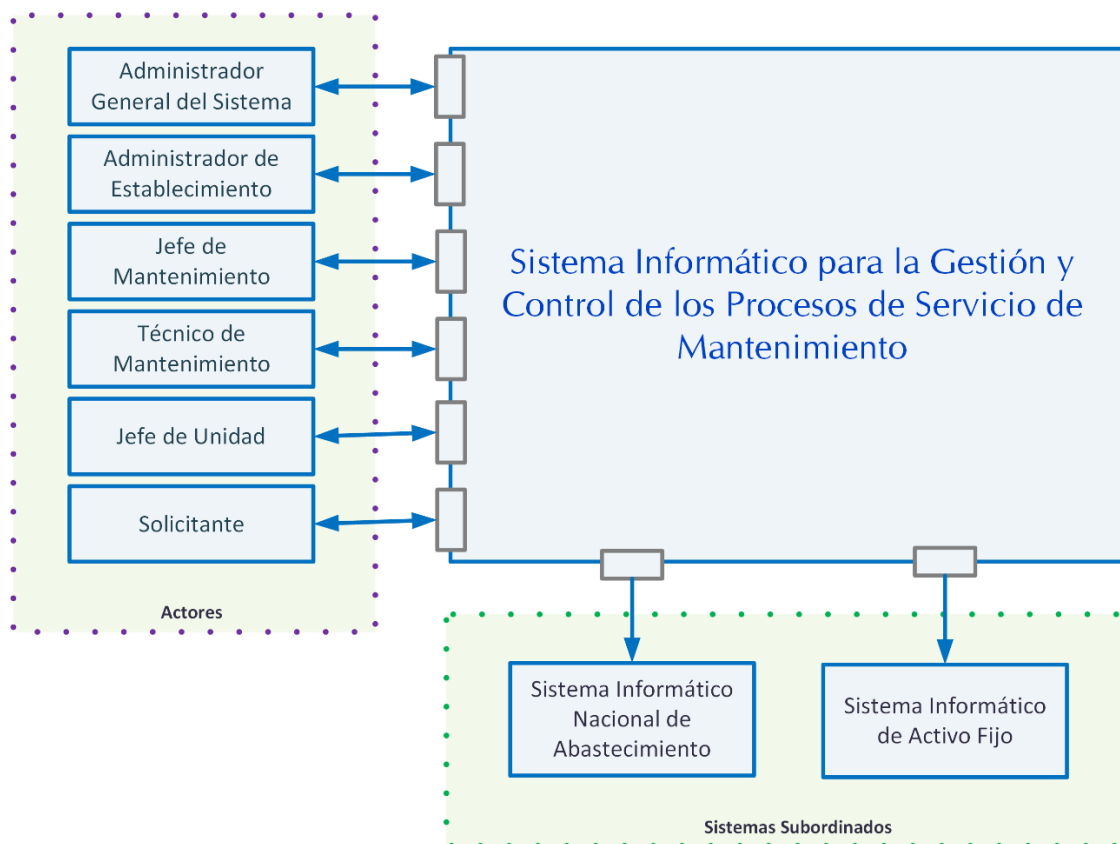


Figura 11-15. Diagrama de Contexto Arquitectónico.

En el Diagrama de Contexto anterior se puede observar los siguientes tipos de elementos:

- **Sistemas Subordinados:** Son sistemas utilizados por el Sistema a desarrollar (Sistema Informático de Mantenimiento) y que proporcionan los datos necesarios para completar la funcionalidad del Sistema de Mantenimiento.
- **Actores:** Son entidades, ya sea personas, dispositivos u otros similares, que interactúan con el Sistema Informático de Mantenimiento, produciendo o solicitando información necesaria, es por ello que la relación entre estos y el Sistema Informático de Mantenimiento se representa con una flecha en doble sentido.

⁶⁶ Pressman, R. (2005). Ingeniería del Software, Un Enfoque Practico (6a Ed.), 10.4.1. Representación del Sistema en el Contexto. McGraw-Hill.



- Interfaces: Las interfaces están representadas por los pequeños recuadros de color gris, y es por medio de estas interfaces que existe la comunicación entre el Sistema y las entidades externas.

Una vez que se ha modelado el contexto del Sistema a desarrollar, se debe definir y refinar los componentes que constituyen la arquitectura del mismo. Este proceso, se realiza de manera iterativa hasta que se obtiene una estructura arquitectónica completa. El siguiente paso es entonces el Diseño Arquitectónico.



11.3 Diseño Arquitectónico.

El diseño Arquitectónico es un proceso creativo, en el que se diseña cómo se organiza el sistema, de forma que este satisfaga los requisitos funcionales y no funcionales. El Diseño Arquitectónico da un punto de vista del sistema que permite verlo como un todo, y a la vez constituye el punto de partida de los subsiguientes modelos de diseño (como el modelo de clases o el modelo de datos), siendo esta una de las herramientas de comunicación entre los interesados en el desarrollo del sistema⁶⁷. Por tanto, el diseño de la arquitectura del sistema, tendrá un efecto profundo en el desarrollo de todo el proyecto, porque los resultados de esta, guiarán todos los demás aspectos del diseño.

11.3.1 Diseño Arquitectónico de Alto Nivel

En un diseño de alto nivel, se describen los componentes principales del Sistema y el modo en que interactúan entre sí, sin entrar en demasiados detalles. Para ello se utiliza el siguiente Diagrama de Paquetes (Figura 11-16), en el cual se identifican los elementos principales que constituyen el Sistema Informático:

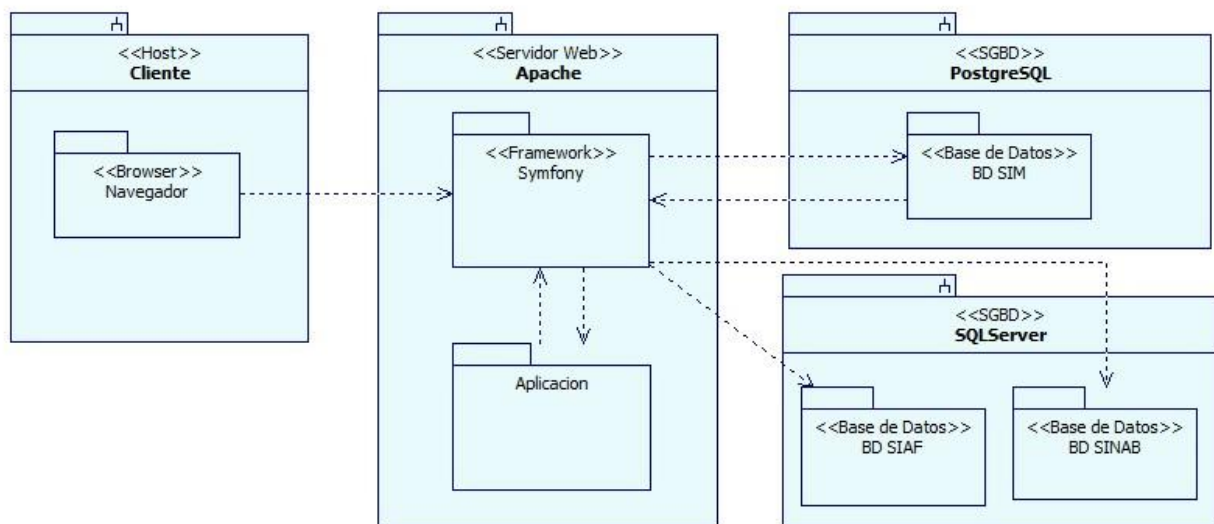


Figura 11-16. Diseño Arquitectónico a Nivel de Paquetes.

En el Diagrama de la Figura 11-16, solamente se identifican de forma general los principales elementos del Sistema a desarrollar. Hay cuatro Subsistemas claramente definidos y dentro de cada una de ellos una serie de Paquetes, los cuales se describen a continuación:

- **Cliente:** Representa la máquina cliente o host desde donde el usuario interactúa con el software por medio de un navegador web. El término host es usado en informática para referirse a las computadoras conectadas a una red.

⁶⁷ Pressman, S. R. (2010). Ingeniería del Software (7a ed.), 9.1.2 ¿Por qué es Importante la arquitectura? McGraw-Hill.



- **Navegador Web:** Representa la herramienta que permite la visualización o acceso a recursos/servicios (páginas web) de otros equipos (como servidores web o servidores de base de datos). Desde aquí el navegador, el usuario podrá acceder a la aplicación, realizar consultas de datos, realizar registros etc. Los Navegadores Web que pueden ser utilizados como clientes, y que a la vez han sido probados por el equipo de desarrollo son:
 - ✓ Iceweasel (*)
 - ✓ Mozilla Firefox
 - ✓ Google Chrome.
- (*) De los navegadores mencionados, el equipo de desarrollo recomienda el uso de Iceweasel, ya que es el navegador oficial de Debian 7.0 (Sistema Operativo sobre el cual funcionará la solución), que además reúne todas las características de su equivalente Mozilla Firefox. Además, es en este navegador en donde las Pruebas de Software han sido ejecutadas.
- **Apache:** Es el servidor web que alojará los recursos que el usuario solicita y que son vitales para el funcionamiento del sistema. El servidor web es el encargado de recibir las peticiones que el usuario realiza por medio del navegador, procesarlas y devolver una respuesta. Por ejemplo, si el usuario solicita una página web, el servidor se encarga de enviarla para que puede visualizarla. El servidor Apache contendrá, el framework de desarrollo y la aplicación a desarrollar.
 - **Symfony:** Es un completo framework diseñado para facilitar el desarrollo de las aplicaciones web basado en el patrón Modelo Vista Controlador y el cual utiliza el lenguaje de programación php.
 - **Aplicación:** Representa el conjunto de archivos de código fuente que automatizarán los procesos del negocio, permitirán la intercomunicación con otros sistemas, las configuraciones para acceder a la base de datos, interfaz de usuario, entre otros.
- **PostgreSQL:** Es un Sistema Gestor de Bases de Datos Open Source. Permite el almacenamiento, modificación, eliminación y extracción de datos en una base de datos.
 - **BD SIM:** Dentro de PostgreSQL, se tendrá la Base de Datos propia para el SIM.
- **SQLServer:** Al igual que PostgreSQL, es un SGBD, con la diferencia que ha sido desarrollado por Microsoft. Dentro de él, se encuentran desarrolladas dos fuentes de datos que alimentarán el Sistema a desarrollar:
 - **BD SIAF:** Es la Base de Datos del SIAF, contiene información relacionada con el inventario de equipos que posee cada una de las dependencias del MINSAL.
 - **BD SINAB:** Es la Base de Datos del SINAB, contiene información relacionada a los costos de materiales y repuestos.

Luego de tener un diseño de muy alto nivel, como en el diagrama descrito anteriormente (Figura 11-16), este debe irse refinando. Por lo cual, luego de identificar de manera superficial los principales componentes que constituyen el Sistema y sus interacciones, se deben organizar utilizando algún Modelo o Patrón arquitectónico.



Generalmente, no es necesario crear una nueva arquitectura de software para cada Sistema Informático que se desarrolla. La arquitectura de un sistema de software puede estar basado en un patrón de arquitectura ya definido o estilo particular.

Un patrón arquitectónico es un estilo o descripción abstracta de buenas prácticas, que ha sido probado en diferentes sistemas y entornos. Por lo tanto, un patrón arquitectónico describe la organización de una arquitectura que ha tenido éxito en sistemas anteriores.

Para el desarrollo del SIM se utiliza una combinación de dos patrones arquitectónicos: Modelo de Tres Capas y el Patrón MVC.

11.3.2 Modelo de Tres Capas.

Es una arquitectura cliente-servidor en el que el objetivo primordial es la separación de la lógica del negocio, los datos y la presentación al usuario (Interfaz de Usuario).

- Capa de presentación: Es la capa con la cual el usuario interactúa, por lo cual también se le denomina Capa de Usuario, ya que presenta al usuario la información solicitada y captura los datos ingresados por él. Esta capa se comunica únicamente con la capa de negocio.
- Capa de negocio: Se denomina también Capa de Lógica del Negocio porque es aquí donde se establecen todas las reglas que deben cumplirse. Esta Capa se comunica con la Capa de Presentación, para recibir las solicitudes o peticiones de usuario y presentar los resultados o respuestas a dichas solicitudes, y con la Capa de Datos, para solicitar al gestor de base de datos almacenar o recuperar datos de él.
- Capa de datos: Es donde residen los datos, recibe solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

En la Figura 11-17 se representa gráficamente el Modelo de Tres Capas y la relación entre estas. El SIM funcionara en un ambiente web, por lo cual los usuarios podrán acceder al sistema mediante un navegador web desde un equipo Cliente localizado en cualquier establecimiento del MINSAL el cual hará el papel de Capa de Presentación, desde aquí se realizarán solicitudes o peticiones a un Servidor Web el cual hace el papel de Capa de Negocio, este a su vez se comunica con el Sistema Gestor de Base de Datos para poder satisfacer las peticiones solicitadas desde la Capa de Presentación.

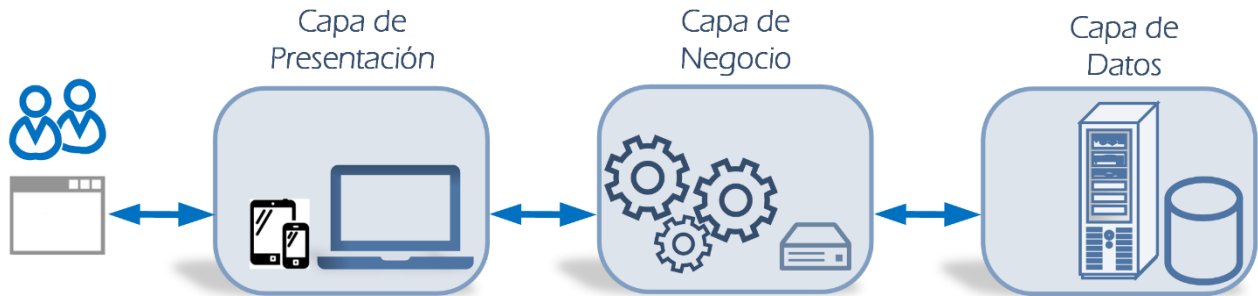


Figura 11-17. Modelo de Tres Capas

11.3.3 Patrón MVC (Modelo-Vista-Controlador).

El Patrón o Arquitectura MVC (Modelo-Vista-Controlador), separa la presentación y la interacción de los datos del sistema. El sistema se estructura en tres componentes lógicos que interactúan entre sí.

El componente Modelo gestiona los datos del sistema y las operaciones asociadas a esos datos. El componente Vista define y administra la forma en que se presentan los datos para el usuario. El componente Controlador gestiona la interacción del usuario, es decir, sus peticiones o solicitudes (por medio de selección de opciones de menú, clics sobre botones, pulsación de teclas, etc.) y pasa estas interacciones al Modelo en caso de ser necesario, y finalmente retornando una respuesta a la Vista, además de realizar el procesamiento y otros cálculos.

A simple vista, el Patrón MVC podría confundirse con el Modelo de Tres Capas, sin embargo el patrón MVC, sostiene separar los componentes de un sistema (separar los elementos visuales/gráficos de su correspondiente programación y acceso a datos), mientras que el Modelo de Tres Capas organiza los componentes de un sistema en relación a su implementación física.

En la Figura 11-18 se representa gráficamente el Patrón MVC, y una descripción del comportamiento de este Patrón:

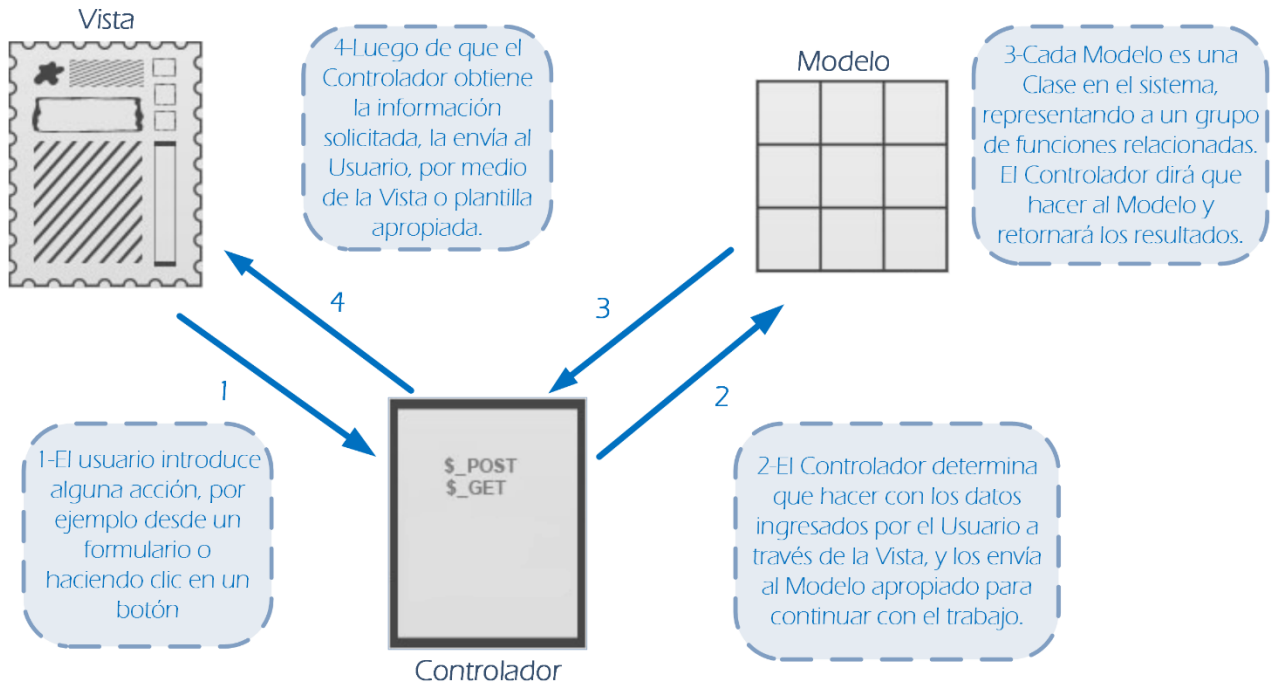


Figura 11-18. Patrón MVC.

Para observar de una manera más clara la secuencia del funcionamiento de este Patrón, se presenta el Diagrama de Secuencia del Patrón MVC en la Figura 11-19:

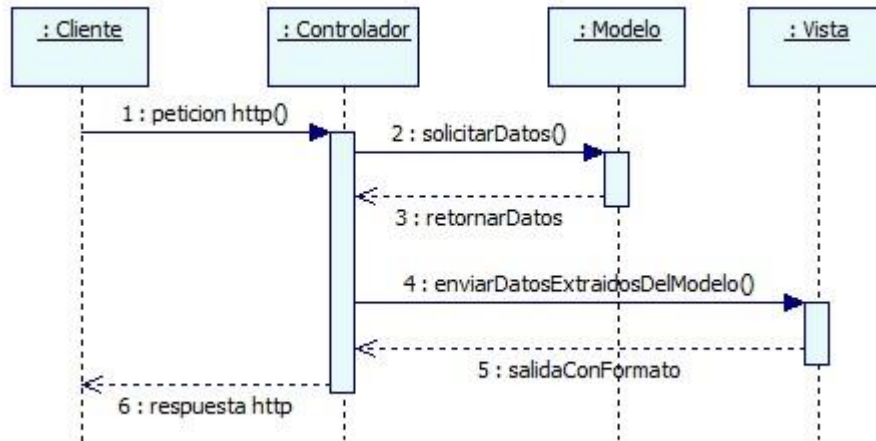


Figura 11-19. Diagrama de Secuencia del Patrón MVC.

El Diagrama de Secuencia (Para conocer la nomenclatura de los Diagramas de Secuencia, ver sección 8.6.2 del presente documento) de la Figura 11-19 indica que:



1. El Cliente envían una petición http (Flecha 1 de la Figura 11-19), por ejemplo: inicia sesión, solicita una página web, envía un formulario, etc.
2. El Controlador, decide cómo manejar las peticiones, en este caso, solicita datos al Modelo para realizar las acciones requeridas (Flecha 2 de la Figura 11-19), por ejemplo: solicita los datos de un empleado, como el nombre, apellidos, cargo, etc.
3. El Modelo envía los datos solicitados al Controlador (Flecha 3 de la Figura 11-19).
4. El Controlador, transfiere los datos recibidos del Modelo a la Vista para ser formateados (Flecha 4 de la Figura 11-19), es decir, aplicar algún estilo, codificación, incrustarlos en alguna plantilla, etc.
5. Finalmente, cuando la vista realiza su trabajo (Flecha 5 de la Figura 11-19), el Controlador retorna la respuesta al cliente, por ejemplo: la página web solicitada o la información de algún empleado.

Este Patrón Arquitectónico es útil en el desarrollo del SIM, debido a que una aplicación web o un módulo de software que sigue la estructura MVC separa los tres tipos principales de funcionalidad en tres tipos de archivos: modelos, vistas y controladores. Esto permite que cada parte sea diseñada, implementada y probada de forma independiente de cualquier otra, manteniendo el código organizado.

Debido a la separación y organización del código la detección y solución de errores es mucho más fácil, así como la modificación del código para agregar nuevas funcionalidades. Al mismo tiempo favorece la reutilización de componentes. Además, Symfony es un Framework que trabaja con el Patrón MVC. Por lo cual, el uso de este Patrón es indispensable para el desarrollo del SIM.

11.3.4 Funcionamiento Interno del Framework Symfony

En el Diseño Arquitectónico, se debe elegir los principales elementos en los que se basará la Arquitectura del Sistema, esto incluye también el uso de Frameworks. Los Frameworks, afectan en gran medida la Arquitectura del Sistema, por tanto su especificación en dicho Diseño es sumamente importante. La utilización de Symfony es un requerimiento, es decir, que su especificación en el Diseño Arquitectónico no es una opción, sino una necesidad inminente.

Es importante entonces conocer la Arquitectura del Framework Symfony, para saber cómo se acopla a la Arquitectura de Sistema a desarrollar. El Diagrama de la Figura 11-20, presenta los principales componentes de Symfony y la forma en que estos se comunican o interactúan para realizar su trabajo, mostrando a la vez, el proceso completo desde que el usuario realiza una petición mediante una URL, hasta que el servidor le entrega como respuesta una página web:

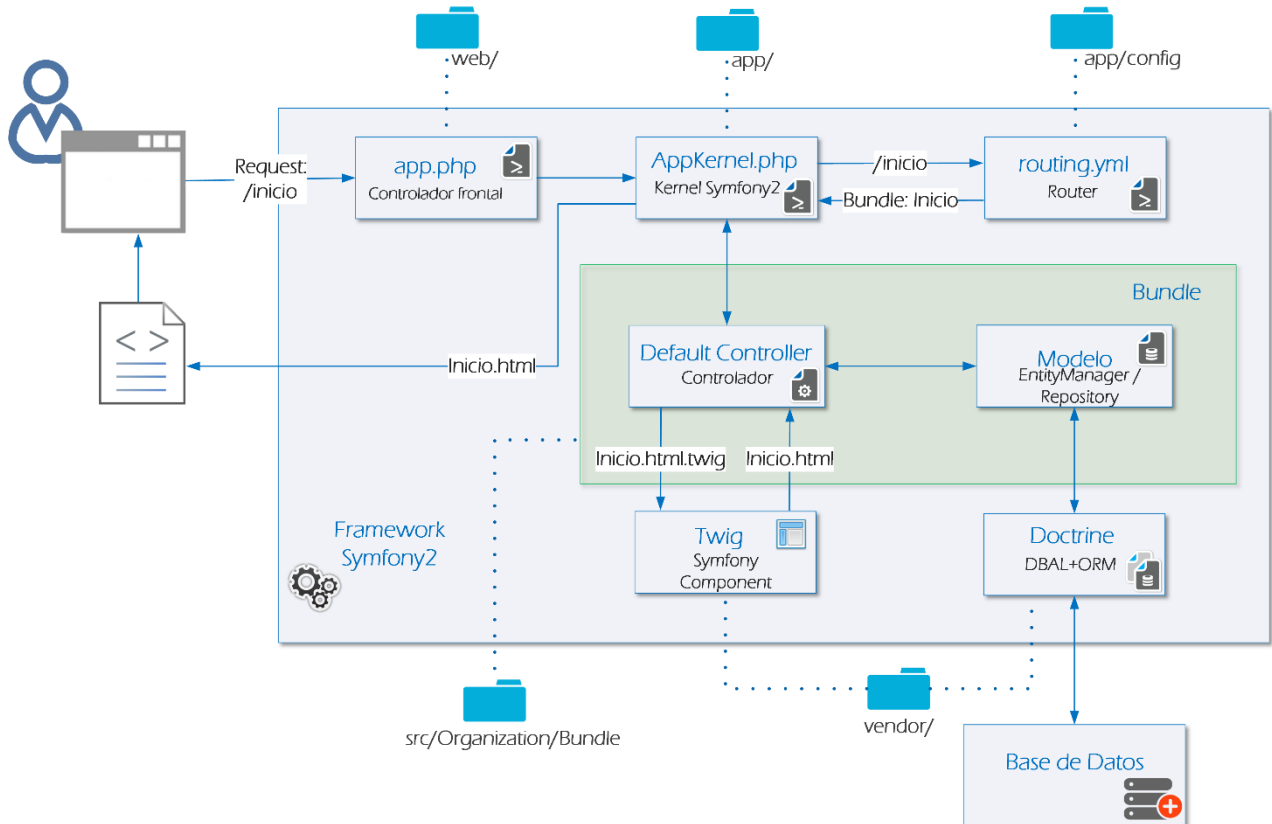


Figura 11-20. Funcionamiento Interno de Symfony.

Descripción del Diagrama (Figura 11-20).

El usuario desde su navegador, realiza una **Petición** (request), es decir, solicita la página de inicio del sitio web a través de una URL. Por ejemplo: "http://tbs115.localhost/inicio".

Dentro del directorio **web/** de la aplicación, se encuentra el script **app.php** al cual le son redirigidas todas la peticiones. Este archivo se denomina **controlador frontal** y es el único punto de entrada a la aplicación. El **controlador frontal**, también es el encargado de arrancar la aplicación, por esta razón, crea una instancia del **kernel** de la aplicación mediante una instancia de la clase **AppKernel.php**, que se encuentra dentro de la carpeta **app/**.

El kernel registra todos los bundles activos de la aplicación y carga el archivo de configuración **config_prod.yml** (archivo de configuración para el entorno de producción), que se encuentra dentro de la carpeta **app/config/**. A su vez, este archivo importa el archivo **config.yml** (archivo de configuración general de Symfony), ubicado dentro de la misma carpeta, el cual carga las rutas del archivo **routing.yml**.

El archivo **routing.yml**, determina cuál es el controlador que se debe ejecutar para cada URL solicitada por los usuarios. Esta transformación se realiza mediante la información de las rutas configurada en este archivo.

A continuación, el **controlador frontal** crea un objeto de tipo Request rellenándolo con la información de la petición del usuario y se lo pasa al **kernel**. El procesamiento de la petición comienza



notificando algunos eventos y continúa preguntando al componente de enrutamiento qué controlador se debe ejecutar para atender la petición del usuario.

Una vez localizado el controlador y la acción que corresponden a la URL solicitada, en este caso representado por el controlador **Default Controller**, se ejecuta su código. Si es necesario, el controlador hace uso del **Modelo** para buscar información en la base de datos. Normalmente esta búsqueda se realiza mediante un **Repositorio** propio creado para las entidades que **Doctrine2** genera a partir del mapeo objeto-relacional, y en el caso de registrar información nueva se utiliza un **EntityManager**.

Si el **Controlador** devuelve un objeto de tipo **Response**, el kernel lo transforma en la página que se devuelve al usuario. Si el **Controlador**, especifica también el uso una plantilla, el kernel hace uso de la **Vista** para renderizar la plantilla según el motor de plantillas configurado en la aplicación que en este caso, es Twig. La ejecución finaliza devolviendo al usuario la página creada a partir de la plantilla.

Nota:

- Para una definición más detallada de los elementos anteriormente descritos (resaltados en negrita) y que están presentes en el Diagrama de la Figura 11-20, ver Glosario Técnico en la sección 16 del presente documento.
- Para conocer más detalladamente la Estructura de Archivos de Symfony vea la sección 11.1.10 del presente documento.



11.3.5 Diseño a Nivel de Componentes

Ahora que se han definido los Patrones Arquitectónicos a utilizar para el desarrollo de Sistema Informático de Mantenimiento (Modelo de Tres Capas y Patrón MVC) y la utilización del Framework Symfony, se tiene la base para organizar los Componentes que conformaran el Sistema y determinar la relación entre estos.

Un Componente, es una unidad de software independiente y funcional, que puede organizarse con otros componentes para crear un sistema de software⁶⁸.

Para representar cómo un sistema de software es dividido en componentes y mostrar las dependencias entre estos, se utiliza el Diagrama de Componentes, el cual provee una vista arquitectónica de alto nivel. Así mismo, permite visualizar con más facilidad la estructura general del sistema y el comportamiento del servicio que estos componentes proporcionan y utilizan a través de las interfaces.

La Figura 11-21 presenta el Diagrama de Componentes del Sistema, tomando en cuenta el Modelo de Tres Capas y el patrón MVC según las especificaciones del Framework Symfony.

Para conocer la nomenclatura de los Diagramas de Componentes, ver sección 8.6.2 del presente documento.

⁶⁸ Sommerville, I. (2011). Ingeniería de Software (9a ed.), 17.1 Componentes y Modelos de Componentes. Pearson Education.

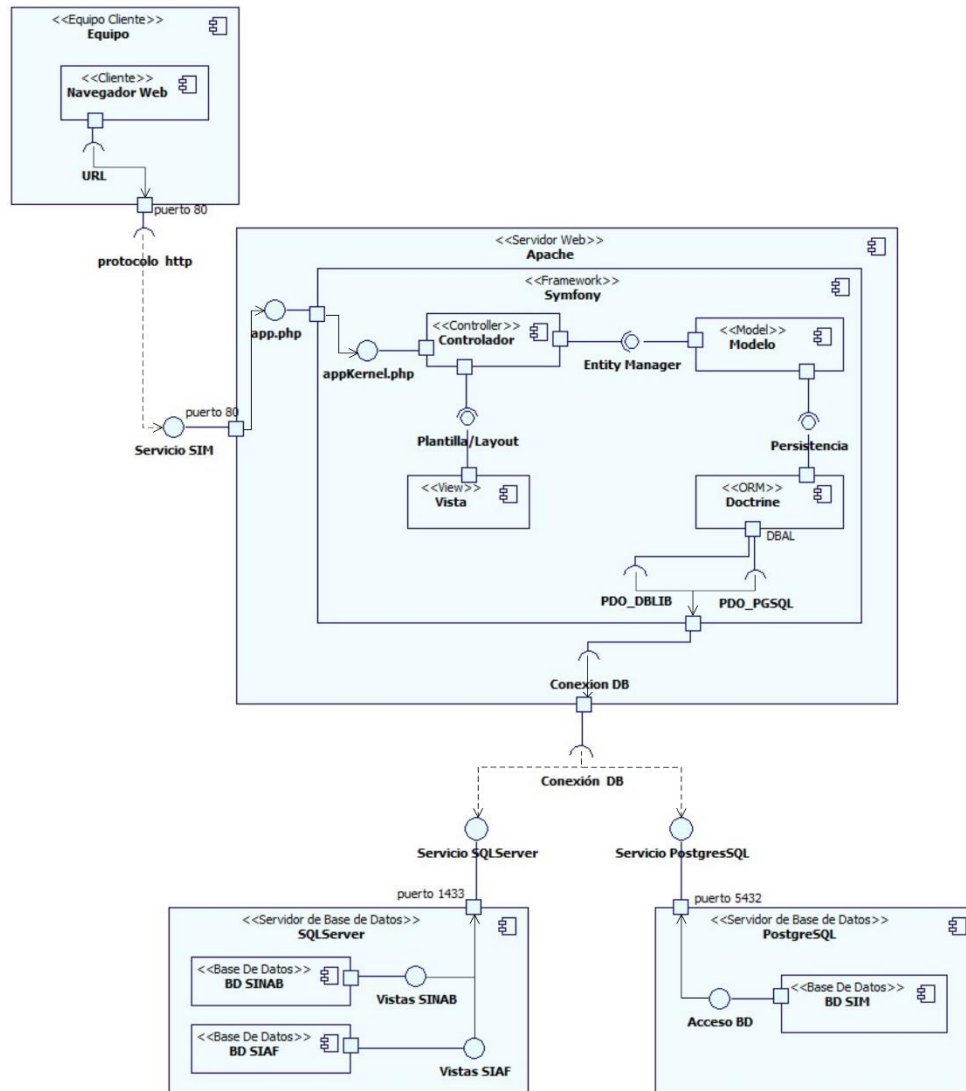


Figura 11-21. Diagrama de Componentes SIM.

11.3.5.1 Descripción del Diagrama de Componentes del Sistema.

En la Figura 11-21, El **Navegador Web** (ya definido en la sección 11.3.1 del presente documento) es la herramienta que el usuario utiliza en su **Equipo** para acceder a algún recurso de la aplicación en el **Servidor Web**, esto lo logra mediante una Interfaz Requerida **URL**. A través de una **URL**, el usuario envía una petición al **Servidor Web**, utilizando el protocolo **HTTP**, que utiliza una conexión hacia el **Servidor Web** abierta en el **Puerto 80**.

- **URL:** Localizador Uniforme de Recursos, se usa para nombrar recursos en Internet para su localización o identificación, como por ejemplo documentos textuales, imágenes, vídeos, consultas a bases de datos, etc. Ejemplo de URL: www.salud.gob.sv



- **HTTP:** Es un protocolo de comunicación orientado a transacciones y sigue el esquema petición-respuesta entre un cliente y un servidor, que por defecto utiliza el Puerto 80. El cliente que efectúa la petición y el servidor responde con el recurso solicitado.

El **Servidor Web**, se encuentra ejecutando el Servicio **Apache**, el cual escucha las peticiones entrantes de los **Cientes** en el **Puerto 80**, poniendo a disposición de los clientes el Servicio cuando este sea requerido (por lo cual es representado como una Interfaz Proporcionada, ya que pone a disposición el servicio web).

Según la configuración del Servidor Web Apache, las peticiones son redirigidas al archivo **app.php**, el cual se explicó en la sección anterior (11.3.4). Así mismo, el archivo AppKernel.php se encarga de arrancar la aplicación y mediante el uso de la configuración de rutas (routing.yml) determina que controlador ejecutar y sus métodos.

Los Controladores, disponen de **EntityManager** para hacer uso de los **Modelos** generados a partir del Mapeo Objeto-Relacional realizado por **Doctrine**, teniendo acceso a la **Persistencia**, y por tanto poder interactuar con la Base de Datos.

Doctrine utiliza un **DBAL**. Un DBAL es la Capa de Abstracción y Acceso de Datos de Doctrine, que se encuentra justo arriba de la Capa de Drivers **PDO**, y ofrece una serie de funciones flexibles para comunicarse con la Base de Datos, es decir, facilita la realización de consultas y otras operaciones de Base de Datos. La esencia del **DBAL** es hacer uso de diferentes Drivers de Base de Datos. Por ejemplo, DBAL contiene drivers **PDO** para la conexión con PostgreSQL (**PDO_PGSL**) y con SQLServer (**PDO_DBLIB**).

PDO_PGSQL: es un controlador que implementa la interfaz PHP Data Objects (PDO) para permitir el acceso desde PHP a bases de datos PostgreSQL.

PDO_DBLIB: Es un controlador que implementa la interfaz PHP Data Objects (PDO) para habilitar el acceso desde PHP a bases de datos de Microsoft SQL Server y Sybase a través de la biblioteca FreeTDS.

Al realizar la conexión con los Gestores de Base de Datos, a través de los servicios de **SQLServer** (mediante el **Puerto 1433**) y **PostgreSQL** (mediante el **Puerto 5432**), se tendrá el acceso a las **Vistas** proporcionadas en las Base de Datos del SIAF y SINAB y a la Base de Datos completa del Sistema a desarrollar (SIM) respectivamente.

Nota: Para una definición más detallada de los elementos anteriormente descritos (resaltados en negrita) y que están presentes en el del Diagrama de la Figura 11-21, vea las definiciones en el Glosario Técnico (sección 16 del presente documento).

Otra forma de representar los Componentes que Constituyen el Sistema Informático de Mantenimiento y sus interacciones, y para tener una idea más clara de su arquitectura, en la Figura 11-22 se plasma la Representación del Modelo de Tres Capas y el Patrón MVC para el Sistema Informático de Mantenimiento:

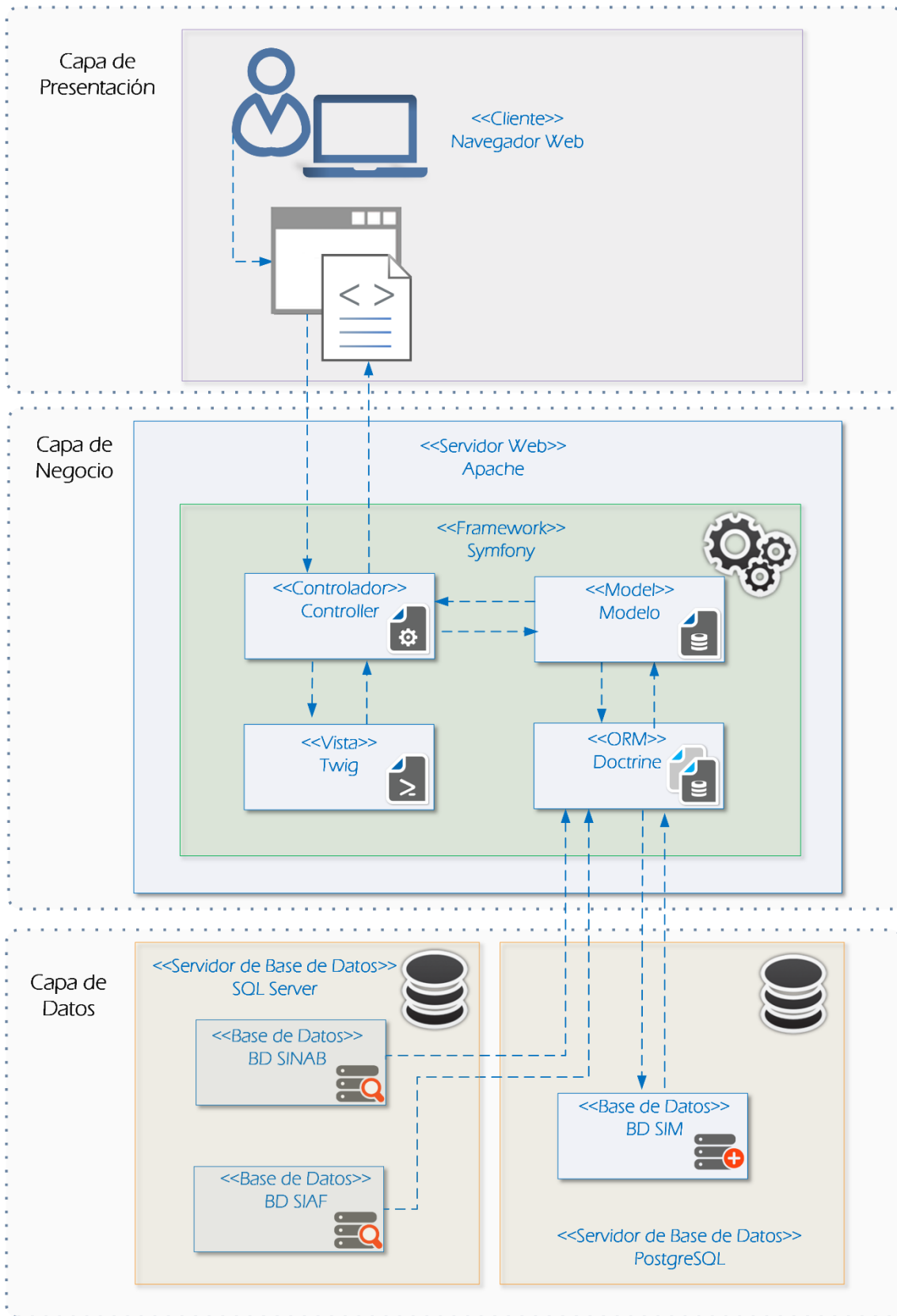


Figura 11-22. Representación del Modelo de Tres Capas y el Patrón MVC del SIM



11.3.6 Diseño de la Arquitectura Física (Despliegue)

Se ha presentado hasta este punto, una serie de diagramas que muestra la arquitectura del Sistema a desarrollar, sin embargo, estos diagramas cubren el punto de vista lógico. Es importante también diseñar la forma en que los componentes que constituyen el Sistema, estarán físicamente distribuidos. Esto significa, que se debe comprender como los diferentes componentes del Sistema, se van a ejecutar físicamente y la manera en que cada uno de ellos se comunicara con otros.

Para lograr este objetivo, se utiliza el Diagrama de Despliegue. Los Diagramas de Despliegue modelan precisamente, la Arquitectura Física del Sistema, ya que, muestran la relación entre los componentes de software y hardware del Sistema y su distribución física para su ejecución o procesamiento. Es importante mencionar que se considera, la relación de los componentes de software y hardware del Sistema y su distribución, pero no su localización en el mundo real.

El Diagrama de Despliegue del Sistema se muestra en la Figura 11-23. En el Diagrama, se observan diferentes elementos que representan el despliegue físico del software y el hardware. Los recuadros de colores que se observan, son conocidos como Nodos, mientras que los recuadros en color blanco son conocidos como Artefactos.

Un Nodo, representa un recurso computacional o de procesamiento de un sistema⁶⁹, es decir, un dispositivo de hardware o algún entorno de ejecución de software, por ejemplo, computadoras personales, sensores, impresoras, servidores o virtualizaciones. Los Nodos pueden interconectarse mediante Asociaciones, que muestran como los Nodos se comunican entre sí, formando la estructura de red.

Un Artefacto, representa una entidad que se implementa físicamente⁷⁰, es decir, un elemento concreto que es el resultado de un proceso de desarrollo, por ejemplo, archivos ejecutables, librerías, componentes de software, documentos o bases de datos. Los Artefactos son desplegados en Nodos, indicando que el Nodo brinda soporte a estos artefactos para que puedan procesarse o ejecutarse.

⁶⁹ ¹² IBM. (2013, 08). Deployment diagrams. Extraído de: Help - Designing a software application using models.:<http://publib.boulder.ibm.com/infocenter/rsdvhhelp/v6r0m1/index.jsp?topic=%2Fcom.ibm.xtools.modeler.doc%2Ftopics%2Fcddepd.html>.

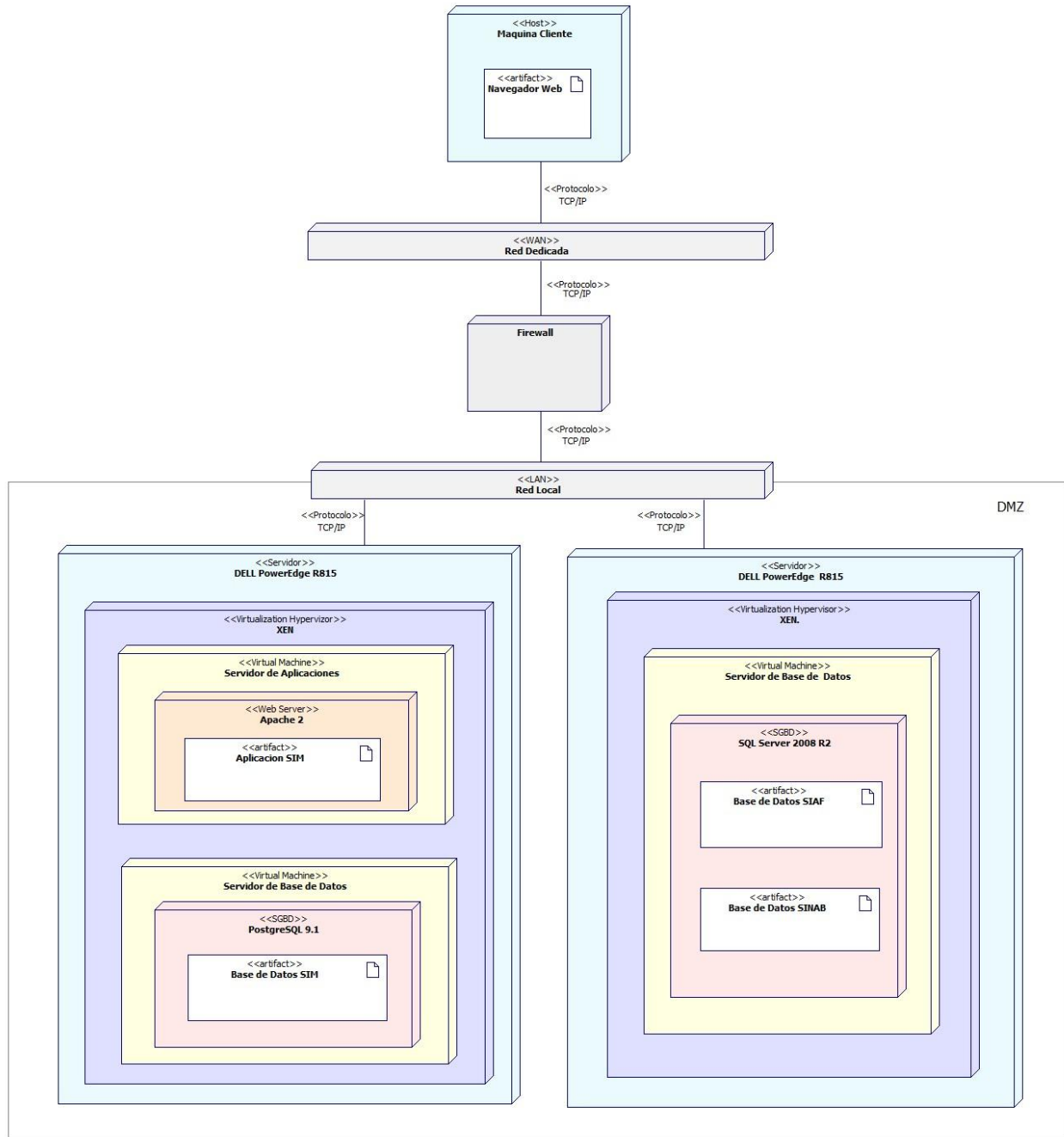


Figura 11-23. Diagrama de Despliegue del Sistema.



En el Diagrama anterior (Figura 11-23), se observan los siguientes Nodos y Artefactos:

Nodos:

Dispositivos Físicos Computacionales:

- **Maquina Cliente:** Representa el equipo (computadora) donde el usuario se encontrará trabajando, ya ingresando dantos, realizando consultas, entre otras acciones. También se le conoce como “Host”, ya que es un dispositivo conectado a una red. Este equipo debe cumplir con los requerimientos mínimos para un óptimo funcionamiento (Para ver la descripción y requerimientos mínimos de la maquina cliente ver sección 10.4.3.1, del presente documento).
- **DELL PowerEdge R815:** Representa los equipos “Servidores” en donde se encuentra el Software y las Bases de Datos respectivamente. Estos proveen los servicios a los otros nodos cliente para ser utilizados. El nombre de estos nodos, se debe a su marca y modelo, y en ellos funcionará la solución (Para ver las especificaciones técnicas del Servidor PowerEdge R815, ver sección 10.4.3.1 del presente documento).
Los servidores se encuentran, lógicamente dentro un DMZ o Zona Desmilitarizada, por sus siglas en inglés (ver definición en Glosario Técnico, sección 16 del presente documento).

Dispositivos Físicos de Red:

- **Red Dedicada:** Representa la red de área amplia que interconecta diferentes establecimientos y dependencias del MINSAL. Se le denomina a la vez “WAN” (red de área amplia), porque es una red de computadoras que abarca varias ubicaciones físicas considerablemente distantes.
- **Red Local:** Representa la red local del MINSAL (entiéndase MINSAL, en este contexto, como el edificio o área física donde se encuentran las oficinas del mismo). Esta red, no debe confundirse con la “red interna” (debido a la implementación del DMZ). A esta red se encuentran conectados los diferentes Servidores que son administrados por la Dirección de Tecnologías de Información y Comunicación (DTIC), y sobre los cuales funcionará el Software y su Base de Datos.
- **Firewall:** Es un dispositivo de red, utilizado para bloquear el acceso no autorizado, permitiendo al mismo tiempo comunicaciones autorizadas, evitando que los usuarios de Internet no autorizados tengan acceso a la red privada conectada. La utilización del firewall es imprescindible para conformar un DMZ, ya que en el DMZ se ubican los servidores de la organización que deben permanecer accesibles desde la red exterior.

**Dispositivos de Virtualización:**

- XEN: Es un Monitor de Máquinas Virtuales, que también se denomina como Hipervisor. Es una plataforma que permite aplicar diversas técnicas de control de virtualización para utilizar diferentes Máquinas Virtuales, es decir, simular la presencia de diferentes equipos físicos (servidores en este caso), que en realidad son virtuales, en una misma computadora.
- Máquinas Virtuales: En el Diagrama de la Figura 11-23, se muestran tres máquinas virtuales: Un Servidor de Aplicaciones y dos Servidores de Base de Datos. Estas son virtualizadas sobre los servidores físicos PowerEdge R815 y controladas por el Hipervisor XEN. Una Máquina Virtual (VM) consta como mínimo de un Disco Virtual con la instalación de un Sistema Operativo y su configuración (memoria RAM, número de procesadores, entre otras). En estas, se encuentra instalado el Sistema Operativo sobre el cual corren los programas y aplicaciones necesarias para que el software del SIM funcione correctamente, al cual el usuario accede virtualmente.

Software:

- Apache 2: Representa el Servidor Web (el cual ya ha sido definido anteriormente en la sección 11.3.1 del presente documento) del SIM, instalado en el Servidor de Aplicaciones.
- PostgreSQL: Representa el SGBD (el cual ya ha sido definido anteriormente en la sección 11.3.1 del presente documento) instalado en el Servidor de Base de Datos. Este Servidor de Base de Datos se encuentra virtualizado en el mismo servidor físico que el Servidor de Aplicaciones.
- SQL Server: Representa el SGBD de Microsoft (el cual ya ha sido definido anteriormente en la sección 11.3.1 del presente documento) instalado en el Servidor de Base de Datos. Este Servidor de Base de Datos se encuentra virtualizado en otro servidor físico, diferente al del SIM.

Artefactos:

- Navegador Web: Representa el navegador web (definido en la sección 11.3.1 del presente documento) que ejecuta el usuario en su Máquina Cliente.
- Aplicación SIM: Representa los diferentes elementos propios de la aplicación, construidos en base al Framework Symfony, y que realizarán los procesos principales del negocio. Es el código fuente del Software que se ejecuta en el Servidor Web Apache.
- Base de Datos SIM, SIAF y SINAB: Representa las Bases de Datos (definidas en la sección 11.3.1 del presente documento) en donde se almacenarán y recuperarán los datos del SIM, y desde donde se extraerá información de activo fijo (BD SIAF) e información sobre los vales de salida y costos de materiales y repuestos (BD SINAB).



Como ya se ha mencionado, el Diagrama de la Figura 11-23 considera la relación de los componentes de software y hardware del Sistema y su distribución, pero no su localización en el mundo real.

Para compensar esta situación, se presenta en la Figura 11-24, un esquema más detallado del despliegue físico del Sistema Informático de Mantenimiento y la localización de sus elementos.

Además de los elementos ya descritos en el Diagrama de la Figura 11-23, la Figura 11-24 presenta los siguientes: Los servidores se encuentran en una Zona Desmilitarizada o **DMZ** (Para conocer más sobre la definición de DMZ, vea el Glosario Técnico), para que puedan ser accesibles desde la red externa, en este caso de la **WAN**. Ya que a la WAN se interconectan los diferentes **Establecimientos del MINSAL**, dentro de los cuales están los Departamentos de Mantenimiento y las diferentes Dependencias del MINSAL que necesitan los servicios de mantenimiento, estos podrán acceder a la aplicación del SIM, realizando su trabajo a diario. Mientras que los Administradores Generales del SIM, que son personas que laboran en la Dirección de Tecnologías de Información y Comunicación (**DTIC**), tal como su nombre lo indica, se encargaran de administrar y darle mantenimiento al Software y la Base de Datos. Para el desarrollo del proyecto, al Equipo de Trabajo se le ha proveído acceso **SSH** (Para conocer la definición de SSH, ver Glosario Técnico) hasta el Servidor de Aplicaciones del SIM, que es una Máquina Virtual controlada por el Hipervisor **XEN**, montada sobre un Servidor Físico PowerEdge R815 dentro de la DMZ. Esto con el fin de realizar copias y actualizaciones del proyecto (código fuente, archivos y base de datos) desde cualquier localización, siempre y cuando se tenga acceso a internet. El acceso a internet tanto del MINSAL, como del equipo de desarrollo, es un servicio brindado por el **ISP** (Proveedor de Servicios de Internet por sus siglas en inglés) CTE Telecom Personal S.A. de C.V.

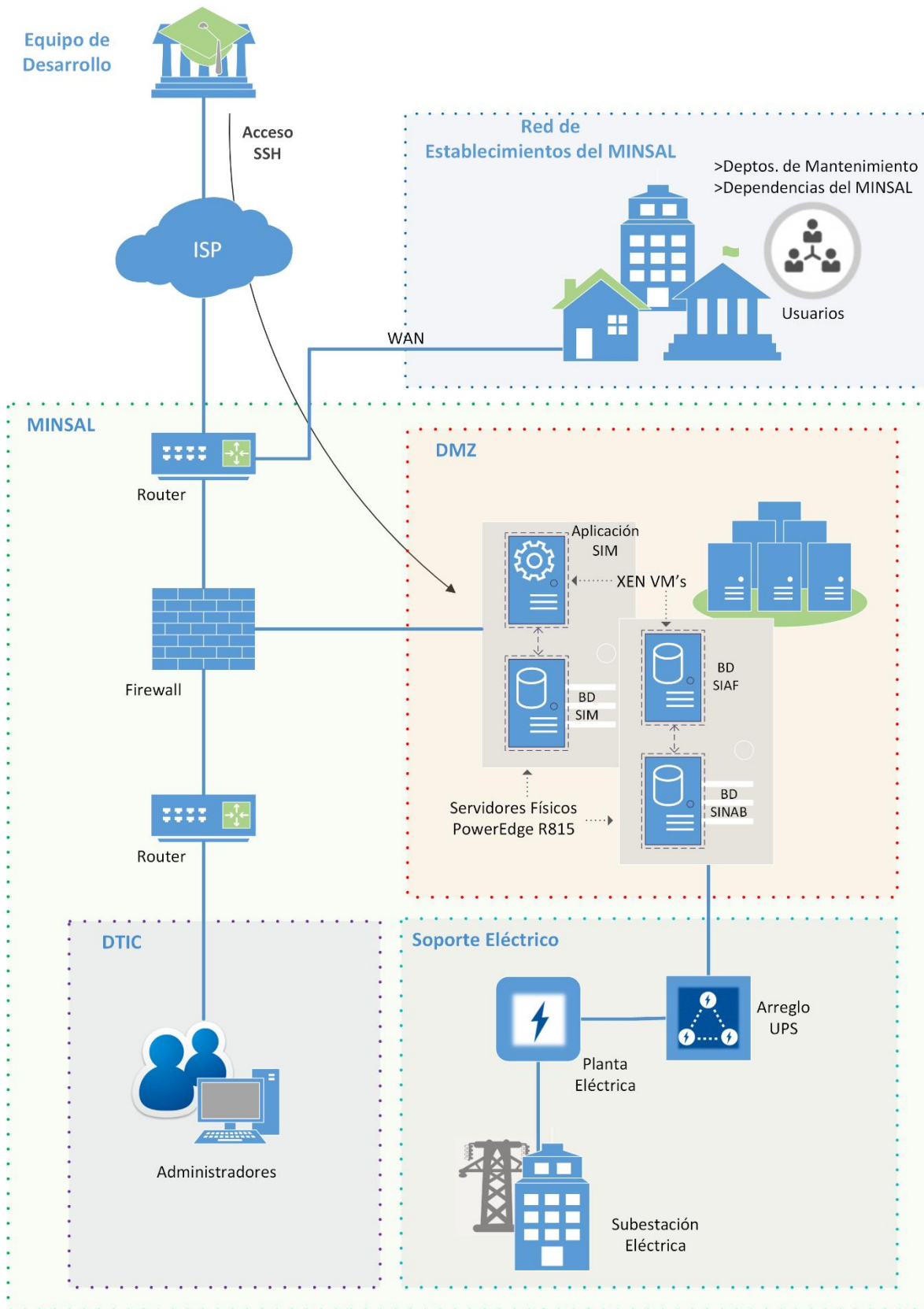


Figura 11-24. Esquema del despliegue físico del SIM.

11.3.7.2 Modelo Lógico⁷².

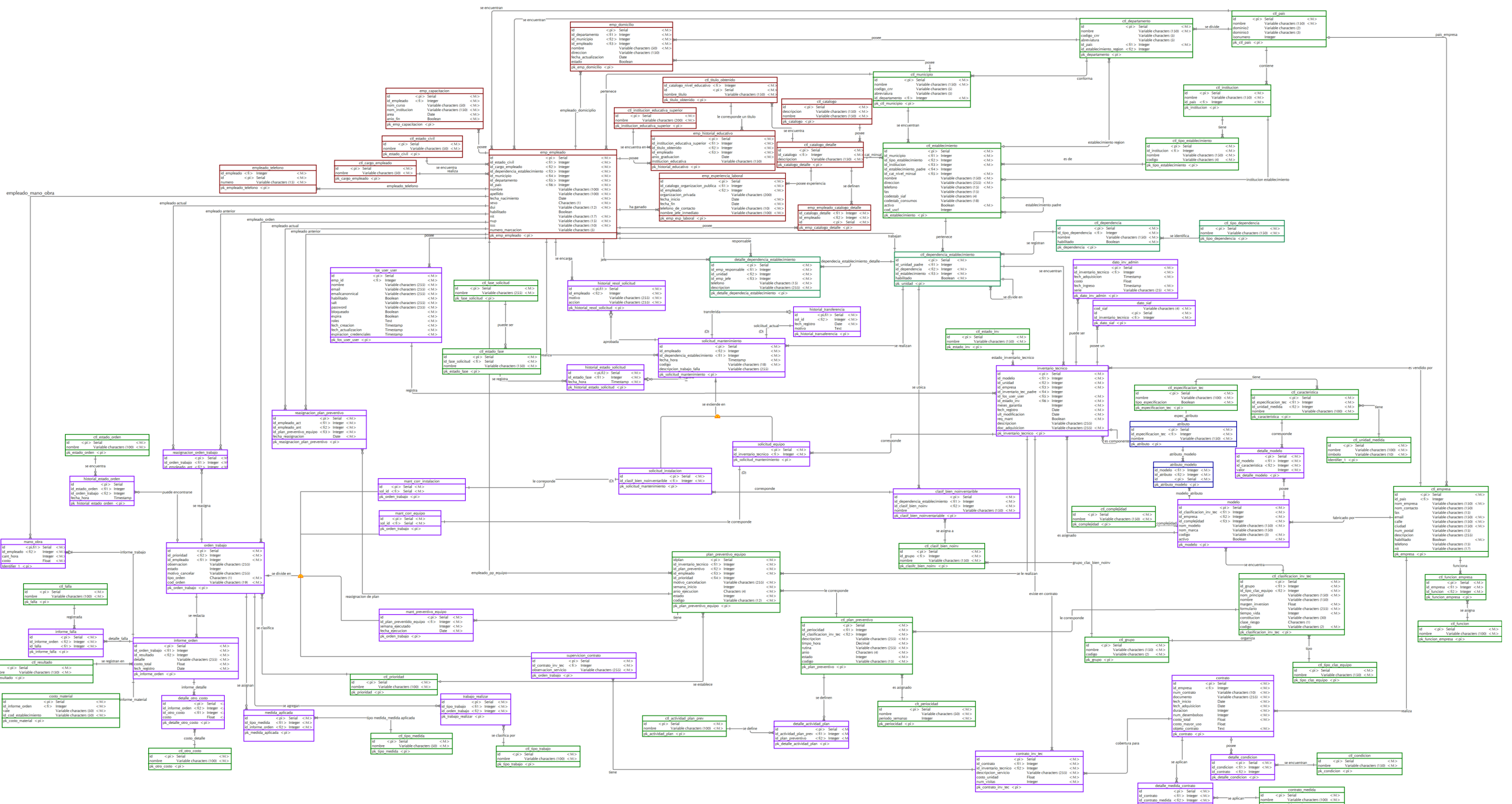


Figura 11-26 Modelo Lógico.

⁷² Para ver el Diagrama del Modelo Lógico del Sistema SIM en Formato de Imagen, abrir archivo "Modelo Logico", que se encuentra dentro del directorio /Documentos/3 – Diseño/Modelo Logico/ del DVD anexo.



11.3.8 Conclusión Sobre Diseño Arquitectónico

Para desarrollar el diseño de un sistema, desde su concepción básica hasta el diseño detallado orientado a objetos, existen diferentes puntos de vista y opiniones según diversos autores. Según Sommerville, entre las actividades por hacer para lograr un diseño orientado a objetos con el uso de UML, están:

1. Comprender y definir el Contexto y las Interacciones Externas con el Sistema.
2. Diseñar la Arquitectura del Sistema.
3. Identificar los objetos principales en el Sistema.
4. Desarrollar los Modelos de Diseño.

Hasta este punto, se ha definido el Contexto y las Interacciones Externas con el Sistema en la sección 11.2 del presente documento. En la que se presenta las relaciones entre el Sistema que se diseñara y el ambiente externo. Esto es esencial para definir cómo el sistema proporcionará la funcionalidad requerida y como estructurarlo para lograrlo.

Se ha diseñado la Arquitectura del Sistema, cuyo desarrollo también está lleno de diversos puntos de vista, opiniones y consejos de cómo realizarlo. Sin embargo, no hay una guía concreta y correcta de cómo hacerlo y cómo saber cuándo la arquitectura está verdaderamente completa. Por tanto, a partir de una recopilación, investigación y análisis de diversas fuentes (las cuales aparecen en la Bibliografía), se ha guiado el desarrollo de la misma. Diversos autores concuerdan que para realizar el Diseño Arquitectónico se necesita de múltiples Vistas, ya que es imposible representar toda la información relevante sobre la arquitectura de un sistema en un solo modelo arquitectónico, pues cada Vista presenta únicamente una perspectiva del sistema⁷⁴. Por ejemplo, el modelo “4 + 1” de Philippe Kruchten⁷⁵ propone el uso de cuatro vistas arquitectónicas fundamentales: Vista Lógica, Vista de Proceso, Vista de Desarrollo y Vista Física.

La Vista Lógica, describe las partes del modelo de diseño significativas arquitectónicamente, tales como su descomposición en subsistemas y paquetes.

La Vista de Proceso, muestra como en el tiempo de operación, el Sistema está compuesto de procesos de interacción.

La Vista de Desarrollo, muestra como el Software está descompuesto en elementos que facilitan el desarrollo.

La Vista Física, describe una o más configuraciones físicas (hardware) sobre las cuales el software es distribuido y ejecutado.

Para brindar la Vista Lógica del Sistema se han presentado los siguientes Diagramas:

- El Diseño Arquitectónico a Nivel de Paquetes (Figura 11-16).
- La Descripción del Modelo de Tres Capas y el Patrón MVC (Figura 11-17 y la Figura 11-18).
- El Funcionamiento Interno de Symfony (Figura 11-20).

⁷⁴ Sommerville, I. (2011). Ingeniería de Software (9a ed.), 6.2 Vistas arquitectónicas. Pearson Education.

⁷⁵ Kruchten, P. (1995). The 4 + 1 View Software Architecture. Rational Software Corp.



Para brindar la Vista de Procesos se ha presentado el siguiente Diagrama:

- Diagrama de Secuencia del Patrón MVC (Figura 11-19).

Para brindar la Vista de Desarrollo se han presentado los siguientes Diagramas:

- Diagrama de Componentes del Sistema (Figura 11-21).

Para brindar la Vista Física se han presentado los siguientes Diagramas:

- Diagrama de Despliegue del Sistema (Figura 11-23).
- Esquema del Despliegue Físico del SIM (Figura 11-24).

Además de las cuatro vistas ya mencionadas, se ha definido la Vista de Datos, que muestra la Arquitectura de las estructuras de datos que utilizará el Sistema. La Vista de Datos se puede observar en el apartado 11.3.7, el cual se compone del Diagrama Entidad - Relación, Modelo Lógico y Físico de la Base de datos.

Todas estas Vistas constituyen el Diseño Arquitectónico del Sistema Informático de Mantenimiento. Sin embargo, este aún no está completo. Para completar la Arquitectura de Sistema, es necesario realizar los Modelos de Diseño. Como ya se ha mencionado, el Diseño Arquitectónico se realiza de manera iterativa hasta que se obtiene una estructura arquitectónica completa. Los Modelos de Diseño, concluirán todo el Diseño del Sistema.

Por tanto, un paso importante en el proceso de diseño es definir los Modelos de Diseño a un mayor nivel detalle que el que se ha presentado hasta el momento. Al utilizar UML para el diseño del Sistema, se especificaran dos tipos de modelos de diseño: Los Modelos Estructurales y los Modelos Dinámicos.

Tal como se ha especificado en la sección 8.6.2 del presente documento, se utilizaran los Modelos Dinámicos: Diagramas de Casos de Uso, Diagramas de Secuencia y Diagramas de Actividad. Así como los Modelos Estructurales: Diagramas de Clases y Diagramas de Componentes (este último ya utilizado); para especificar de forma detallada, en cada incremento (según la Metodología de Desarrollo de Software definida en la sección 8.4.1 del presente documento), el Diseño Orientado a Objetos del Sistema y completar de esta forma la Arquitectura del Sistema Informático de Mantenimiento.



11.4 Diseño de Seguridad

11.4.1 Registro de Usuarios y Activación mediante Correo Electrónico.

El sistema tendrá la capacidad de registrar nuevos Usuarios del Sistema, los cuales pertenecerán a cada uno de los empleados, es decir, un Usuario estará única y exclusivamente asociado a un Empleado. Permitiendo de esta forma, conocer las acciones de cada Empleado, mediante su Usuario, en el Sistema.

El Administrador General, los Administradores de Establecimiento y los Jefes de Mantenimiento, serán los encargados de crear Usuarios. Al momento de su creación, el Administrador, solamente necesitará ingresar el Nombre de Identificación del Usuario (conocido en inglés como “Username”) y el Correo Electrónico al cual se le enviarán las notificaciones. Así mismo, definirá el Grupo al que pertenece el Usuario y de ser necesario, Roles específicos para dicho Usuario (La administración de Grupos y Roles se explica a continuación en la sección 11.4.2).

Cuando se cree el Usuario, el Sistema enviará automáticamente al Correo Electrónico proporcionado por el Empleado, una notificación con un enlace de activación de cuenta, para que dicho Empleado confirme y habilite su cuenta. Al habilitar o activar la cuenta, el Sistema generará una contraseña aleatoria (la generación de contraseñas se explica en la sección 11.4.3), y nuevamente, se notificará mediante el Correo Electrónico provisto, las credenciales de ingreso al Sistema (Nombre de Usuario y Contraseña generada).

De esta forma, los nuevos Usuarios podrán ingresar al Sistema, sin que el Administrador o cualquier otra persona, conozca sus credenciales.

El proceso anteriormente descrito puede observarse de forma gráfica en la Figura 11-28, en dónde los pasos que intervienen se resumen a continuación:

- 1) Creación de usuarios por parte de los Administradores o Jefe de Mantenimiento.
- 2) El sistema envía enlace de activación al usuario.
- 3) El usuario revisa su cuenta de correo y activa su cuenta de usuario con el enlace recibido.
- 4) El sistema notifica la activación exitosa del nuevo usuario y se envían en un nuevo correo que contendrá las credenciales para poder entrar al sistema.
- 5) El usuario recibe las credenciales en su cuenta de correo.

En la Figura 11-28, se pueden observar los algoritmos y métodos de codificación que se utilizan para obtener contraseñas seguras, los cuáles se explican más a fondo en la sección 11.4.3.

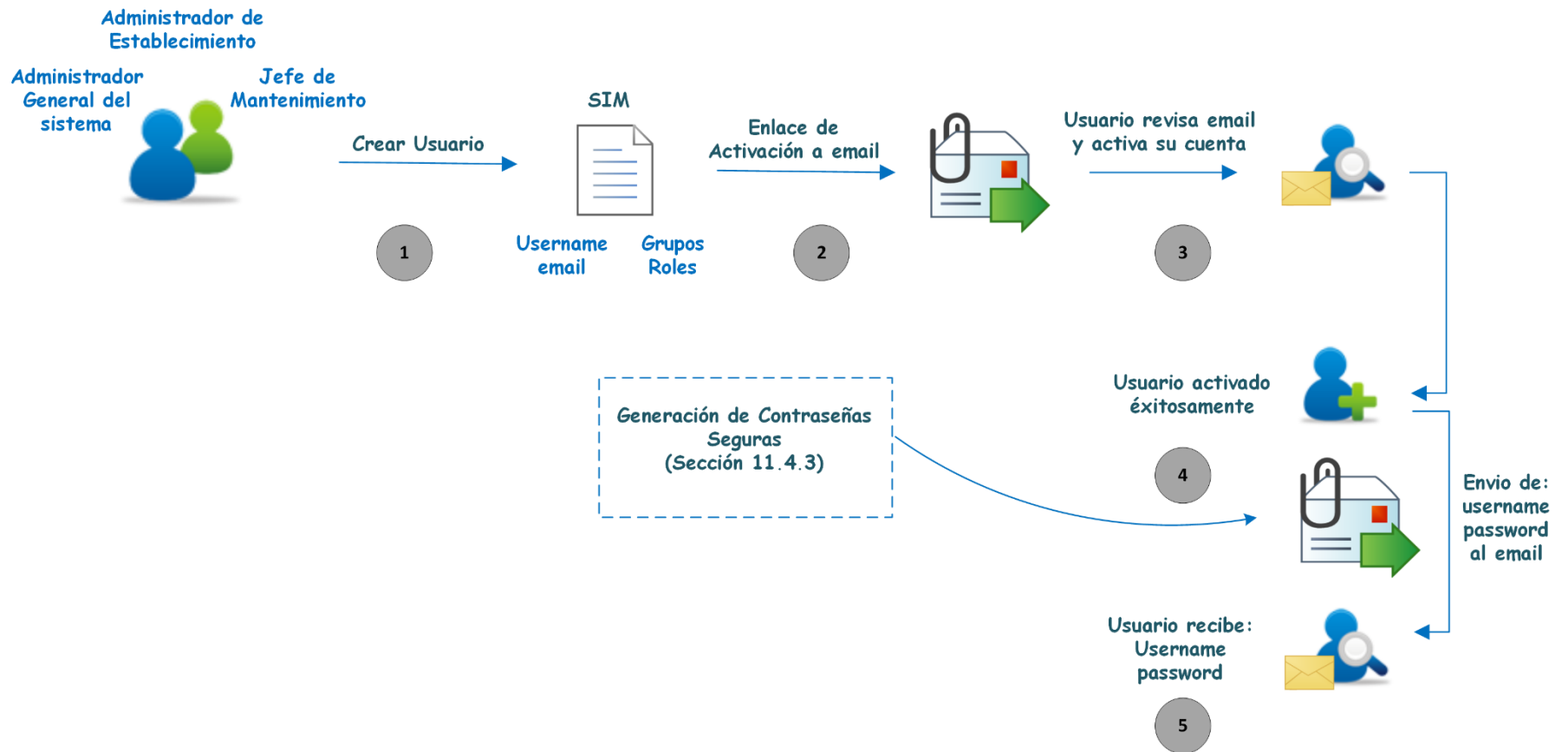


Figura 11-28 Proceso de Creación y Activación de Usuarios



11.4.2 Privilegios Mediante Grupos y Roles.

La manera de permitir a un Usuario realizar acciones dentro del Sistema, es mediante Roles. Los Roles determinan que es lo que le está permitido ver, agregar, modificar, eliminar o conocer al Usuario. Por esta razón también se le conocen como privilegios de Usuario.

Cuando un Usuario es creado, hay dos formas de otorgarle privilegios:

- **Roles:** Un Rol, en este contexto, es una acción específica que un Usuario puede realizar. Por ejemplo: Registrar un Nuevo Equipo, Realizar (crear) una Solicitud o Consultar una Orden de trabajo.
- **Grupos:** Un Grupo, en este contexto, es un conjunto de Roles. El objetivo de un Grupo, es definir una serie de Roles que constantemente serán asignados a diversos Usuarios, y que juntos conforman una agrupación que se administra más fácilmente dentro del Sistema. Por ejemplo, se podría crear un Grupo llamado “Secretaria”, el cual agrupará los Roles: Consultar Solicitudes, Consultar Ordenes de Trabajo y Consultar Datos de Empleados. Como es de esperar, habrán muchos Usuarios que realizaran el papel de “Secretaria”, y el beneficio es que los Usuarios solamente se asocian con el Grupo “Secretaria”, en vez de otorgar (por cada uno) cada uno de los Roles que una “Secretaria” podría tener.

Los Actores definidos en la sección 10.3.2.1 de este documento (Clasificación de Usuarios), se implementaran entonces mediante Grupos, es decir, que según los Actores definidos, se crearan Grupos conformados por las acciones (Roles) que los actores del sistema pueden realizar.

De igual forma, si un Usuario necesita un privilegio especial o específico para él, no será necesario cambiar o modificar el Grupo al cual está asociado, sino solo otorgarle o añadirle (al Usuario que lo necesita, no al Grupo) el Rol. Por ejemplo, si un Usuario que juega el papel de “Secretaria” necesita en algún dado momento Realizar Solicitudes, pero no se desea otorgar este privilegio a todos los Usuarios que juegan el papel de “Secretaria”.

11.4.3 Generación de Contraseñas Seguras.

Cuando un Empleado activa o habilita su cuenta de Usuario, mediante un enlace enviado a través de un correo electrónico, el Sistema generará una contraseña aleatoria, y nuevamente, se notificará mediante el correo electrónico provisto, las credenciales de ingreso al Sistema.

La contraseña generada, se creara mediante la clase TokenGenerator del Bundle FOSUSER, y tendrá una longitud final de 50 caracteres.

El método generateToken de esta Clase realiza los siguientes pasos, para poder brindar una contraseña segura:



1. Si la extensión **Open SSL** de PHP está habilitada:

- a) Se utiliza la función de PHP llamada **openssl_random_pseudo_bytes**, la cual genera una cadena de bytes pseudo-aleatoria:

```
string openssl_random_pseudo_bytes (int $length [, bool &$crypto_strong ] )
```

En donde:

\$length: es la longitud de la cadena de bytes deseada. Debe ser un entero positivo. PHP intentará asignar este parámetro a un entero no nulo para usarlo. En este caso, generateToken utiliza el tamaño de 32 bytes.

\$crypto_strong: Si se pasó en la función, mantendrá un valor booleano que determina si el algoritmo usado fue "criptográficamente fuerte", por ejemplo, si es seguro para el uso con contraseñas. TRUE si es fuerte, de otro modo es FALSE.

- b) Si la cadena de bytes generada por la función `openssl_random_pseudo_bytes` es "criptográficamente fuerte", es decir que la variable `$crypto_strong` es true, esta se codifica en Base64.

- c) Codificación Base64: La codificación Base64 se realiza mediante la función de PHP llamada **base64_encode**:

```
string base64_encode ( string $data )
```

En donde:

\$data: es la cadena a codificar, que en este caso sería la cadena de bytes pseudo-aleatoria generada por la función anterior.

La función entonces codifica `$data` en base64. Este tipo de codificación está diseñado para que datos binarios sobrepasen capas de transporte que no son de 8-bits al 100%, por ejemplo el cuerpo de un e-Mail. La codificación en Base64 hace que los datos sean 33% más largos que los datos originales.

- d) Finalmente se elimina cualquier carácter "=" de la cadena de caracteres resultante.

2. Si la extensión **Open SSL** de PHP no está habilitada:

- a) Se utiliza la función **mt_rand**, la cual genera un número entero aleatorio de 10 caracteres.
 b) Se utiliza la función **uniqid(\$prefijo, true)**, para generar una cadena de 23 caracteres cuyo prefijo será el número entero aleatorio generado por **mt_rand**.
 c) Se codifica la cadena de caracteres generada por **uniqid** en hash, mediante la función **hash**, utilizando el algoritmo **sha256**.

```
string hash ( string $algo , string $data [, bool $raw_output = false ] )
```




En donde:

\$algo: Es el nombre del algoritmo de cifrado seleccionado (es decir "md5", "sha256", "haval160,4", etc.), que en este caso será sha256.

\$data: es la cadena a cifrar.

\$raw_output: Cuando se establece en TRUE la salida serán datos binarios sin formato, FALSE la salida serán dígitos hexadecimales en minúsculas, que en este caso será TRUE.

d) Finalmente se elimina cualquier carácter "=" de la cadena de caracteres resultante.

11.4.4 Cifrado de Contraseñas en la Base de Datos.

Aunque se generen contraseñas seguras como se vio en la sección anterior (11.4.3), y tomando en cuenta que los Usuarios podrán cambiar sus contraseñas generadas por defecto a contraseñas más fáciles de recordar (por ejemplo, "Mi-p4ssw0rd"), estas no se deben almacenar como texto plano en la Base de Datos, para evitar la observación directa de la misma por las personas que tengan acceso a dichos datos.

Para ello se utiliza la configuración de seguridad de Symfony, mediante el parámetro "encoders", específicamente en el "archivo security.yml" dentro del directorio "app/config".

```
encoders:  
    FOS\UserBundle\Model\UserInterface: sha512
```

Por medio de esta configuración y el uso del algoritmo sha512, el Sistema automáticamente cifrará mediante hash el Password del usuario 5000 veces y luego lo codificará en Base64.

11.4.5 Inicio de Sesión de Usuario e Inactividad.

Los Usuarios solo podrán ingresar al Sistema mediante el previo Inicio de Sesión (conocido en inglés como Login), especificando su Nombre de Usuario y Contraseña. De lo contrario no podrá realizar ninguna otra acción.

En caso de que se ingrese la dirección URL de algún recurso o página web del Sistema de forma directa, y el Usuario no ha iniciado sesión, este será redirigido a la página de Inicio de Sesión.

Adicionalmente, el Sistema verifica la inactividad del Usuario, que es el evento que inicia desde el momento en que el usuario deja de realizar alguna actividad en el sistema. Por motivos de seguridad se ha establecido en el sistema un periodo de inactividad, si el usuario alcanza al límite del tiempo del periodo de inactividad, el sistema cierra la sesión del usuario de forma automática.



11.4.6 Proceso de Seguridad Lógica del Sistema.

El proceso de seguridad lógica del sistema está basado en el sistema de seguridad de Symfony, el cual consta en identificar primero al usuario (**autenticación**) y comprobando después si ese usuario tiene acceso al recurso solicitado (**autorización**).

Autenticación

El sistema de seguridad de Symfony se activa cuando un usuario hace una petición (request) a una URL que está protegida por un firewall o cortafuegos. El trabajo del firewall consiste en determinar si el usuario necesita estar autenticado, y si lo necesita, enviar una respuesta al usuario para iniciar el proceso de autenticación.

Autorización

Los roles son la clave del sistema de autorización: un usuario puede acceder a un recurso específico solicitado, sólo si cuenta con el rol necesario, por ejemplo: ROLE_ADMIN.

Para comprender fácilmente cómo funcionan la autenticación y autorización, se puede observar en la Figura 11-29, la manera en que un cliente (usuario) realiza una petición (request) y al final obtiene su respectiva respuesta (response):

- 1) Inicialmente un cliente realiza una petición de un recurso específico.
- 2) El firewall de autenticación es activado y solicita las credenciales respectivas.
- 3) El archivo AdminSecurityController.php permite comparar las credenciales registradas por el cliente con las almacenadas en la base de datos del SIM, las cuales se encuentran encriptadas específicamente en la tabla fos_user_user.
- 4) Luego de pasar la etapa de **autenticación** exitosamente, se activa la etapa de **autorización** en donde se verifica si el usuario posee los roles o pertenece al grupo necesario para poder acceder al recurso solicitado, esto es apoyado por MenuBuilder.php, que es donde se verifican los roles y en base a esto se muestran los recursos respectivos (opciones de menú).
- 5) Para finalizar se realiza una respuesta en donde se muestra el recurso solicitado, lo cual se ilustra como archivos html, css y javascript.

Adicionalmente la forma de autenticación y las configuraciones respectivas que symfony adopta sobre el manejo de roles, rutas a recursos, firewall de seguridad, las importa por medio del archivo config.yml, el cual se apoya de: routing.yml, parameters.yml, y security.yml.

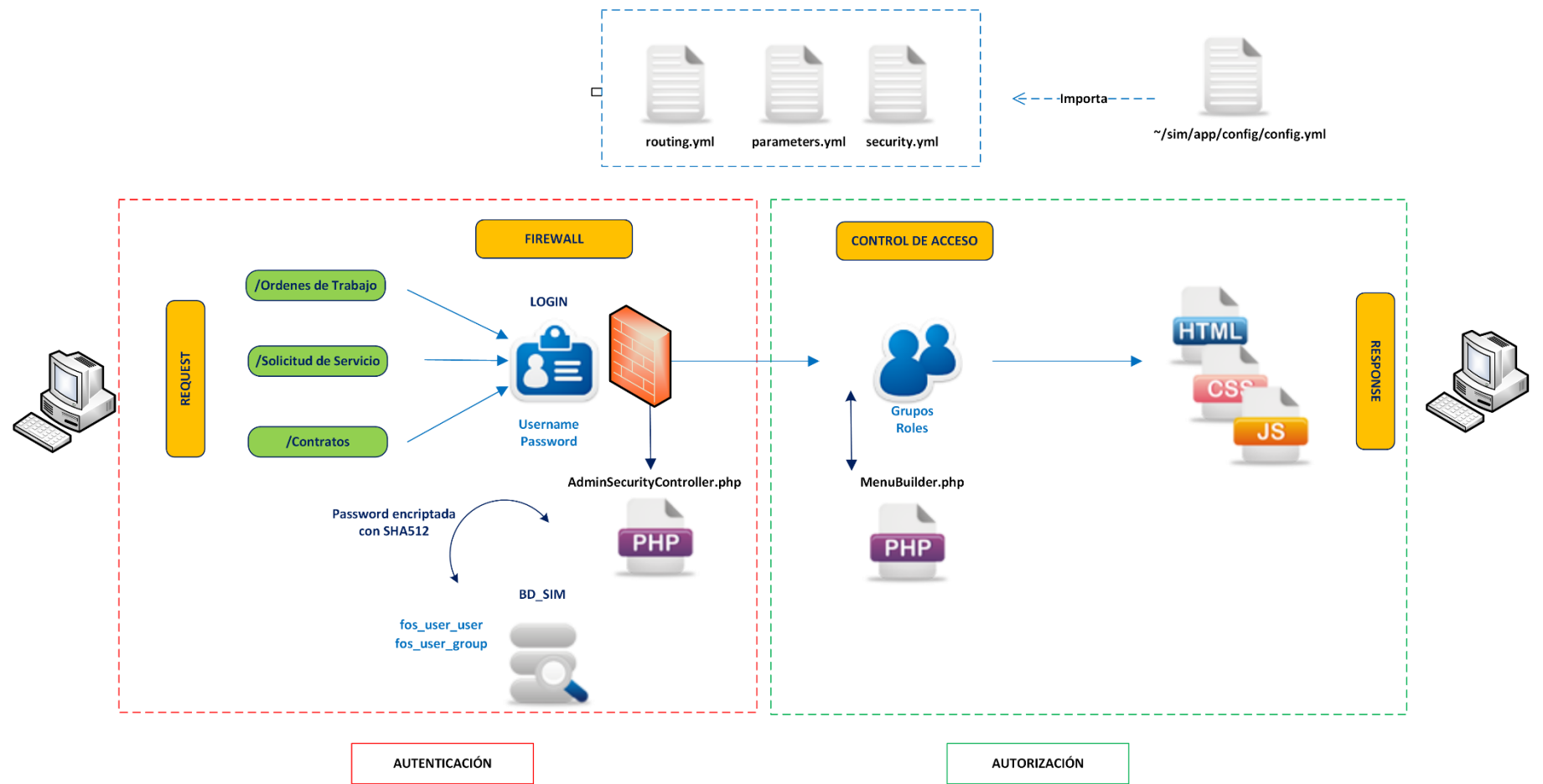


Figura 11-29 Proceso de Seguridad Lógica del Sistema



12 Capítulo VI

Incrementos de Software

El desarrollo incremental, es un modelo de tipo evolutivo, que se basa en diseñar una implementación o versión inicial del sistema, exponerla a la opinión y criterio del usuario, y luego desarrollar las demás implementaciones en sus diversas versiones (realizando correcciones y agregando nuevas funcionalidades o características), hasta producir el sistema completo. A cada una de estas versiones se le conoce como Incrementos de Software.

En el Diseño Preliminar (Capítulo anterior) se realiza una descripción de la arquitectura del sistema a implementar, los modelos y la estructura de datos, las interfaces con otros componentes del sistema, entre otros. En este no se realiza un diseño detallado del sistema, el diseño total del sistema será realizado de manera iterativa a medida se vayan desarrollando los Incrementos de Software.

En este Capítulo se realiza el Diseño Detallado de cada Incremento, que parte del Diseño Preliminar, realizando un diseño más profundo de manera iterativa hasta que se obtiene una estructura arquitectónica completa. Los Modelos de Diseño, concluirán todo el Diseño del Sistema Informático de Mantenimiento.

Así mismo, tal como lo especifica el Modelo de Desarrollo de Software Incremental, el Diseño detallado en cada Incremento, se traduce en una versión más del Sistema, que posee nuevas y mejores funcionalidades y características. Esto se realiza, mediante la Etapa de Programación en cada Incremento, dando como resultado una nueva versión del Código Fuente del Software, siguiendo los Estándares previamente definidos.



12.1 Incrementos de Software

Cada Incremento de Software, da cumplimiento a uno o varios requerimientos. En este sentido, cada Incremento se enfoca en dar soporte a determinada funcionalidad, al finalizar su desarrollo.

Por esta razón, los requerimientos son asignados a un determinado Incremento de Software, y por ello la realización de estos no es al azar, sino, que es el resultado de un análisis adecuado para organizarlos y ordenarlos, de manera que garantice la exitosa realización de los mismos. Esto se debe a que algunos elementos del Sistema dependen de la presencia o existencia de otros.

Los Incrementos iniciales del Software, abarcan el desarrollo de aquellos elementos que constituyen la base para lograr la funcionalidad deseada del Sistema. Teniendo asegurada la funcionalidad básica del Sistema, se procede al desarrollo de los Incrementos de Software que dan soporte a los procesos de negocio de la organización.

Como se ha mencionado, se realiza un Diseño Detallado en cada Incremento, que parte del Diseño Preliminar (mostrado en el Capítulo anterior), realizando un diseño más profundo de manera iterativa hasta que se obtiene una estructura arquitectónica completa. Y finalmente, el Diseño detallado en cada Incremento, se traduce en una versión más del Sistema, que posee nuevas y mejores funcionalidades y características. Esto se realiza, mediante la Etapa de Programación en cada Incremento.

Para cada Incremento de Software se realizan las siguientes actividades:

1. Descripción Extendida de Casos de Uso.
2. Diseño de Interfaces.
3. Diagramas de Secuencia.
4. Diagramas de Actividad (con la notación BPMN).
5. Diagrama de Clases (Considerando las Clases que corresponden al Incremento).
6. Diseño de Casos de Prueba.
7. Ejecución de Casos de Prueba.
8. Resumen de Resultados de Pruebas.

A continuación se presenta cada uno de los Incrementos de Software del SIM:

Nota: Para el Primer Incremento de Software en este documento, se presenta de forma resumida las actividades que conlleva el desarrollo de cada Incremento (numerales 1-7 del punto anterior). Mientras que para los Incrementos siguientes (del Segundo al Séptimo Incremento), solamente se presenta la descripción de las funcionalidades desarrolladas en cada uno. Para de forma completa el desarrollo de cada Incremento, vea el documento de Etapa 1, dentro del directorio /Documentos/4 – Etapa 1/ , y el documento de Etapa 2, dentro del directorio /Documentos/5 – Etapa 2/ , ambos dentro del DVD anexo.



12.2 Primer Incremento

El Primer Incremento abarca el diseño y desarrollo de las siguientes funcionalidades y elementos:

Establecimientos: Da soporte a la administración de los Establecimientos del Ministerio de Salud y sus respectivos Tipos de Establecimientos, que van desde Unidades Comunitarias de Salud Familiar (UCSF), Hospitales Nacionales, Regiones y otros Centros Administrativos o de Salud pertenecientes al MINSAL.

Dependencias: Involucra la administración de cada una de las Unidades, Departamentos, Servicios Hospitalarios, Áreas, entre otros similares, que conforman cada Establecimiento del Ministerio de Salud. El desarrollo de esta funcionalidad, permite construir la Estructura Orgánica de un Establecimiento del MINSAL dentro del Sistema.

Personal: Abarca la gestión y manejo del personal perteneciente a los establecimientos del MINSAL y que estarán involucrados en la utilización del Sistema, a los cuales posteriormente se le brindara un usuario dentro del mismo. Incluye tanto a empleados comunes, como a los empleados de las áreas de mantenimiento, de quienes se solicita mayor información que los empleados comunes.

Es importante mencionar, que las funcionalidades correspondientes al Primer Incremento, se apegan totalmente a los requerimientos y directrices brindadas por la DTIC, en cuanto a la estandarización de los Catálogos Generales expresados en la Tabla 10-7.

12.2.1 Descripción Extendida de Casos de Uso.

12.2.1.1 Casos de Uso de Gestión de Tipos de Establecimiento.

Código:	CUS-01-01	Nombre:	Consultar Tipo Establecimiento.
Actores:	Administrador General del Sistema.		
Propósito:	Consultar datos de los Tipos de Establecimiento.		
Descripción:	Se presenta los tipos de establecimientos registrados.		
Precondiciones:	<ol style="list-style-type: none"> 1. El usuario debe haberse autenticado. 2. Debe haber seleccionado la opción Tipo de Establecimiento. 3. Existe al menos un registro de Tipo Establecimiento. 		
Post-Condiciones:	Se muestra exitosamente los datos de un registro de Tipo de Establecimiento.		
Flujo Básico			
Paso	Acción		
1	El sistema muestra un listado de los Tipos de Establecimiento registrados (Figura 12-1).		
2	El usuario selecciona un Tipo de Establecimiento de la lista (numeral 1 de la Figura 12-1), dando clic en alguno de los Nombres de Tipo de Establecimiento.		
3	El sistema muestra los datos del Tipo de Establecimiento seleccionado, tal como se muestra en la (Figura 12-1).		
4	Fin del caso de uso.		



Flujos Alternos	
Paso en el que fue invocado	Acción
2	A. Búsqueda por filtro de información. <ol style="list-style-type: none"> 1. El usuario ingresa los datos correspondiente a los campos del filtro por los cuales desea realizar la búsqueda. 2. El usuario da clic en el botón Filtrar (Numeral 4 de la Figura 12-1). 3. El sistema filtra los resultados y continúa con el paso 2 del flujo básico.
Frecuencia Esperada:	Anual
Importancia:	Moderada
Urgencia:	Moderada
Referencias Cruzadas:	RFU-09



Código:	CUS-01-02	Nombre:	Agregar Tipo Establecimiento.
Actores:	Administrador General del Sistema.		
Propósito:	Agregar un nuevo Tipo de Establecimiento.		
Descripción:	El usuario realiza el registro de un nuevo tipo de establecimiento.		
Precondiciones:	<ol style="list-style-type: none"> 2. El usuario debe haberse autenticado. 3. Debe haber seleccionado la opción Tipo de Establecimiento. 		
Post-Condiciones:	Se creó un nuevo registro de Tipo de establecimiento.		
Flujo Básico			
Paso	Acción		
1	El sistema muestra un listado de los Tipos de Establecimiento registrados (Figura 12-1).		
2	El usuario da clic en el botón Agregar nuevo (Numeral 6 de la Figura 12-1).		
3	El sistema muestra la pantalla de creación de un nuevo Tipo de Establecimiento (Figura 12-3) y el ingreso de los siguientes datos: <ul style="list-style-type: none"> • Nombre del nuevo Tipo Establecimiento (Numeral 1 de la Figura 12-3). • Código de Tipo de Establecimiento (Numeral 2 de la Figura 12-3). 		
4	El usuario ingresa la información solicitada.		
5	El usuario da clic en el botón Crear (Numeral 3 de la Figura 12-3)		
6	El sistema valida los datos ingresados.		
7	Se crea el nuevo Tipo de Establecimiento y muestra un mensaje de éxito al usuario.		
8	Fin del caso de uso.		
Flujos Alternos			
Paso en el que fue invocado	Acción		
5	A. El usuario da clic en el botón Crear y regresar (Numeral 4 de la Figura 12-3). <ol style="list-style-type: none"> 1. El sistema realiza los pasos 6 y 7 del flujo básico. 2. El sistema despliega el listado de los grupos registrados (numeral 1 de la Figura 12-3). B. El usuario da clic en el botón Crear y agregar otro (Numeral 5 de la Figura 12-3). <ol style="list-style-type: none"> 1. El sistema realiza los pasos 6 y 7 del flujo básico. El sistema continúa con el paso 3 del flujo básico. 		
6	B. Error de validación. <ol style="list-style-type: none"> 1. El sistema muestra un mensaje de error debido a que alguno de los datos requeridos se ha ingresado incorrectamente o no se ha especificado. 2. El sistema continúa con el paso 4 del flujo básico. 		
Frecuencia Esperada:	Anual		
Importancia:	Alta		
Urgencia:	Moderada		
Referencias Cruzadas:	RFU-09		



Código:	CUS-01-03	Nombre:	Modificar Tipo Establecimiento.
Actores:	Administrador General del Sistema.		
Propósito:	Actualizar los datos de un Tipos Establecimiento registrado anteriormente.		
Descripción:	Se muestra un formulario de un Tipo de Establecimiento, en donde se pueden modificar los datos necesarios.		
Precondiciones:	<ol style="list-style-type: none"> 3. El usuario debe haberse autenticado. 4. Debe haber seleccionado la opción Tipo de Establecimiento. 5. Existe al menos un registro de Tipo de Establecimiento. 		
Post-Condiciones:	Se actualizó correctamente los datos de un registro de Tipo de Establecimiento.		
Flujo Básico			
Paso	Acción		
1	Se realiza el caso de uso Consultar Tipo Establecimiento.		
2	El usuario da clic en el botón Editar (numeral 1 de la Figura 12-1)		
3	El sistema se muestra la interfaz de modificación de datos (Figura 12-4).		
4	El usuario modifica los datos deseados.		
5	El usuario da clic en el botón Actualizar (numeral 3 de la Figura 12-4).		
6	El sistema valida los datos ingresados.		
7	El sistema registra la modificación de los datos del Tipo de Establecimiento y muestra un mensaje de éxito.		
8	Fin del caso de uso.		
Flujos Alternos			
Paso en el que fue invocado	Acción		
6	A. Error de validación. <ol style="list-style-type: none"> 1. El sistema muestra un mensaje de error debido a que alguno de los datos que han sido modificados es incorrecto o no ha sido especificado. 2. El sistema continúa con el paso 2 del flujo básico. 		
Frecuencia Esperada:	Anual.		
Importancia:	Moderada.		
Urgencia:	Moderada.		
Referencias Cruzadas:	RFU-09		



Código:	CUS-01-04	Nombre:	Eliminar Tipo Establecimiento.
Actores:	Administrador General del Sistema.		
Propósito:	Eliminar completamente un Tipo de Establecimiento registrado anteriormente.		
Descripción:	Se presenta un formulario para la eliminación de un Usuario.		
Precondiciones:	<ol style="list-style-type: none"> 3. El usuario debe haberse autenticado. 4. Debe haber seleccionado la opción Tipo Establecimiento. 5. Existe al menos un registro de Tipo Establecimiento. 		
Post-Condicion:	Se eliminó correctamente el Tipo de Establecimiento.		
Flujo Básico			
Paso	Acción		
1	Se realiza el caso de uso consultar Tipo Establecimiento.		
2	El usuario da clic en el botón Editar (Numeral 1 de la Figura 12-1)		
3	El sistema se muestra la interfaz de modificación de datos (Figura 12-4).		
4	El usuario da clic en el botón Eliminar (numeral 5 de la Figura 12-4).		
5	El sistema muestra un mensaje de confirmación de eliminación.		
6	El usuario confirma la acción.		
7	El registro de Tipo Establecimiento se elimina correctamente.		
8	Fin del caso de uso.		
Flujos Alternos			
Paso en el que fue invocado	Acción		
6	B. Cancelación de acción. <ol style="list-style-type: none"> 1. El usuario cancela la acción. 2. El sistema retorna al formulario actual. 		
7	B. Error de eliminación. <ol style="list-style-type: none"> 1. El sistema muestra un mensaje de error debido a que existen otros registros (Establecimientos) que hacen referencia al registro de Tipo de Establecimiento a eliminar. 		
Frecuencia Esperada:	Anual.		
Importancia:	Moderada.		
Urgencia:	Moderada.		
Referencias Cruzadas:	RFU-09		



12.2.2 Diseño de Interfaces

12.2.2.1 Interfaces de Gestión de Tipos de Establecimiento.

Interfaz de Listar Tipo de Establecimiento

DTIC Dirección de Tecnologías de Información y Comunicaciones - Ministerio de Salud

Figura 12-1. Interfaz Listar Tipo de Establecimiento.

Interfaz de Consulta de Tipo de Establecimiento

DTIC Dirección de Tecnologías de Información y Comunicaciones - Ministerio de Salud

Figura 12-2. Interfaz Consultar Tipo de Establecimiento



Interfaz de Creación de Tipo de Establecimiento

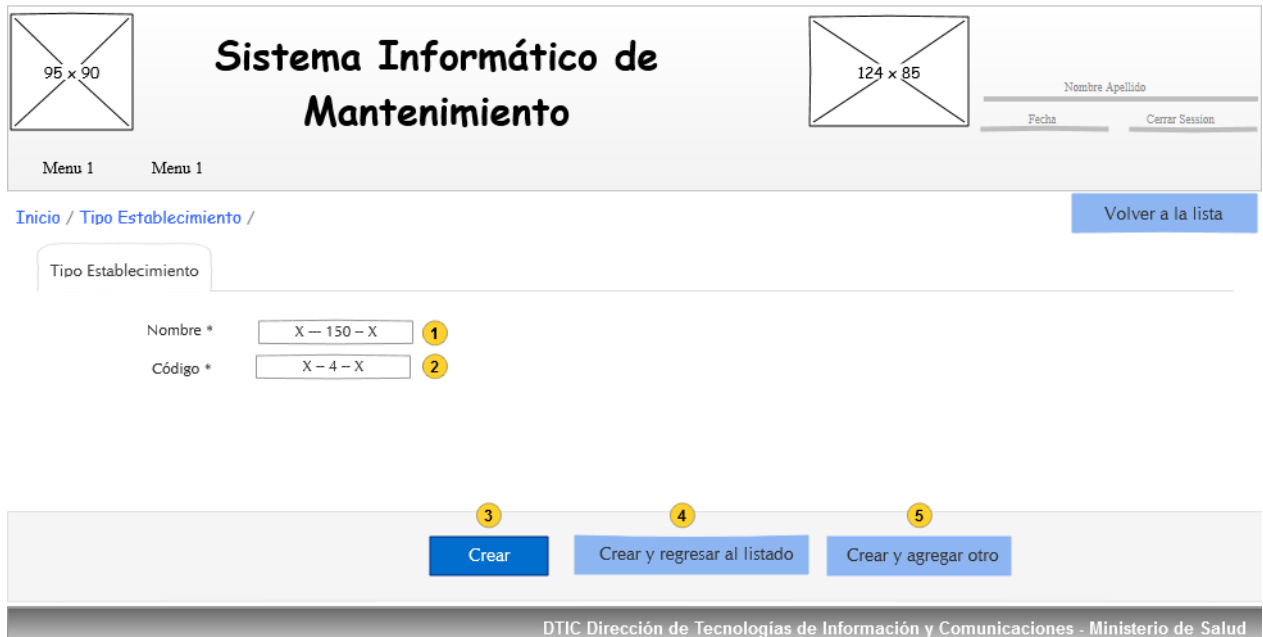


Figura 12-3. Interfaz Crear Tipo de Establecimiento

Interfaz de Modificación-Eliminación de Tipo de establecimiento

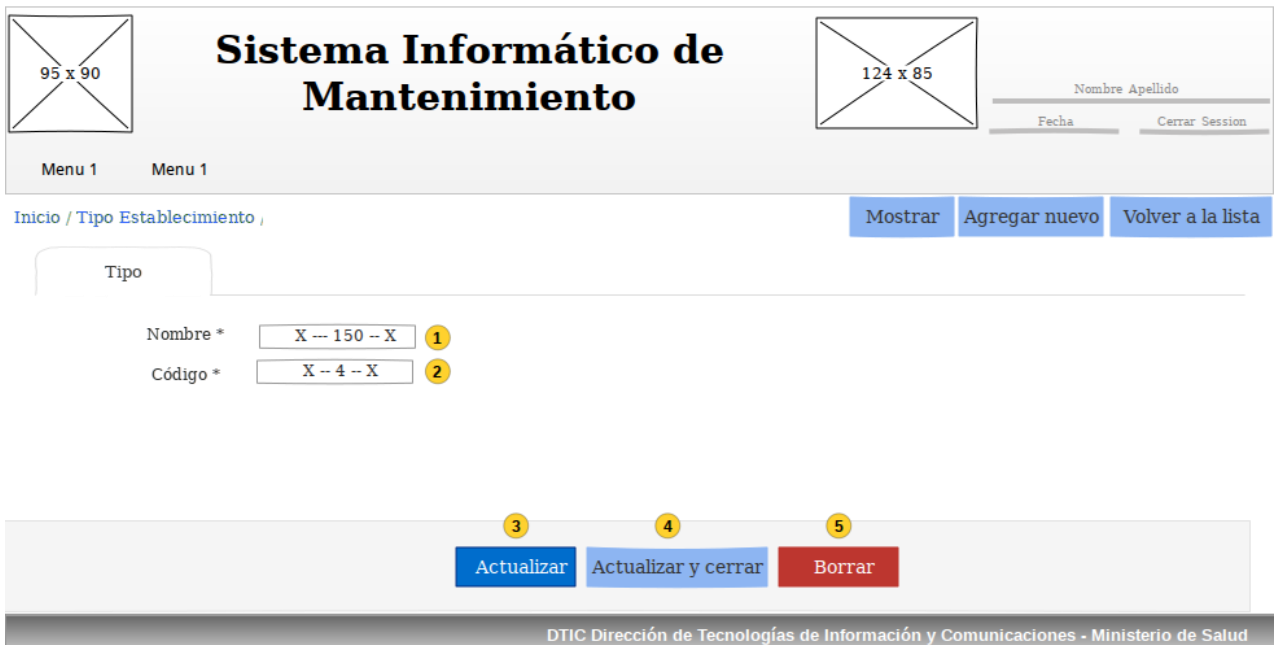


Figura 12-4. Interfaz Modificar - Eliminar Tipo de Establecimiento



12.2.3 Diagrama de Secuencia.

12.2.3.1 Diagrama de Secuencia de Gestión de Tipos de Establecimiento.

Diagrama de Secuencia - Crear Tipo de Establecimiento.

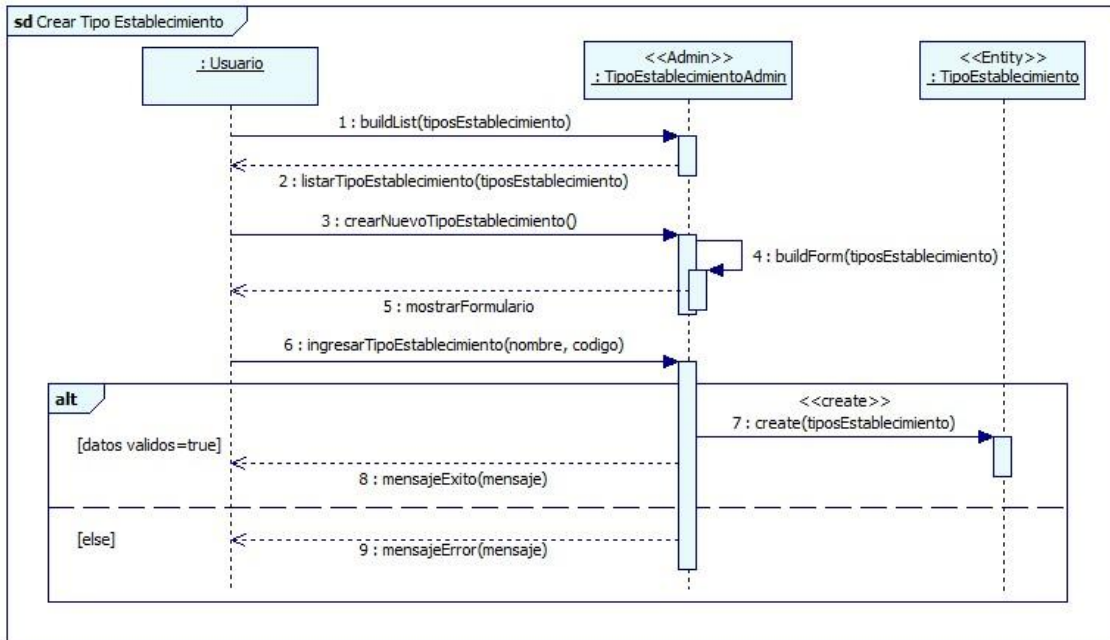


Figura 12-5. DS Crear Tipo de Establecimiento.



Diagrama de Secuencia – Consultar Tipo de Establecimiento.

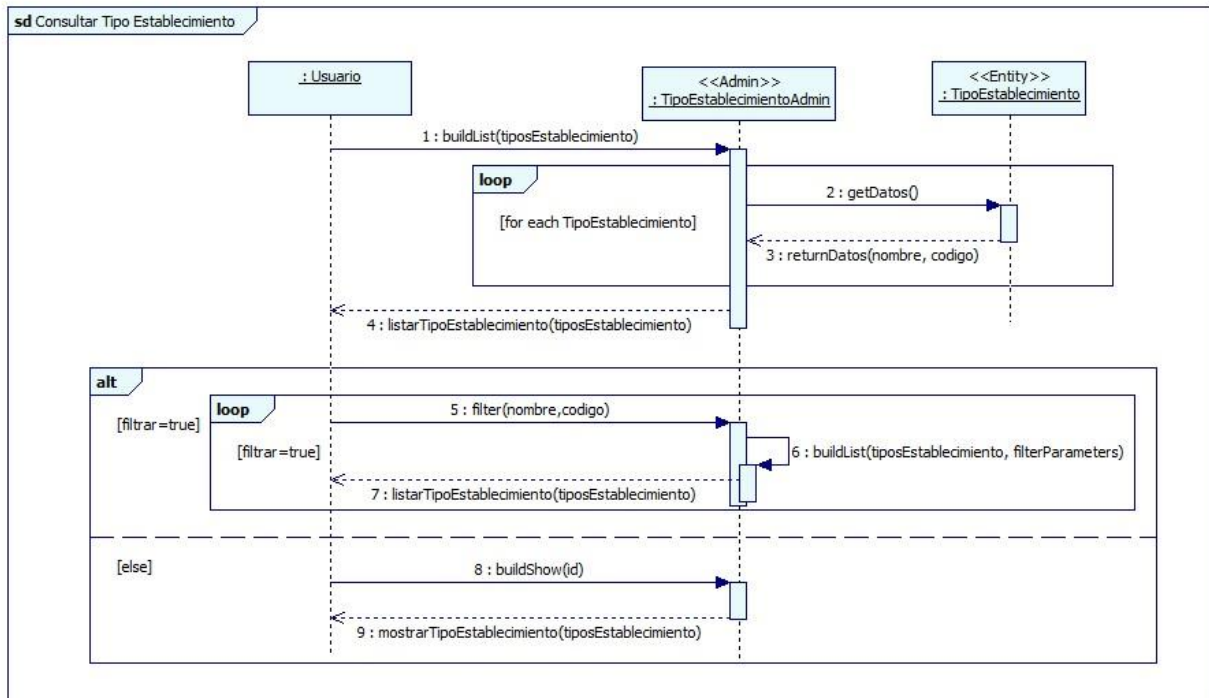


Figura 12-6. DS Consultar Tipo de Establecimiento.

Diagrama de Secuencia – Modificar Tipo de Establecimiento.

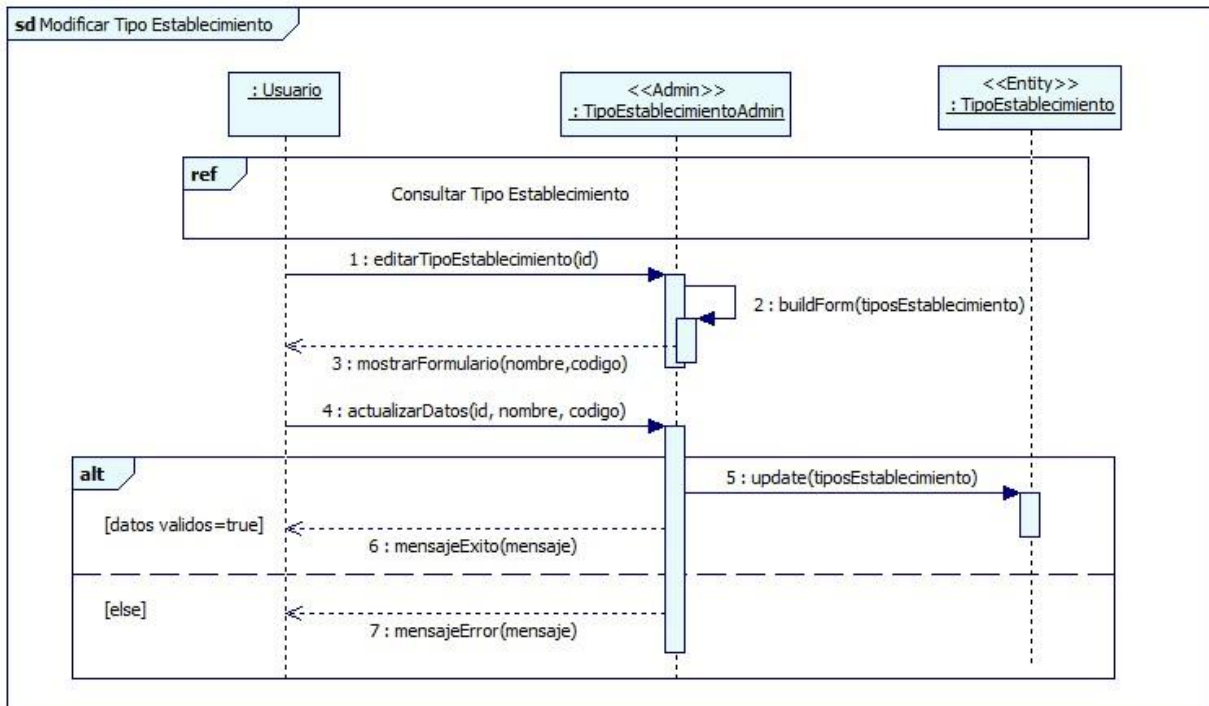


Figura 12-7. DS Modificar Tipo de Establecimiento.



Diagrama de Secuencia – Eliminar Tipo de Establecimiento.

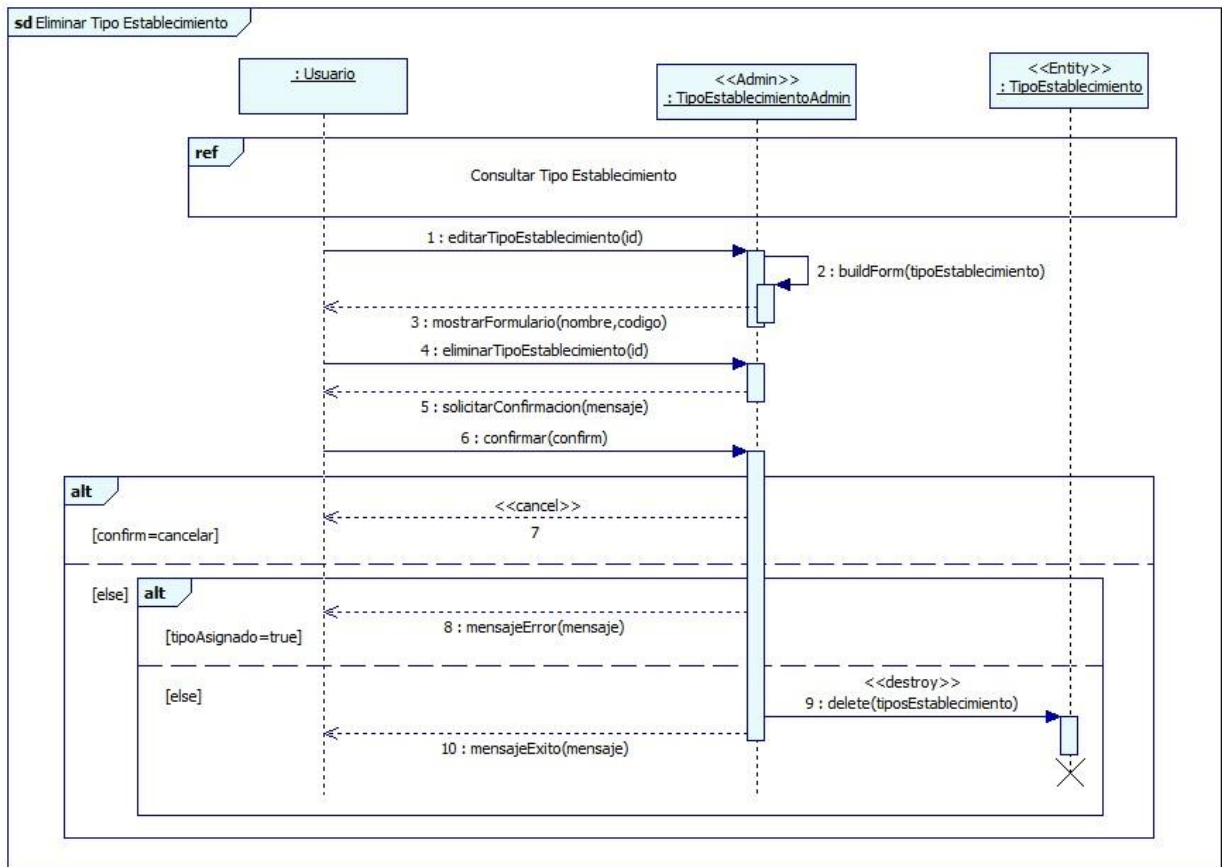


Figura 12-8. DS Eliminar Tipo de Establecimiento.



12.2.4 Diagramas de Actividad con la Notación BPMN.

12.2.4.1 Diagramas de Actividad de Gestión de Tipos de Establecimiento.

Diagrama de Actividad Crear Tipo de Establecimiento.

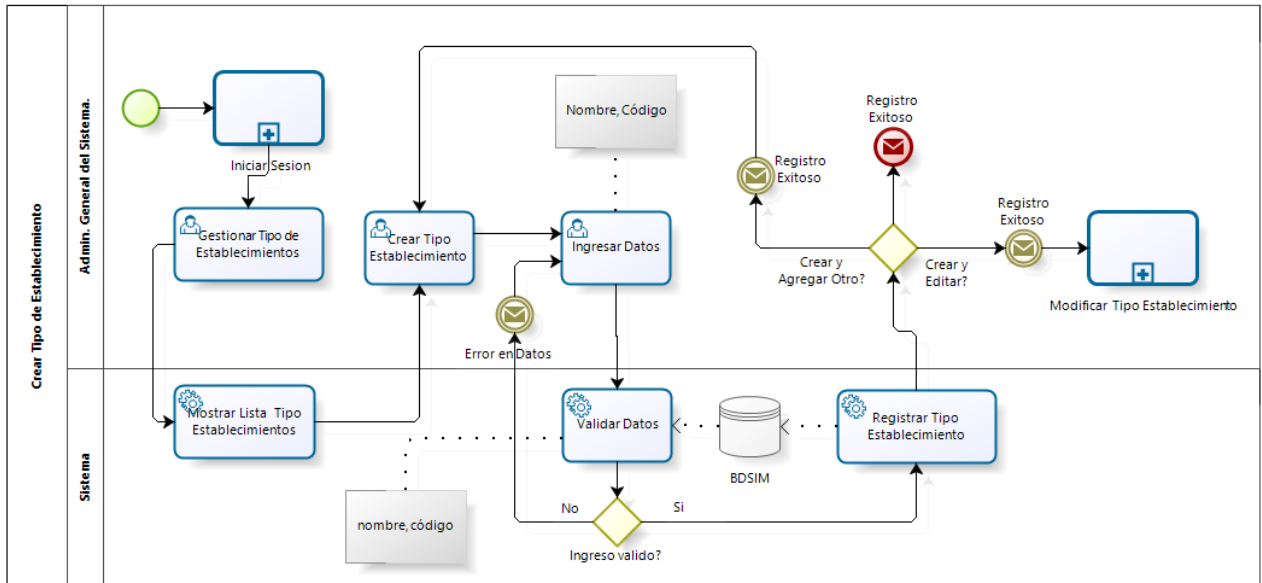


Figura 12-9. Diagrama de Actividad Crear Tipo de Establecimiento

Diagrama de Actividad Consultar Tipo de Establecimiento.

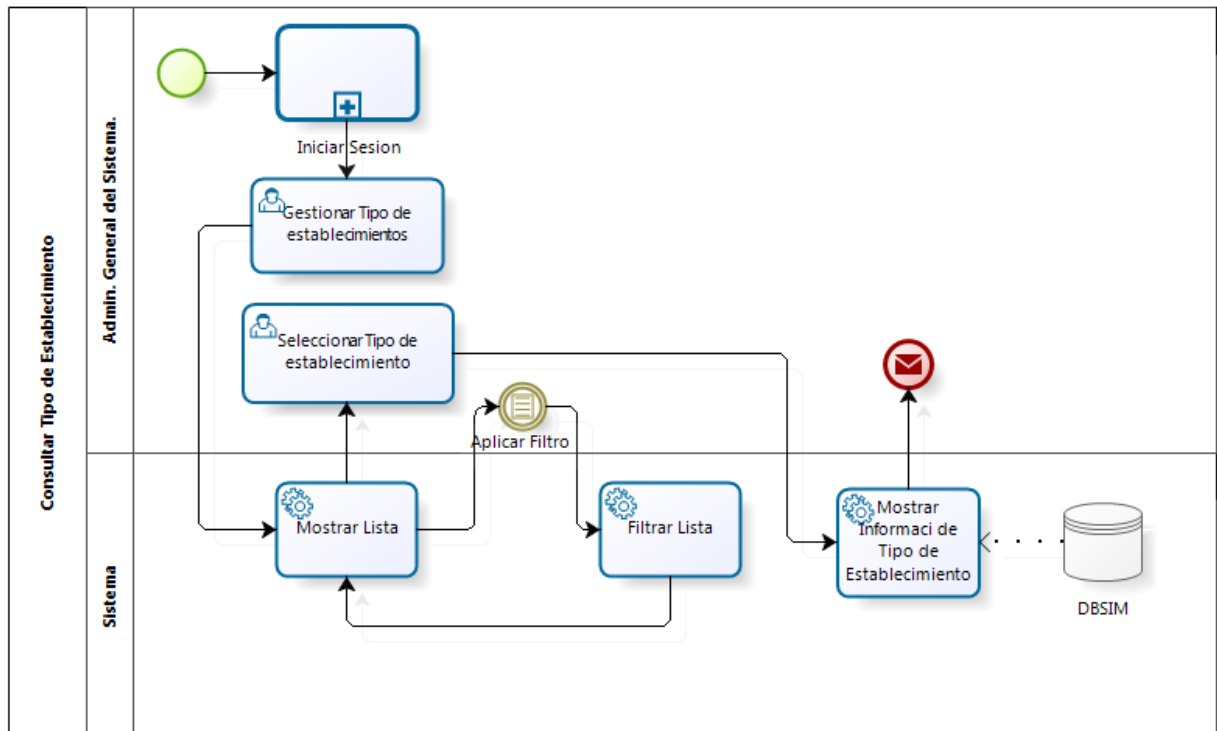


Figura 12-10. Diagrama de Actividad Consultar Tipo de Establecimiento



Diagrama de Actividad Modificar Tipo de Establecimiento.

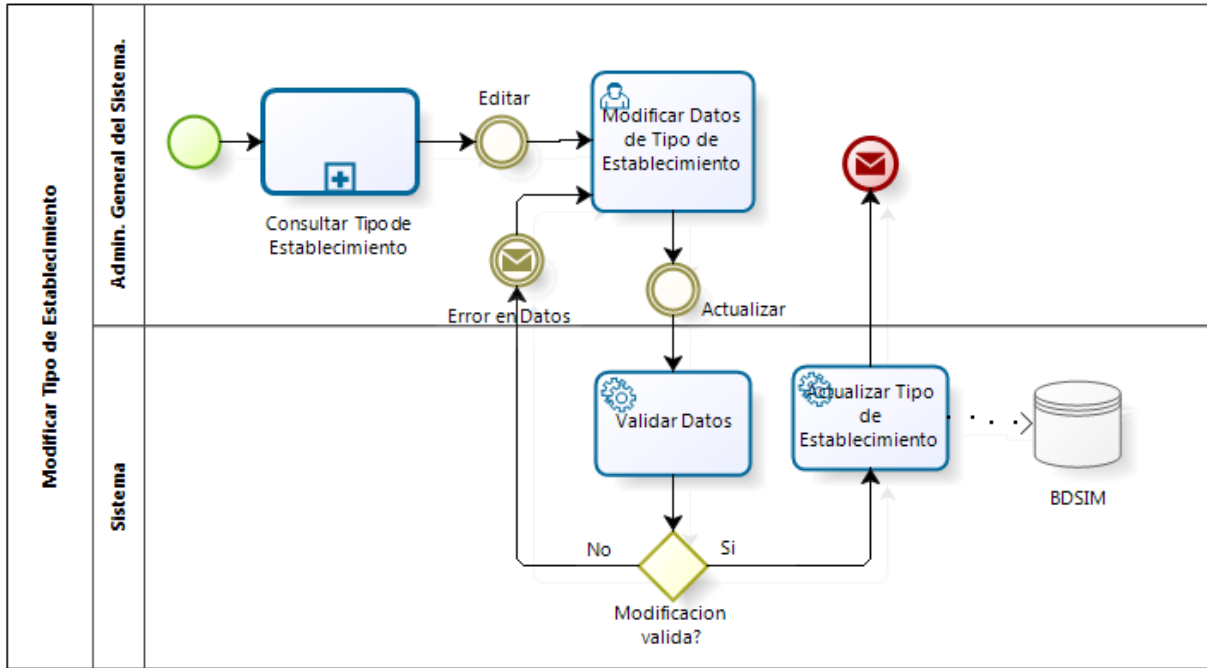


Figura 12-11. Diagrama de Actividad Modificar Tipo de Establecimiento.

Diagrama de Actividad Eliminar Tipo de Establecimiento.

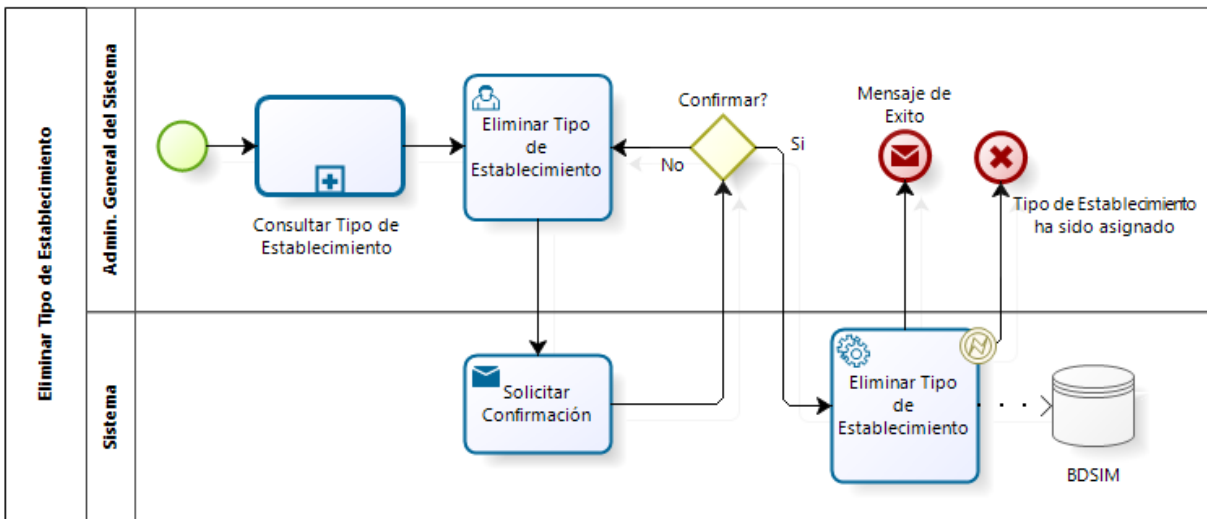


Figura 12-12. Diagrama de Actividad Eliminar Tipo de Establecimiento.



12.2.5 Diagramas de Clases del Primer Incremento (Establecimientos, Dependencias y Empleados).

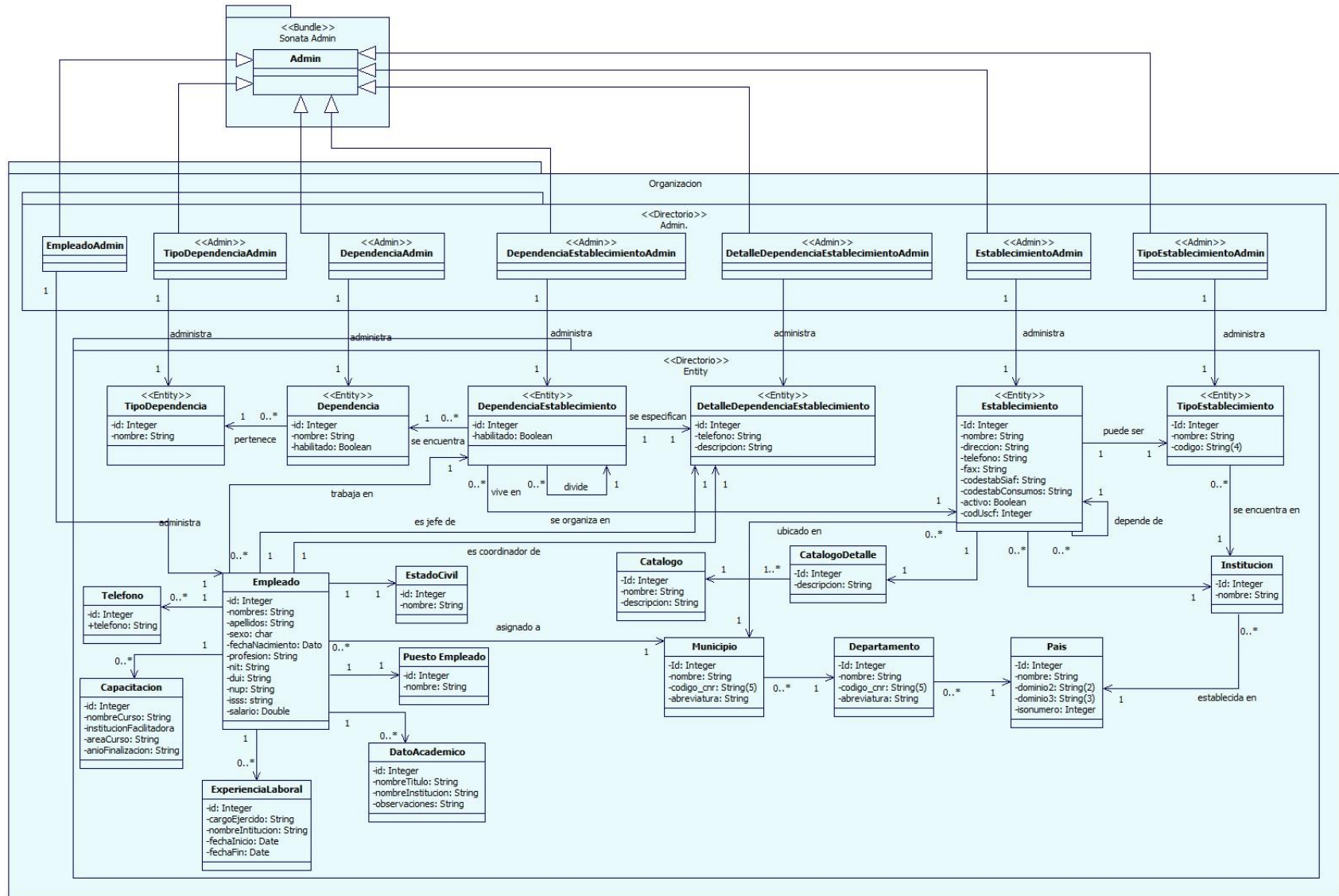


Figura 12-13 Diagrama de Clases de Establecimiento, Dependencia y Empleado.



12.2.6 Diseño de Casos de Prueba.

12.2.6.1 Diseño de Casos de Prueba de Integración de Subsistemas.

Suite de pruebas de Tipo de Establecimiento.

Esta Suite, agrupa los Casos de Prueba para Gestión de Tipos de Establecimiento, junto con sus correspondientes diseños para su posterior ejecución en la herramienta Selenium IDE.

Caso de Prueba:	Agregar un Tipo de Establecimiento Datos Válidos.
Código:	CP-01-01
Versión:	1.0
Suite	Registro de un nuevo Tipo de Establecimiento
Descripción General	
Propósito:	Agregar un nuevo tipo de establecimiento, ingresando datos válidos en los campos requeridos.
Requerimiento Funcional a Probar:	Agregar un tipo de establecimiento (CU-01-02)
Dependencias:	N/A
Ambiente de Pruebas	
Hardware	Ninguna configuración especial
Software	Sistema Operativo: Debian 7.0 Wheezy (stable version) Navegador Web: Iceweasel 22.0.1 Herramienta de Pruebas: Selenium IDE 2.2.0 Otros: Ambiente de pruebas de Symfony (app_dev.php)
Inicialización o Pre-Condiciones:	Iniciar Sesión.
Finalización o Post-Condiciones:	Tipo de Establecimiento registrado exitosamente.
Desarrollo de la Prueba	
Acciones:	<ol style="list-style-type: none"> 1. Ingresar a la página de crear un nuevo Tipo de Establecimiento. 2. Ingresar el nombre y el código. 3. Crear el Tipo de Establecimiento.
Datos de entrada:	Nombre: 'Hospital Departamental' Código: 'HD'
Resultados	
Resultados/ Datos Esperados:	Un mensaje de éxito de registro.
Severidad:	Normal.



Caso de Prueba:	Agregar un Tipo de Establecimiento Datos Inválidos.
Código:	CP-01-02
Versión:	1.0
Suite	Registro de un nuevo Tipo de Establecimiento
Descripción General	
Propósito:	Agregar un nuevo tipo de establecimiento, ingresando datos inválidos en los campos requeridos.
Requerimiento Funcional a Probar:	Agregar un tipo de establecimiento (CU-01-02)
Dependencias:	N/A
Ambiente de Pruebas	
Hardware	Ninguna configuración especial
Software	Sistema Operativo: Debian 7.0 Wheezy (stable version) Navegador Web: Iceweasel 22.0.1 Herramienta de Pruebas: Selenium IDE 2.2.0 Otros: Ambiente de pruebas de Symfony (app_dev.php)
Inicialización o Pre-Condiciones:	Iniciar Sesión.
Finalización o Post-Condiciones:	Ocurre un error en el registro de un nuevo Tipo de Establecimiento.
Desarrollo de la Prueba	
Acciones:	<ol style="list-style-type: none"> 1. Ingresar a la página de crear un nuevo Tipo de Establecimiento. 2. Ingresar el nombre y el código. 3. Crear el Tipo de Establecimiento.
Datos de entrada:	Nombre: '*/*..FFF' Código: '1/.'
Resultados	
Resultados/ Datos Esperados:	Se muestra un mensaje de error.
Severidad:	Normal.



12.2.7 Ejecución de Pruebas.

12.2.7.1 Ejecución de Casos de Prueba de Integración de Subsistemas.

Ejecución de Suite de Pruebas de Tipo de Establecimiento.

Caso de Prueba:	Agregar un Tipo de Establecimiento datos válidos.		
Código:	CP-01-01.		
Suite	Registro de un nuevo Tipo de Establecimiento		
Correlativo:	1		
Desarrollo de la Prueba			
Diseño Selenium IDE:	Command	Target	Value
	open	/sim/web/app_dev.php/admin/minsal/sim/ctltipoestablecimiento/create	
	sendKeys	//input	Hospital Departamental
	sendKeys	//div[2]/div/input	HD
	click	name=btn_create_and_edit	
	verifyTextPresent		Elemento creado satisfactoriamente.
Datos de entrada:	Nombre: 'Hospital Departamental' Código: 'HD'		
Resultados			
Resultados/Datos Esperados:	Mensaje que se efectuó satisfactoriamente el registro.		
Resultados/Datos Obtenidos:	Se crea satisfactoriamente el nuevo Tipo de Establecimiento y se muestra un mensaje de éxito.		
Resultado:	Éxito [X] Fallo []		
Código de Seguimiento:	N/A		
Evidencia:			



Cumplió Post-Condición:	Si
-------------------------	----

Caso de Prueba:	Agregar un Tipo de Establecimiento datos inválidos.
Código:	CP-01-02.
Suite	Registro de un nuevo Tipo de Establecimiento
Correlativo:	1

Desarrollo de la Prueba

Diseño Selenium IDE:	Command	Target	Value
	open	/sim/web/app_dev.php/admin/minsal/sim/ctltipoestablecimiento/create	
	sendKeys	//input	*/*..FFF"
	sendKeys	//div[2]/div/input	1/.
	click	name=btn_create_and_edit	
	verifyAlertNotPr...		

Datos de entrada:	Nombre: '/*..FFF' Código: '1/.'
-------------------	------------------------------------

Resultados

Resultados/Datos Esperados:	Mensaje de error ya que los datos de entrada no son válidos.
Resultados/Datos Obtenidos:	Mensaje de error
Resultado:	Éxito[X] Fallo[]
Código de Seguimiento:	N/A

Evidencia:

The screenshot shows the Selenium IDE interface. On the left, the test case 'AgregarTipoEstablecimiento *' is listed. The main area displays a table of commands and their results:

Command	Target	Value
open	/sim/web/app_dev.php/admin/minsal/sim/ctltipoestablecimiento/create	
sendKeys	//input	*/*..FFF"
sendKeys	//div[2]/div/input	1/.
click	name=btn_create_and_edit	
verifyTextPresent		Please match the requested format.

Below the table, the 'Log' tab is active, showing the following execution steps:

```
[info] Executing: |open| /sim/web/app_dev.php/admin/minsal/sim/ctltipoestablecimiento/create |
[info] Executing: |sendKeys| //input | /*..FFF" |
[info] Executing: |sendKeys| //div[2]/div/input | 1/. |
[info] Executing: |click| name=btn_create_and_edit |
[info] Executing: |verifyTextPresent| | Please match the requested format. |
```

Cumplió Post-Condición:	Si
-------------------------	----



12.2.8 Resumen de Resultados de Pruebas.

12.2.8.1 Suite de Pruebas de Tipo de Establecimiento.

Caso de Prueba		Resultado	Defectos / Bugs	
CP-01-01 – Agregar Tipo de Establecimiento Datos Válidos.		Éxito	0	
CP-01-02 – Agregar Tipo de Establecimiento Datos Inválidos.		Éxito	0	
CP-01-03 – Consultar Tipo de Establecimiento.		Éxito	0	
CP-01-04 – Modificar Tipo de Establecimiento Datos Válidos.		Éxito	0	
CP-01-05 – Modificar Tipo de Establecimiento Datos Inválidos.		Éxito	0	
CP-01-06 – Eliminar Tipo de Establecimiento Valido.		Éxito	0	
CP-01-07 – Eliminar Tipo de Establecimiento Invalido.		Éxito	0	
Total Resultados				
Ejecuciones	Éxitos	Fallos	Defectos / Bugs	Reportados
7	7	0	0	0



12.3 Segundo Incremento

El Segundo Incremento abarca el diseño y desarrollo de las siguientes funcionalidades y elementos:

Administración de Grupos y Roles: Da soporte a la gestión de los diversos privilegios dentro del Sistema por medio de Roles y la gestión de Usuarios del mismo.

Un Rol, en este contexto, es una acción específica que un Usuario puede realizar. Por ejemplo: Registrar un Nuevo Equipo o Realizar una Solicitud. Mientras que un Grupo, en este contexto, es un conjunto de Roles.

El objetivo de un Grupo, es definir una serie de Roles que constantemente serán asignados a diversos Usuarios, y que juntos conforman una agrupación que se administra más fácilmente dentro del Sistema. Por ejemplo, se podría crear un Grupo llamado “Secretaria”, como es de esperar, habrán muchos Usuarios que realizarán el papel de “Secretaria”, y el beneficio es que los Usuarios solamente se asocian con el Grupo “Secretaria”, en vez de otorgar (por cada uno) cada uno de los Roles que una “Secretaria” podría tener.

Administración de Usuarios: El objetivo principal es la gestión y registro nuevos Usuarios dentro del Sistema, los cuales pertenecerán a cada uno de los empleados, es decir, un Usuario estará única y exclusivamente asociado a un Empleado. Permitiendo de esta forma, conocer las acciones de cada Empleado, mediante su Usuario, en el Sistema.

El desarrollo de este Incremento también involucra considerar aspectos de seguridad, los cuales han sido especificados como parte del Diseño de Seguridad, descritos en las Secciones 11.4.1, 11.4.2 y 11.4.3 del presente documento.

12.4 Tercer Incremento

El Tercer Incremento abarca el diseño y desarrollo de las siguientes funcionalidades y elementos:

Inventario Técnico: Permitirá el registro y manejo de Bienes que forman parte los establecimientos del MINSAL, y a los cuales se brindan los servicios de mantenimiento. Es necesario recordar, que los Bienes se han categorizado en dos grandes grupos:

Los Bienes Inventariables, es decir, todo aquel Bien que puede ser registrado dentro del Inventario Técnico, como lo son: equipo médico, maquinaria, mobiliario, entre otros. Y los Bienes No Inventariables, que es todo aquel Bien que no puede ser registrado dentro de un Inventario Técnico, dentro de ellos se encuentran, por ejemplo, planta física, instalaciones, entre otros, pero al igual que los Bienes Inventariables, necesitan de mantenimiento y por lo tanto el Sistema debe dar soporte a estos Bienes que no tienen la capacidad de ser ingresados en un inventario de activo fijo. Sin embargo, este último se realizara en un incremento posterior.

El Inventario Técnico, lo comprenden todos aquellos Equipos, Maquinarias y Mobiliario que poseen un número de activo fijo o que tiene la capacidad de ser ingresado a un inventario de activo



fijo, pero, que por diversas razones no ha sido ingresado, sin embargo el Sistema debe soportar esta característica especial. Cada equipo, posee un Modelo, y cada Modelo está conformado por Características Técnicas, Atributos, Especificaciones Técnicas, y en el caso de las Áreas de Mantenimiento del MINSAL, deberán pertenecer a una determinada Clasificación y Grupo de Equipos.

El Sistema dará soporte a la creación de cualquier nuevo Modelo que se pueda presentar, mediante una estructura flexible de Base de Datos y que cumpla con los detalles ya mencionados.

Gestión de Empresas: Involucra la administración de cada una de las empresas que brindan algún tipo de servicio a los Establecimientos del MINSAL (conocido como Mantenimiento de Terceros o de Empresas Externas, que más adelante serán asociadas a un determinado Contrato para mantenimiento de equipos) o que se encuentran relacionadas con la fabricación, venta o suministro de repuesto de un determinado Bien que se registra en el Inventario Técnico.

12.5 Cuarto Incremento

El Cuarto Incremento abarca el diseño y desarrollo de las siguientes funcionalidades y elementos:

Gestión de Condiciones de Contrato: Registro y manejo de todas aquellas condiciones que son establecidas en un Contrato de Mantenimiento por Terceros (Empresas Externas).

Gestión de Medidas de Contrato: Gestión de las Medidas que se han de aplicar a los diferentes Equipos que se encuentran cubiertos por un determinado Contrato con Terceros.

Administración de Contratos: Es la principal funcionalidad del Incremento. La cual dará soporte al registro de nuevos Contratos por Terceros. Esto implica que una empresa externa, se hará responsable de dar el servicio de mantenimiento a diversos equipos, siendo el personal de las áreas de mantenimiento, responsable únicamente de darle seguimiento y supervisión a estos. Los Contratos serán asociados con Condiciones que se deben cumplir y Medidas que se deben aplicar, así como el equipo que será incluido dentro del contrato. Un contrato puede ser definido para un establecimiento en específico o para un conjunto de estos.

12.6 Quinto Incremento

El Quinto Incremento abarca el diseño y desarrollo de las siguientes funcionalidades y elementos:

Bien No Inventariable: Como se ha mencionado, un Bien No Inventariable, es todo aquel Bien que no puede ser registrado dentro de un Inventario Técnico, dentro de ellos se encuentran, por ejemplo, planta física, instalaciones, entre otros, pero al igual que los Bienes Inventariables (cuya funcionalidad fue desarrollada en el Tercer Incremento), necesitan de mantenimiento y por lo tanto el Sistema debe dar soporte a estos Bienes que no tienen la capacidad de ser ingresados en un inventario de activo fijo.



Solicitudes de Servicio de Mantenimiento: Da soporte a uno de los procesos vitales de las áreas de mantenimiento. Cuando un equipo, instalación o infraestructura presenta problemas en una determinada Unidad (Servicio, Área o Departamento), el Jefe de dicha Unidad es el encargado de realizar y enviar una Solicitud de Servicio de Mantenimiento al Departamento de Mantenimiento correspondiente. Cuando el Jefe de Mantenimiento recibe la Solicitud, la revisa y determina si procede o no el trabajo solicitado. Si no procede, cancela la Solicitud y se le informa al solicitante. Pero, si una Solicitud no procede debido a la capacidad de resolución del Departamento de Mantenimiento, esta es transferida al Nivel Superior.

Al mismo tiempo, involucra la Consulta del Estado de la Solicitud por parte del solicitante, mostrando la Fase y Estado de la misma, y si en caso de requerirse, su cancelación.

Ficha de Vida de Equipos: Es el historial que presenta un equipo a lo largo del tiempo, en la cual se muestran los mantenimientos que han sido brindados a dicho equipo, mostrando detalles de los trabajos realizados y los costos en los que se incurre.

12.7 Sexto Incremento

El Sexto Incremento abarca el diseño y desarrollo de las siguientes funcionalidades y elementos:

Órdenes de Trabajo: Al igual que en el Incremento anterior, se desarrolla uno de los puntos principales y medulares en los procesos de servicio de mantenimiento, que abarca el desarrollo de las Órdenes de Trabajo. Las Órdenes de Trabajo, son el medio por el cual se ejecutan los diferentes tipos de mantenimiento, las cuales son asignadas a un determinado Técnico de Mantenimiento, indicándole cual es el trabajo que debe realizar. Las Órdenes de Trabajo que debe soportar el Sistema, son cuatro tipos:

- **Órdenes de Trabajo de Mantenimiento Correcto de Equipos:** Orden de Trabajo que se crea a partir de una Solicitud de Mantenimiento para un Equipo (que pertenece a un Inventario Técnico).
- **Órdenes de Trabajo de Mantenimiento Correctivo de Instalaciones:** Orden de Trabajo que se crea a partir de una Solicitud de Mantenimiento para Instalaciones (que engloba Planta Física, Infraestructura y similares).
- **Órdenes de Trabajo de Supervisiones de Contratos:** Son generadas a partir de un Contrato con Terceros. Al momento en que la empresa realice una visita para dar mantenimiento a un equipo, se crea una Orden de Trabajo de Supervisión del trabajo a realizar por la empresa, la cual es asignada a un Técnico de Mantenimiento.
- **Órdenes de Trabajo de Mantenimiento Preventivo de Equipos:** Se generan a partir de Planes de Mantenimiento Preventivo Planificado, en la que el Técnico Responsable desarrolla el mantenimiento preventivo de un equipo, dando así cumplimiento con los Planes y las Rutinas que debe aplicar.

Informes de Orden de Trabajo: Los Técnicos encargados de ejecutar una Orden de Trabajo, son responsables de ingresar los Informes de Trabajo. Deben ingresar información que servirá para el



cálculo del costo de los diferentes tipos de mantenimiento. Por tanto, el Informe de la Orden de Trabajo comprende:

- Reporte de Mano de Obra: Las horas invertidas por el/los Técnicos de Mantenimiento, que permitirá calcular costos en base al salario.
- Reporte de Otros Costos: Otros costos en los que se ha incurrido, como viáticos, transporte, etc.
- Reporte de Materiales y/o Repuestos Utilizados: Se deberá especificar los Materiales y Repuestos que han sido utilizados para reparar o dar atención a un equipo. Esto se hace por medio del código de los Vales de Salida, registrados en el SINAB (Sistema Nacional de Abastecimiento), por lo cual, el SIM se conectara con este para extraer los datos necesarios.
- Reporte de Fallas Detectadas: Registro de las fallas encontradas al revisar o inspeccionar el equipo.

12.8 Séptimo Incremento

El Séptimo Incremento abarca el diseño y desarrollo de las siguientes funcionalidades y elementos:

Rutinas de Equipo: Las Rutinas especifican una serie de pasos o actividades a seguir, que se aplican a una determinada Clase de Equipos, para completar el mantenimiento preventivo de los mismos. Una de las características especiales de las Rutinas, es que se realizan cada cierto periodo de tiempo, por lo cual es necesario definir su Periodicidad (mensual, trimestral, etc.).

Planes de Mantenimiento Preventivo: El Sistema debe soportar la creación y gestión de los Planes de Mantenimiento Preventivo. Los Planes de Mantenimiento Preventivo se crean para un determinado Equipo, para un determinado año, asignando un Técnico Responsable de llevar a cabo el Plan y una Prioridad. Estos Planes se asocian con una determinada Rutina que será aplicada al Equipo, la cual define cada cuanto tiempo será brindado el mantenimiento preventivo (Periodicidad). Esto permite conocer en qué momento (Semana del Año) será necesario dar mantenimiento al equipo, y en ese momento entonces, será generada una Orden de Trabajo de Mantenimiento Preventivo.

Una de las funcionalidades importantes de este Incremento es la visualización del Programa de Mantenimiento Preventivo Planificado, conocido por sus siglas como PMPP. En el cual se muestran los diferentes Planes de Mantenimiento Preventivo registrados, visualizados a los largo del año, comparando la ejecución planificada versus la real (Semana planificada versus la semana que se ejecutó realmente la Orden de Trabajo correspondiente).

Este Incremento finaliza el Diseño Detallado y desarrollo del Código Fuente del Software.



13 Capítulo VII

Plan de Implementación

Todo el esfuerzo del proyecto se ve reflejado en el producto final, un Sistema Informático funcional y que satisface los requerimientos del usuario.

Sin embargo, hay actividades necesarias que deben realizarse para llevar a un estado de operación el Sistema desarrollado. A este conjunto de actividades se le conoce como Plan de Implementación.

El Plan de Implementación tiene por objetivo definir todas aquellas actividades a realizar para poner el Sistema Informático de Mantenimiento en funcionamiento, o también conocida como puesta en marcha del Sistema dentro de la organización, y finalmente en cada uno de los establecimientos del MINSAL, presentando los recursos tecnológicos y humanos necesarios para realizar dichas actividades de implementación.



13.1 Alcance de la Implementación.

El Sistema Informático para la Planeación y Gestión de los Procesos de Servicio de Mantenimiento del Área de Mantenimiento General, Nivel Regional y Local del Ministerio de Salud (en adelante, SIM), es un complejo y visionario sistema informático que llevará beneficios a todos o la gran mayoría de Establecimientos de Salud, ya que todos son sujetos a los servicios de mantenimiento de equipo médico-hospitalario, infraestructura, instalaciones, inmuebles y otros similares.

El Ministerio de Salud es una organización sumamente extensa, que abarca el territorio nacional por medio de la Red de Establecimientos de Salud (Regiones, Hospitales, Unidades de Salud, entre otros.), por cual, la implementación del SIM para todos los establecimientos será llevada a cabo de forma gradual.

Por tal razón, el alcance del Plan de Implementación del SIM propuesto en el presente documento, se apega al Método de Implementación Piloto, que por su definición, implica instalar el Sistema en un área de la organización y luego de aprobado y verificar su funcionamiento, extenderlo al resto de la misma. Por lo que se consideraran determinados Establecimientos de Salud, para iniciar la implementación en los mismos, y luego extenderla a los demás Establecimientos del país, completando así la integración completa del SIM al Ministerio de Salud.

El Plan de Implementación está enfocado a las actividades a realizar para la implementación del Sistema en cada uno de estos establecimientos, presentando los recursos tecnológicos y humanos necesarios para realizar estas actividades de implementación, entre otras cosas que se necesitan o se requieren como la Arquitectura del Sistema, Conocimientos, Redes y otros.



13.2 Requerimientos de la Implementación.

13.2.1 Recursos Tecnológicos

13.2.1.1 Requerimientos de Hardware.

Los requerimientos de hardware especifican las características técnicas necesarias del equipo cliente y servidor que permitirán la implementación del Sistema Informático.

Especificaciones Técnicas de Servidor

A continuación en la Tabla 13-1 se presentan las características y especificaciones técnicas del Hardware que dará soporte a la instalación y configuración del Sistema Informático SIM, desde el Servidor Web donde estará alojado el Sistema, en conjunto con el SGBD (Sistema Gestor de Base de Datos) que maneja la Base de datos del mismo, además de todo el Software descrito en el Manual de Instalación:

Característica	Especificación Técnica
Modelo	Dell PowerEdge R815
Factor de Forma	2U rack
Procesador	AMD Opteron 6100 (48 Núcleos)
Sockets de Procesadores	4
Front Side Bus / HyperTransport	HyperTransport-3 Links
Cache	L2: 512K / core L3: 12MB
Chipset	AMD
Memoria RAM	64 GB (32 DIMM Slots) 1333MHz
I/O Slots	6 PCIe G2 slots + storage slot: 5 x8 slots 1 x4 slot 1 x4 storage slot
Disco Duro	3 TB
Conexiones de Red	Broadcom BMC57710 10Base-T Copper Single Port NIC, PCI-E x8.
Disponibilidad	Hot-plug hard drives, hot-plug redundant power, dual SD modules, ECC memory, interactive LCD screen
Fuente de Alimentación	<ul style="list-style-type: none"> ▪ Una fuente de alimentación de 1100W no redundante conectable en caliente ▪ Dos fuentes de alimentación 1100W redundantes conectables en caliente
Video	Matrox G200eW w/ 8MB memory
Soporte de Rack	ReadyRail sliding rails 4-post racks
Sistema Operativo	Debian 7.0 Wheezy

Tabla 13-1 Especificaciones Técnicas de Servidor



Requerimientos Mínimos de Máquina Cliente

Para que los usuarios puedan interactuar con el SIM y utilizar las funcionalidades que este provee, es necesario que cuenten con el equipo adecuado en sus estaciones de trabajo, el cual se conecta al Servidor, mediante la red proveída por el MINSAL. A estos equipos se les conoce como maquinas cliente. A continuación, en la Tabla 13-2 se presentan los requerimientos mínimos necesarios que una máquina cliente debe poseer, con las cuáles el usuario podrá utilizar el Sistema adecuadamente:

Característica	Especificación Técnica
Procesador	Intel® Pentium 4, 1.0GHz
Memoria RAM Libre Necesaria	128 MB
Espacio Disco Duro Libre Necesario	1 GB
Conexiones de Red	Tarjeta de Red Ethernet a 10 Gbit/s Tarjeta Inalámbrica 802.11b
Tipo de Equipo	Desktop

Tabla 13-2 Requerimientos Mínimos de Máquina Cliente

Requerimientos de Configuración de Red

Es necesario de un correcto despliegue físico de la red y una correcta configuración lógica de la misma, que permita la interacción entre diferentes dispositivos o sistemas. El MINSAL dispone de una completa infraestructura física de red, descrita a mayor detalle en la Sección 11.3.6 de este documento, por lo que este apartado está enfocado a definir los requerimientos de comunicación en relación a protocolos utilizados, puertos habilitados, etc. Los cuales se muestran en la Tabla 13-3:

Aplicación, Software o Servicio	Configuración de Red Requerida	Descripción
Comunicación de Red	Protocolo: TCP	Se requiere que la comunicación del Sistema se realice por medio del protocolo TCP.
Apache 2 (Servicio Web).	Acceso: Red Interna Protocolo: HTTP Puerto: 80	Se utilizará el puerto 80 para el acceso web al Sistema, por medio del protocolo HTTP.
PostgreSQL 9.1	Acceso: Local Puerto: 5432	Para la comunicación con la base de datos PostgreSQL se utilizará el puerto 5432.
Apache Tomcat 7.0.42	Acceso: Local Puertos: 8080, 8005	El Sistema accederá a los servicios de Apache Tomcat (para la generación de reportes) utilizando los puertos 8080 y 8005.
		El sistema será accesible solamente en la red interna del MINSAL (INTRANET), es decir, que no se podrá acceder a él por medio de Internet.



Aplicación, Software o Servicio	Configuración de Red Requerida	Descripción
Git (Control de Versiones).	Acceso: Red Externa Protocolo: SSH Puerto: 22	Para el control de versiones al dar mantenimiento al Sistema (desarrollo de nuevas funcionalidades) se utilizará Git con protocolo SSH, que utiliza el puerto 22, esto se realizará por medio de una llave pública que se deberá registrar en el servidor de Git.
Envío de Correos Electrónicos	Acceso: Red Interna Protocolo: SMTP Puerto: 25	Para la comunicación por medio de correo electrónico institucional se utilizará el protocolo SMTP (Simple Mail Transfer Protocol), puerto TCP 25.

Tabla 13-3. Requerimientos de Configuración de Red.

13.2.1.2 Requerimientos de Software.

En la Tabla 13-4 se listan los diferentes Softwares requeridos para el funcionamiento del sistema SIM y posteriormente se describe cada uno de ellos:

Software	Versión
Apache	2.2
PHP	5.4
Composer	1.0 alpha 8
Symfony 2	2.4.1
Sonata	-
Doctrine	2.2.4
Twig	1.15
JasperReport Server	5.0.1
Apache Tomcat	6.0.26
PostgreSQL	9.1

Tabla 13-4 Lista de elementos de Software



Apache.

Es un Servidor HTTP Open-Source, el objetivo de este servidor es proporcionar los servicios HTTP de acuerdo con los actuales estándares de HTTP⁷⁶.

PHP

Es un Lenguaje de propósito general que es especialmente adecuado para el desarrollo web⁷⁷.

Composer

Composer es una herramienta para la administración de dependencias en PHP, la cual permite declarar las dependencias necesarias para un proyecto y de las cuales Composer se encargará de instalar⁷⁸, esta herramienta permitirá actualizar las dependencias del proyecto al momento de instalarlo.

Symfony 2

Es un Framework de desarrollo web que utiliza como lenguaje de programación PHP⁷⁹, que servirá como el núcleo sobre el cual se ha de desarrollar el sistema SIM. Este Framework funciona con otros Framework basados en el mismo Symfony, los cuales son integrados con él para brindar mayor funcionalidad, a estos Frameworks se les conoce con el nombre de **Bundle**, el cual se puede entender como un conjunto estructurado de archivos dentro de un directorio, o un “paquete” que presenta una característica única o con una funcionalidad específica.

A continuación se listan los diferentes Bundles que se integrarán con Symfony, algunos de estos Bundles ya vienen por defecto con Symfony mientras que otros serán necesario instalarlos por separado, los cuales se describirán en el apartado de instalación y configuración que se describirá más adelante.

- **FOSUserBundle:** Framework que tiene por objetivo facilitar las tareas de administración de usuarios tales como: registro de usuarios, recuperación y cambio de contraseñas entre otros.
- **FOSJsRoutingBundle:** Permite la generación de rutas que pueden ser accedidas a través de JavaScript.
- **JmsSecurityExtraBundle:** Permite mejorar la seguridad de los componentes de Symfony2.
- **JmsDiExtraBundle:** Mejora el uso de inyecciones de dependencia de Symfony 2, brindando funcionalidades tales como: inyección de dependencia vía anotaciones entre otros.
- **GenemuFormBundle:** Bundle que permite la integración de elementos de JavaScript dentro de

⁷⁶ Apache Software Foundation. (7 de Enero de 2014). Obtenido de Apache HTTP Server Project: <http://httpd.apache.org/>

⁷⁷ PHP Group. (7 de Enero de 2014). Obtenido de PHP: <http://php.net/>

⁷⁸ Adermann, N., & Boggiano, J. (10 de Enero de 2014). *Getting Started*. Obtenido de Composer Dependency Manager for PHP: <http://getcomposer.org/doc/00-intro.md>

⁷⁹ SensioLabsNetwork. (10 de Enero de 2014). *What is Symfony?* Obtenido de Symfony: <http://symfony.com/what-is-symfony>



Sonata

Es un proyecto que tiene por objetivo proporcionar un Framework que brinda ciertas características que facilitan la administración de los elementos del sistema, basados en Symfony 2, integrándose con éste y permitiendo así facilitar la construcción de sitios de manera más fácil⁸⁰. Sonata se compone de un conjunto de Bundles los cuales se describen a continuación:

- **Exporter:** Permite la exportación de los datos en diferentes formatos tales como json, xml, csv, xls.
- **KnpmenuBundle:** Bundle que permite la integración de la de un menú de PHP con Symfony 2⁸¹, brindando con esto una versatilidad al momento de manejar y crear menús personalizados.
- **SonataCacheBundle:** Facilita el almacenamiento de los datos procesados en una chace que trabaja en segundo plano⁸².
- **SonataBlockBundle:** Permite la definición y personalización de nuevos Bloques dentro del sistema.
- **SonataAdminBundle:** Facilita la administración de las operaciones de creación, eliminación, actualización y consulta de los diferentes elementos del sistema.
- **SonataDoctrineORMAdminBundle:** Permite la integración entre SonataAdminBundle y Doctrine para la gestión de los objetos de la base de datos.
- **SonataIntlBundle:** Facilita un conjunto de librerías que permite el formato de texto y fechas dependiendo de la localización geográfica defina en la configuración de Apache.
- **SonataEasyExtendsBundle:** Genera la estructura válida a partir de un Bundle, es decir, permite extender de un Bundle de manera sencilla para su posterior personalización.
- **SonataUserBundle:** Facilita la integración entre FOSUserBundle y Sonata Admin Project, aumentando la funcionalidad de FOSUserBundle.
- **LswDoctrinePDODlib:** Bundle que contiene el driver DBlib que permite la conexión entre SQL Server y Linux.

Doctrine

Conjunto de librerías PHP enfocadas en proporcionar un servicio de persistencia y las funcionalidades relacionadas a esto, es decir, permiten realizar el Mapeo Objeto Relacional de la base de datos en el sistema y operaciones sobre la misma.

Twig

Rápido, seguro y flexible motor de plantillas para PHP, permite compilar plantillas bajo un código optimizado, posee un modo sandbox (modo aislado o de comprobación de código, su funcionalidad es similar a un “try” y “catch”), el cuál evalúa plantillas con código malicioso, etc.

⁸⁰ Sonata Project. (10 de Enero de 2014). *What is Sonata*. Obtenido de Sonata Project Symfony 2 Bundles on Steroids: <http://sonata-project.org/about>

⁸¹ Knplabs. (10 de Enero de 2014). Obtenido de KnpmenuBundle: <https://github.com/KnpLabs/KnpMenuBundle>

⁸² SonataProject. (Enero 10 de 2014). *Introduction*. Obtenido de Sonata CacheBundle Documentation: <http://sonata-project.org/bundles/cache/master/doc/reference/introduction.html>



JasperReport Server

Servidor de reportes que proporciona información crítica en tiempo real o tareas basadas en búsqueda, impresión, o envío de información a través de correos presentado en diferentes formatos.

PostgreSQL

Base de Datos Objeto-Relacional Open Source, en el que se almacenarán los datos del sistema a través de tablas y relaciones en conjunto con los demás objetos de la base de datos del SIM.

Apache Tomcat

Servidor Web Open Source para la implementación de Java Servlet y JavaServerPages tecnologías desarrolladas bajo Java, funcionará en conjunto con JasperReport para la presentación de los reportes del sistema.

Sistema Operativo

El sistema SIM funcionará bajo el Sistema Operativo Linux, en su distribución Debian Wheezy 7.0, para la instalación del sistema SIM es necesario que el usuario tenga un conocimiento intermedio en el manejo del sistema operativo Debian en las siguientes áreas:

- Instalación de paquetes desde repositorios.
- Manejo de comandos básicos y avanzados en consola.
- Iniciar y detener servicios.
- Manejo de editores vi, vim o nano. (editores que funcionan por consola).

Para más información ver la documentación oficial de Debian en el siguiente enlace: <https://wiki.debian.org/>.

Navegador Web

Debido a la no estandarización en lo referente al motor de renderizado de los diferentes navegadores web, el sistema SIM ha sido optimizado para los siguientes navegadores, por lo cual se recomienda su uso:

- Mozilla Firefox 26.0 o superior
- Iceweasel 26.0 o superior (Navegador de Firefox en Debian).
- Google Chrome 31.0.1650.63-1 o superior



13.2.2 Recurso Humano

Para la correcta implementación del Sistema Informático, es necesario contar con el personal adecuado, es decir, personal que posea el conocimiento y las habilidades necesarias para desarrollar en conjunto esta ardua tarea.

La Implementación del Sistema, requiere de la integración de diversos conocimientos y habilidades, no solo en el área de informática. Esto implica que el personal debe tener ciertos perfiles para ser asignados como responsables de una determinada actividad. Estos perfiles, se describen a continuación:

13.2.2.1 Perfiles del Recurso Humano

Perfil - Director de Proyecto.

Perfil:	Director de Proyecto.	Código:	DIR_PROY
Nivel Académico			
Ingeniero de Sistemas Informáticos, Computación o Áreas afines.			
Conocimientos			
Conocimientos Técnico			
<ul style="list-style-type: none"> • Conocimiento avanzado sobre Análisis, Diseño y Desarrollo Sistemas Informáticos. • Conocimiento Avanzado de diversas herramientas tecnológicas del área de Desarrollo de Sistemas, Bases de Datos, Arquitectura de Software, Planeación y Gestión. • Conocimiento sobre Gestión de Proyectos, Administración de Recursos, Presupuestos y Recursos Humanos. • De preferencia con diplomados o cursos sobre Gestión de Proyectos, PMI o similares. 			
Habilidades Personales			
<ul style="list-style-type: none"> • Capacidad de Liderazgo. • Capacidad analítica y de solución de nuevos problemas. • Organizado. • Orientado a metas y objetivos. • Habilidad para toma de decisiones y asignación de recursos. • Compromiso con los objetivos institucionales. • Buenas relaciones interpersonales y habilidad para seleccionar, asignar e inspirar a los demás. 			
Experiencia			
Tres años de experiencia en actividades de Administración de Proyectos Informáticos.			
Cargo Sugerido			
Jefe de Departamento dentro de la DTIC – MINSAL.			

Tabla 13-5 Perfil de Director de Proyecto.



Perfil – Administrador de Servidores.

Perfil:	Administrador de Servidores.	Código:	ADM_SERV
Nivel Académico			
Ingeniero de Sistemas Informáticos, Computación o Áreas afines.			
Conocimientos			
Conocimientos Técnicos			
<ul style="list-style-type: none"> • Manejo avanzado del Sistema Operativo Linux, Distribución Debian 7.0 Wheezy. • Conocimientos avanzados sobre Virtualización e Hipervizores (De preferencia XEN). • Conocimiento avanzado sobre Instalación, Operación y Mantenimiento de Servidores Web, Servidores de Bases de Datos, Servidor de Correo. • Conocimiento Intermedio de Redes. • Conocimientos de los Gestores de Base de Datos PostgreSQL y SQL Server 2008 R2. 			
Habilidades Personales			
<ul style="list-style-type: none"> • Trabajo en equipo. Capacidad analítica y de solución de nuevos problemas. • Disciplina en sus operaciones y responsabilidades diarias. • En constante aprendizaje de nuevas tecnologías y tendencias del mercado. 			
Experiencia			
Dos años en actividades de Administración y Configuración de Servidores.			
Cargo Sugerido			
Administrador de Servidores de la DTIC – MINSAL.			

Tabla 13-6 Perfil de Administrador de Servidores.

Perfil – Desarrollador de Aplicaciones Web.

Perfil:	Desarrollador de Aplicaciones Web.	Código:	DES_APC
Nivel Académico			
Ingeniero de Sistemas Informáticos, Técnico en Programación o Áreas Afines.			
Conocimientos			
Conocimientos Técnicos			
<ul style="list-style-type: none"> • Conocimientos en Administración de Sistemas Informáticos. • Dominio de Programación Orientada a Objetos y Lenguajes PHP, HTML, JavaScript y JQuery. • Conocimientos en Sistemas Webs basados en el Framework Symfony2. • Manejo del Framework Sonata para la administración y la seguridad del Sistema. • Conocimientos de los Gestores de Base de Datos PostgreSQL y SQL Server 2008 R2. • Manejo del Sistema Operativo Linux, Distribución Debian 7.0 Wheezy. 			
Habilidades Personales			
<ul style="list-style-type: none"> • Trabajo en equipo y trabajo bajo presión. • Capacidad de Análisis e Investigación. • Disciplinado. 			
Experiencia			
Dos años de experiencia en Desarrollo de Sistemas Informáticos.			
Cargo Sugerido			
Analista - Programador de la DTIC – MINSAL.			

Tabla 13-7 Perfil de Desarrollador de Aplicaciones Web.



Perfil – Capacitador.

Perfil:	Capacitador.	Código:	CAP_USR
Nivel Académico			
Ingeniero de Sistemas Informáticos, Computación o Áreas afines.			
Conocimientos			
Conocimientos Técnicos			
<ul style="list-style-type: none"> • Conocimientos en Administración de Sistemas Informáticos. • Conocimientos en Sistemas Webs basados en el Framework Symfony2. • Dominio y Manejo total del Producto Final (Sistema Informático de Mantenimiento a Implementar). • Conocimiento y Dominio de los Procesos de Negocio del Área de Mantenimiento, así como de los Conceptos Técnicos utilizados en el Área. • Facilidad para desarrollo de material didáctico, documentos de apoyo, material audiovisual y otros similares. 			
Habilidades Personales			
<ul style="list-style-type: none"> • Facilidad de expresión y comunicación, dándose a entender con claridad y sencillez, utilizando términos técnicos cuando fuere necesario. • Capacidad de enseñanza. • Trabajo en equipo y tolerancia a los demás. 			
Experiencia			
Un año de experiencia en actividades de capacitación o similares.			
Cargo Sugerido			
Analista – Programador de la DTIC – MINSAL.			

Tabla 13-8 Perfil de Capacitador.



Perfil – Coordinador.

Perfil:	Coordinador.	Código:	COO_IMP
Nivel Académico			
Ingeniero de Sistemas Informáticos, Computación o Áreas afines. Jefe de Departamento de Mantenimiento o Director de Establecimiento.			
Conocimientos			
Conocimientos Técnicos			
<ul style="list-style-type: none"> • Conocimientos en Administración de Sistemas Informáticos. • Conocimiento de herramientas o software para la Gestión de Proyectos y Seguimiento de Proyectos, Calendarización y similares. 			
Habilidades Personales			
<ul style="list-style-type: none"> • Facilidad de expresión y comunicación. • Capacidad para toma de decisiones. • Capacidad para coordinar y gestionar diversos recursos como personal, fechas, actividades, reuniones y otras necesidades emergentes. 			
Experiencia			
Un año de experiencia en actividades de coordinación interinstitucional.			
Cargo Sugerido			
Analista – Programador por parte de la DTIC, Jefe del Área de Mantenimiento por parte del Establecimiento donde se implementará.			

Tabla 13-9 Perfil de Administrador General del Sistema.



13.2.3 Arquitectura

El diseño Arquitectónico presenta la forma en que se organiza el sistema, de modo que este satisfaga los requisitos funcionales y no funcionales. El Diseño Arquitectónico da un punto de vista del sistema que permite verlo como un todo, y a la vez constituye el punto de partida de los subsiguientes modelos de diseño⁸³.

En este sentido, la Arquitectura del Sistema se vuelve una de las herramientas de comunicación más importantes entre los interesados del desarrollo del Sistema, y para la Implementación del Sistema no es la excepción, pues esta sirve como una guía para diversos aspectos de diseño y que en esta etapa serán llevados a cabo o desplegados físicamente.

Los aspectos o elementos más importantes a considerar con respecto al Diseño Arquitectónico para la Implementación del Sistema son:

- Los Patrones Arquitectónicos del Sistema:
 - Es un Sistema con un Modelo de Tres Capas (vea la Sección 11.3.2 de este documento).
 - Combinado con el Patrón Modelo-Vista-Controlador (vea la Sección 11.3.3 de este documento).
- Como se estructuran lógicamente los Componentes del Sistema (vea la Sección 11.3.5 de este documento).
- Como se despliega físicamente el Sistema (vea la sección 11.3.6 de este documento).

⁸³ Pressman, S. R. (2010). Ingeniería del Software (7a ed.), 9.1.2 ¿Por qué es importante la arquitectura? McGraw-Hill.



13.3 Estrategia de Implementación.

La implementación de un nuevo sistema, implica modificaciones en los procesos de negocio de la empresa u organización donde será implementado, esto tiene un gran impacto en los resultados que pueda generar la organización, incidiendo positiva o negativamente. Por eso es sumamente importante, determinar la forma más adecuada en que el nuevo sistema será puesto en marcha.

Se han evaluado cuatro Métodos para la Implementación del Sistema de Mantenimiento, el Método Directo, Método en Paralelo, Método Piloto y Método por Etapas.

No puede decirse que Método de Implementación es mejor que otro, ya que esto depende de diversos factores como la situación de la organización, el tamaño o extensión de la misma, los departamentos involucrados, la complejidad de sistema, la disponibilidad de recursos (tiempo, presupuesto y personal) y los problemas que pueden surgir para la organización al tomar alguno de estos métodos.

Se ha realizado un estudio de cada uno de los Métodos y la Evaluación de los mismos⁸⁴. En dicho estudio, se justifica la selección del Método de Implementación más conveniente para el Sistema de Mantenimiento, de los cuales se ha determinado que el más conveniente es el Método Piloto, y para la extensión del Sistema al resto de la organización el Método en Paralelo (debido a que el Método Piloto, posteriormente combina el Método Directo o en Paralelo).

⁸⁴ Para conocer el estudio de los Métodos de Implementación realizado vea el Anexo 1 – Descripción y Evaluación de Métodos de Implementación de Sistemas, del documento de Plan de Implementación, dentro del directorio /Documentos/7 – Plan de Implementacion/ en el DVD anexo.



13.4 Actividades de Implementación.

El proceso de Implementación del SIM, sigue el proceso administrativo basado en macro actividades y desglosado en sub-actividades a realizar.

13.4.1 Planeación de la Implementación.

Objetivo	
Definir una guía sistemática de los elementos y entorno necesario de para la ejecución de la implementación del Sistema desarrollado.	
Actividades	
<ul style="list-style-type: none"> • Definir estrategias y metas. Para implementar el sistema desarrollado y su puesta en marcha. • Delimitación y selección de establecimiento(s). Para la correcta puesta en marcha del sistema desarrollado, es necesario realizar una delimitación de los posibles establecimientos que están aptos para poder implementar el sistema, aquellos que ya utilizan el SIAF y SINAB serían tomados en cuenta para su selección en primera instancia, ya que el SIM se lee datos de ambos sistemas. • Gestión de recursos disponibles y limitantes Permitirá gestionar el recurso humano requerido para la implementación; siendo este personal de la Dirección de Tecnologías de Información y Comunicaciones, Área de Mantenimiento u Personal del Establecimiento dónde se implementará, así como aquel que se necesite contratar con cierto grado de especialización o conocimiento. • Preparación del ambiente organizacional. Establecer fechas para la implementación del sistema con el fin de acordar la disponibilidad del recurso humano correspondiente a las áreas o unidades involucradas. • Envío de comunicado a los establecimientos involucrados. Comunicar con anticipación (memorándum o correo electrónico) al establecimiento seleccionado y las demás áreas o unidades involucradas sobre la ejecución de la implementación del sistema. • Reunión con Directores. Para brindar una descripción del proyecto y del plan de implementación. 	
Tiempo	5 días.
Horas Hombre	112
Costo	\$880.16

Tabla 13-10 Planeación de la Implementación



13.4.1.1 Recurso Humano a Utilizar

Recurso Humano		
Recurso	Unidades	Descripción
Administrador de Proyecto	1	Responsable de definir las estrategias y objetivos de la implementación y seleccionará el recurso humano para ser asignado a la misma. Capaz de gestionar los recursos necesarios y dar directrices.
Coordinador de Informática	1	Apoyará con la gestión de recursos informáticos para preparar la implementación, la selección de establecimientos con capacidad para implementar el sistema así como de la comunicación entre las partes involucradas, como las reuniones.
Coordinador de Mantenimiento	1	Apoyará con la coordinación por parte del Área de Mantenimiento del Establecimiento seleccionado, estableciendo el vínculo necesario con la DTIC.

Tabla 13-11. Recurso Humano Involucrado en la Planeación de la Implementación.

13.4.2 Organización de la Implementación.

Objetivo	
Organizar y planificar el recurso humano necesario para la implementación del sistema desarrollado.	
Actividades	
<ul style="list-style-type: none"> • Definición de equipo ejecutor. Definir la estructura organizativa del equipo que llevará a cabo la ejecución de la implementación del sistema desarrollado, la interrelación de cada una de las partes, líneas de mando y el establecimiento de los medios de comunicación entre los participantes del proceso. • Estudio de perfiles y funciones del equipo ejecutor. Revisar los perfiles de cada uno de los puestos que conformaran el equipo ejecutor; para ello se convocan a los participantes de la implementación para una revisión de las expectativas de las funciones de cada uno. • Definición de responsabilidades. Asignar actividades a cada uno de los miembros del equipo ejecutor y su respectiva responsabilidad dentro de esta. 	
Tiempo	8 días.
Horas Hombre	120
Costo	\$910.88

Tabla 13-12 Organización de la Implementación



13.4.2.1 Recurso Humano a Utilizar

Recurso Humano		
Recurso	Unidades	Descripción
Administrador de Proyecto	1	Responsable de seleccionar el recurso humano adecuado para la ejecución de la implementación, asignando las responsabilidades correspondientes y delegando actividades.
Coordinador de Informática	1	Apoyará al administrador en la selección del recurso humano y colaborando con las gestiones necesarias.

Tabla 13-13. Recurso Humano Involucrado en la Organización de la Implementación.

13.4.3 Ejecución de la Implementación.

Objetivo
Ejecutar las actividades necesarias para la puesta en marcha del proyecto desarrollado bajo el nombre, "Sistema Informático para planeación y gestión de los procesos de servicio de mantenimiento del Área de Mantenimiento General, Nivel Regional y Local del Ministerio de Salud".
Actividades
<ul style="list-style-type: none"> • Instalación y configuración del servidor. Para realizar esta actividad los pasos a seguir se detallan en el manual de instalación del sistema SIM, los cuales son: <ul style="list-style-type: none"> ▪ Repositorios de Debian. ▪ Apache ▪ PHP 5 ▪ PostgreSQL 9.1 ▪ JasperReports Server ▪ Conectividad de la Red • Instalación y configuración del SIM. Para realizar esta actividad los pasos a seguir se detallan en el manual de instalación del sistema SIM, los cuales son: <ul style="list-style-type: none"> ▪ Instalación del SIM. ▪ Carga de la Base de Datos. ▪ Carga de los Reportes ▪ Configuración del Usuario de Acceso al SIM ▪ Ingreso de datos iniciales de los diferentes elementos que son críticos para el funcionamiento del sistema tales como: inventario técnico, empleados, empresas, estructura orgánica de los establecimientos y demás catálogos necesarios. • Instalación y configuración de las máquinas clientes. Para realizar esta actividad los pasos a seguir se detallan en el manual de instalación del sistema SIM, los cuales son: <ul style="list-style-type: none"> ▪ Requerimiento de Hardware ▪ Sistema Operativo



- Navegador Web
 - Conectividad de la Red
- **Prueba de funcionamiento del sistema.**
 Consiste en descubrir posibles problemas con el funcionamiento del sistema, en lo relacionado con: problemas técnicos, errores de ingreso de datos, de software y de red. Se recomienda la realización de los siguientes puntos:
 - Diseño de escenarios de pruebas.
 - Preparación de datos de pruebas.
 - Ejecución de documentación de pruebas.
 - Registrar errores y/o defectos.
 - Realizar correcciones.
 - Reporte de resultados de pruebas.
 - **Preparación del plan de capacitación.**
 Preparación del personal que estará asignado al proceso de capacitación y adiestramiento, definiendo los grupos de usuarios que se capacitarán con sus respectivos contenidos. Las actividades y contenidos de la capacitación se han definido en el Anexo 2 – del documento de Plan de Implementación del DVD anexo, apoyado con el Manual de Usuario.
 Grupos de Usuarios definidos:
 - Administradores y Jefes de Mantenimiento.
 - Técnicos de Mantenimiento.
 - Jefes de Unidad.
 - Solicitantes.
 El Manual de Usuario posee los siguientes temas:
 - Acceso al Sistema.
 - Configuración – Mi Cuenta.
 - Empresa.
 - Administración de Bien.
 - Contrato.
 - Mantenimiento Preventivo.
 - Solicitud de Mantenimiento.
 - Orden de Trabajo.
 - Informe de Orden de Trabajo
 - **Elaboración del material didáctico y audiovisual.**
 Imprimir los manuales de usuario y de administración según la cantidad de cada tipo de usuario.
 Elaborar las diapositivas o presentaciones necesarias para la capacitación.
 - **Gestión de recursos para la capacitación.**
 Gestionar lugar de capacitación. Las salas de capacitación deberán ser lo suficientemente amplias para ubicar a cada usuario.
 El recurso que estará a disposición durante la capacitación será:
 - Computadora individual para cada participante. Esta deberá estar configurada para tener acceso a la red interna del Ministerio de Salud, para poder ingresar al sistema.
 - Manual de Usuario.



<ul style="list-style-type: none"> ▪ Manual de Administración (según el tipo de usuario). ▪ Libreta de anotaciones para cada usuario, lápiz o lapicero. ▪ Computadora por cada capacitador. ▪ Proyector para cada capacitador. <ul style="list-style-type: none"> • Promoción y convocatoria de capacitación. Las áreas o departamentos involucradas del MINSAL, para la capacitación del sistema son las siguientes con los respectivos puestos: Área de Mantenimiento General <ul style="list-style-type: none"> ▪ Jefe de Mantenimiento General. ▪ Jefe de Sección de Equipo Médico. ▪ Jefe de Capacitaciones. ▪ Técnicos de las diferentes secciones (6). ▪ Guardalmacén. Hospital seleccionado <ul style="list-style-type: none"> ▪ Jefe de Mantenimiento Local ▪ Técnicos de las diferentes secciones (6). ▪ Jefes de departamento, área médica, administrativa o de apoyo (5). ▪ Jefes de sub-unidades o ambientes (5). Dirección de Tecnologías de Información y Comunicaciones <ul style="list-style-type: none"> ▪ Analista Programador 	
<ul style="list-style-type: none"> • Capacitación de usuario. Se realizarán siete capacitaciones consecutivas con los grupos definidos para la capacitación (detallado en el Anexo 2 del documento de Plan de Implementación dentro del directorio /Documentos/7 – Plan de Implementacion/ del DVD anexo). 	
Tiempo	40 días.
Horas Hombre	200
Costo	\$12,201.84

Tabla 13-14. Ejecución de la Implementación.



13.4.3.1 Recurso a Utilizar

La ejecución de las actividades en la capacitación y adiestramiento de usuarios, se requiere de recurso técnico, humano y materiales que den soporte y completen dichas actividades, por lo cual se presentan a continuación en la Tabla 13-15, Tabla 13-16 y Tabla 13-17 respectivamente:

Recurso Técnico		
Recurso	Unidades	Descripción
Servidor de Aplicaciones y Bases de Datos	1	Permitirá el alojamiento de los archivos fuentes del sistema, así como de la base de datos.
Computadoras	20	Se utilizarán 2 computadoras para capacitadores y 18 para los usuarios, se realizarán parejas de 2 en la primera capacitación debido a que en ella asistirán los 4 grupos y se hacen un total de 28 usuarios los cuales serán divididos en 2 salas.
Direcciones IP	20	Direcciones IP de acceso a la red local y al sistema.
Proyector	2	Para las presentaciones y ejemplificación del manejo de la herramienta.
Energía en la Sala de Capacitación	Según horas de capacitación	Energía eléctrica disponible en el local que se utilizará para las capacitaciones.
Sala de Capacitación	2	Cada sala de capacitación debe tener una capacidad de espacio para ubicar 10 máquinas y 20 usuarios, para no estar en un espacio muy reducido que dificulte el movimiento.

Tabla 13-15 Recurso Técnico

Recurso Humano		
Recurso	Unidades	Descripción
Administrador de Proyecto	1	Responsable de analizar, planificar, coordinar y controlar la ejecución de las actividades que se han de llevar a cabo en la implementación del sistema desarrollado.
Coordinador de Informática	1	Apoyará con la gestión de recursos informáticos, selección de establecimientos con capacidad para implementar el sistema así como de la comunicación entre las partes involucradas.
Coordinador de Mantenimiento	1	Apoyará con los conocimientos necesarios que se deben saber sobre el entorno del negocio del área de mantenimiento.



Recurso Humano		
Desarrollador de Aplicaciones	2	Dará mantenimiento a mejoras futuras al sistema, así como corrección de los bugs que puedan descubrirse en la implementación y uso.
Capacitador	4	Brindará las capacitaciones a los usuarios sobre la utilización del sistema.
Administrador de Servidores	1	Estará apoyando con la instalación y configuración de los servidores web donde estará alojado el sistema, la base de datos, el sistema operativo, las virtualizaciones necesarias y el acceso a la red.

Tabla 13-16 Recurso Humano

Materiales		
Recurso	Unidades	Descripción
Manual de Usuario	24	Guía de usuario para el uso del sistema SIM.
Manual de Administración	4	Guía de usuario para la administración del sistema SIM.
Libreta de Anotaciones	28	Ayudará a tomar notas a los usuarios
Lápiz	28	Para escribir en la libreta de anotaciones.

Tabla 13-17 Materiales



13.4.3.2 Costos Basados en Horas Hombres.

Nombre de tarea	Duración	Comienzo	Fin	Costo	Horas
Implementación del SIM	57 días	lun 05/05/14	mar 22/07/14	\$14,250.94	922
Planeación de la Implementación	5 días	lun 05/05/14	vie 09/05/14	\$880.16	112
Definir estrategias y metas	1 día	lun 05/05/14	lun 05/05/14	\$62.88	8
Delimitación y selección de establecimientos	0.5 días	lun 05/05/14	lun 05/05/14	\$95.20	12
Gestión de los recursos disponibles y limitantes	2 días	mar 06/05/14	mié 07/05/14	\$380.80	48
Preparación del ambiente organizacional	1 día	jue 08/05/14	jue 08/05/14	\$190.40	24
Envío de comunicado a los establecimientos involucrados	0.5 días	jue 08/05/14	jue 08/05/14	\$23.36	4
Reunión con Directores	1 día	vie 09/05/14	vie 09/05/14	\$127.52	16
Organización de la Implementación	8 días	lun 12/05/14	mié 21/05/14	\$910.88	120
Definición del equipo ejecutor	5 días	lun 12/05/14	vie 16/05/14	\$628.80	80
Estudios de perfiles y funciones del equipo ejecutor	1 día	lun 19/05/14	lun 19/05/14	\$62.88	8
Definición de responsabilidades	2 días	mar 20/05/14	mié 21/05/14	\$219.20	32
Ejecución de la Implementación	40 días	jue 22/05/14	mié 16/07/14	\$12,201.84	200
Instalación y configuración del Servidor	3 días	jue 22/05/14	lun 26/05/14	\$140.16	24
Instalación y configuración del SIM	1 día	mar 27/05/14	mar 27/05/14	\$93.44	16
Instalación y configuración de las máquinas clientes	5 días	mié 28/05/14	mar 03/06/14	\$9,334.40	160
Prueba de funcionamiento del sistema	16.5 días	mié 04/06/14	jue 26/06/14	\$1,541.76	416
Diseño de escenarios de pruebas	3 días	mié 04/06/14	vie 06/06/14	\$280.32	48
Preparación de datos de pruebas	3 días	lun 09/06/14	mié 11/06/14	\$280.32	48
Ejecución de documentación de pruebas	2 días	jue 12/06/14	vie 13/06/14	\$186.88	32
Registrar errores y/o defectos	0.5 días	lun 16/06/14	lun 16/06/14	\$46.72	8
Realizar correcciones	5 días	lun 16/06/14	lun 23/06/14	\$467.20	80
Reporte de resultados de pruebas	3 días	lun 23/06/14	jue 26/06/14	\$280.32	48
Preparación del plan de capacitación	2 días	jue 26/06/14	lun 30/06/14	\$186.88	32
Elaboración del material didáctico y audiovisual	3 días	lun 30/06/14	jue 03/07/14	\$560.64	96
Gestión de recursos para la capacitación	1 día	jue 03/07/14	vie 04/07/14	\$93.44	16
Promoción y convocatoria de capacitación	1 día	vie 04/07/14	lun 07/07/14	\$46.72	8
Capacitación de usuario	7 días	mar 08/07/14	mié 16/07/14	\$204.40	35
CAP-01	1 día	mar 08/07/14	mar 08/07/14	\$17.52	3



Nombre de tarea	Duración	Comienzo	Fin	Costo	Horas
CAP-02	1 día	mié 09/07/14	mié 09/07/14	\$29.20	5
CAP-03	1 día	jue 10/07/14	jue 10/07/14	\$35.04	6
CAP-04	1 día	vie 11/07/14	vie 11/07/14	\$35.04	6
CAP-05	1 día	lun 14/07/14	lun 14/07/14	\$29.20	5
CAP-06	1 día	mar 15/07/14	mar 15/07/14	\$17.52	3
CAP-07	1 día	mié 16/07/14	mié 16/07/14	\$40.88	7
Control de la Implementación	4 días	jue 17/07/14	mar 22/07/14	\$258.06	39
Verificación de cumplimiento de actividades	1 día	jue 17/07/14	jue 17/07/14	\$94.32	12
Elaboración de informes de implementación	2 días	vie 18/07/14	lun 21/07/14	\$140.16	24
Verificación de informes de implementación	1 día	mar 22/07/14	mar 22/07/14	\$23.58	3



13.4.3.3 Resumen de Costos basados en Horas Hombres.

Macro actividad	Costo
Planeación de la Implementación	880.16
Organización de la Implementación	910.88
Ejecución de la Implementación	12,201.84
Control de la Implementación	258.06
Total	\$14,250.94

Tabla 13-18 Resumen de Costos Basados en Horas Hombres

13.4.3.4 Costo por Materiales.

Materiales			
Recurso	Unidades	Costo Unitario	Costo Total
Manual de Usuario	24	6.00	144.00
Manual de Administración	4	1.80	7.20
Libreta de Anotaciones	28	1.50	42.00
Lápiz	28	0.15	4.20
Totales			\$197.40

Tabla 13-19 Costo por Materiales

13.4.3.5 Total de Costos.

Detalle de Costos	Total
Total de costos por horas hombre	14,250.94
Total de costos por materiales	197.40
Total final de costos	\$14,448.34

Tabla 13-20 Total de Costos



13.4.4 Control de la Implementación.

Objetivo	
Asegurar la calidad de la implementación del sistema desarrollado según lo planificado y programado.	
Actividades	
<ul style="list-style-type: none"> • Verificación de cumplimiento de actividades. Verificar que la implementación del sistema se realice según lo planificado; para que de esta manera logre establecer la trayectoria a seguir ante posibles variaciones; creando así, una actividad de auditoría con lo que se persigue, verificar que la implementación se ejecuta según los elementos de control establecidos. • Elaboración de informes de Implementación. Elaborar los informes sobre el seguimiento del plan de implementación del sistema y de los inconvenientes detectados durante las actividades de verificación realizadas como parte del control de avance tanto financiero como de recurso humano y tiempo. • Verificación de Informes de Implementación. Verificar los informes creados en la implementación del sistema para su posterior análisis de resultados. 	
Tiempo	4 días.
Horas Hombre	39
Costo	\$258.06

Tabla 13-21. Control de la Implementación.

13.4.4.1 Recurso Humano Involucrado

Recurso Humano		
Recurso	Unidades	Descripción
Administrador de Proyecto	1	Verificará el cumplimiento y avance de las actividades de implementación. Revisará los informes de la implementación (Formularios), aplicará los Índices de Control de Avance y tomará acciones de corrección.
Desarrollador de Aplicaciones	2	Elaborar Formulario de Control de Implementación.
Capacitador	4	Elaborar Formularios de Control de Implementación.
Administrador de Servidores	1	Elaborar Formularios de Control de Implementación.

Tabla 13-22. Recurso Humano Involucrado en el Control de la Implementación.



13.5 Cronograma de Implementación.

Nombre de tarea	Duración	Predecesoras	Nombres de los recursos
Implementación del SIM	57 días		
Planeación de la Implementación	5 días		
Definir estrategias y metas	1 día		Administrador del Proyecto
Delimitación y selección de establecimientos	0.5 días		Administrador del Proyecto, Coordinador de Informática, Coordinador de Mantenimiento
Gestión de los recursos disponibles y limitantes	2 días	3,4	Administrador del Proyecto, Coordinador de Informática, Coordinador de Mantenimiento
Preparación del ambiente organizacional	1 día	5	Administrador del Proyecto, Coordinador de Informática, Coordinador de Mantenimiento
Envío de comunicado a los establecimientos involucrados	0.5 días	5	Coordinador de Informática
Reunión con Directores	1 día	6,7	Coordinador de Informática, Coordinador de Mantenimiento
Organización de la Implementación	8 días		
Definición del equipo ejecutor	5 días	8	Administrador del Proyecto, Coordinador de Informática
Estudios de perfiles y funciones del equipo ejecutor	1 día	10	Administrador del Proyecto
Definición de responsabilidades	2 días	11	Administrador del Proyecto, Coordinador de Informática
Ejecución de la Implementación	40 días		
Instalación y configuración del Servidor	3 días	12	Administrador de Servidores
Instalación y configuración del SIM	1 día	14	Administrador de Servidores, Desarrollador de Aplicaciones
Instalación y configuración de las máquinas clientes	5 días	15	Desarrollador de Aplicaciones
Prueba de funcionamiento del sistema	16.5 días		
Diseño de escenarios de pruebas	3 días	16	Desarrollador de Aplicaciones
Preparación de datos de pruebas	3 días	18	Desarrollador de Aplicaciones
Ejecución de documentación de pruebas	2 días	19	Desarrollador de Aplicaciones
Registrar errores y/o defectos	0.5 días	20	Desarrollador de Aplicaciones
Realizar correcciones	5 días	21	Desarrollador de Aplicaciones
Reporte de resultados de pruebas	3 días	22	Desarrollador de Aplicaciones



Nombre de tarea	Duración	Predecesoras	Nombres de los recursos
Preparación del plan de capacitación	2 días	23	Capacitador
Elaboración del material didáctico y audiovisual	3 días	24	Capacitador
Gestión de recursos para la capacitación	1 día	25	Capacitador, Coordinador de Informática
Promoción y convocatoria de capacitación	1 día	26	Capacitador
Capacitación de usuario	7 días		
CAP-01	1 día	27	Capacitador
CAP-02	1 día	29	Capacitador
CAP-03	1 día	30	Capacitador
CAP-04	1 día	31	Capacitador
CAP-05	1 día	32	Capacitador
CAP-06	1 día	33	Capacitador
CAP-07	1 día	34	Capacitador
Control de la Implementación	4 días		
Verificación de cumplimiento de actividades	1 día	35	Administrador del Proyecto
Elaboración de informes de implementación	2 días	37	Capacitador, Coordinador de Informática, Desarrollador de Aplicaciones
Verificación de informes de implementación	1 día	38	Administrador del Proyecto



13.6 Control de la Implementación

El Control de la Implementación, consiste en realizar actividades de supervisión para vigilar el avance del proceso de implementación y el cumplimiento de la programación de éste. Para ello se deben realizar controles adecuados sobre las actividades y el seguimiento de las mismas. Es por ello que se propone la siguiente Metodología de Control de la Implementación:

13.6.1 Metodología de Control de la Implementación

El Control de la Implementación, se basará en dos técnicas que se complementan:

- Formularios de Control y Medición
- Índices de Control de Avance

13.6.1.1 Formularios de Control

Con el fin de recolectar información sobre las actividades llevadas a cabo, se hace uso de formularios. Los cuales, con un formato preestablecido, serán completados por las personas encargadas de la ejecución de las actividades de implementación.

El llenado de formularios de control es realizado principalmente por los encargados de cada actividad de implementación, quienes deben ingresar datos sobre fechas de inicio o fin, porcentaje de avance, entre otros; que al mismo tiempo servirán como insumos para el cálculo de Índices de Control.

Esto permite un control de las actividades realizadas por el personal y el porcentaje de avance de las mismas, y por supuesto, contar con la información necesaria para una periódica revisión del progreso total de la implementación, y de esta forma realizar ajustes, verificar problemas y darles una solución viable, para evitar el retraso del proyecto o el aumento de costos.

Los formularios a utilizar para el seguimiento y control de la implementación son los siguientes:

- Formulario de Control de Responsabilidades
- Formulario de Control de Actividades
- Formulario de Control de Costos



Formulario de Control de Responsabilidades

Para llevar el control de los responsables de cada actividad a realizar para la implementación del Sistema, se utilizarán los Formularios de Asignación de Responsabilidades. De esta forma se tendrá oportunamente datos del responsable.

Este formulario poseerá la siguiente estructura (Tabla 13-23):

No: ISIM-FCR-0000			
Ministerio de Salud de El Salvador - MINSAL Dirección de Tecnologías de Información y Comunicación - DTIC Implementación del Sistema Informático de Mantenimiento - SIM			
Formulario de Control de Responsabilidades			Pág.: 1 De: 2
Información del Responsable:			
Nombre del Empleado:	3		
Teléfono:	4	Email:	5
		Rol Asignado:	6
Actividades Asignadas a su Cargo			
Código de la Actividad	Nombre de la Actividad	Fecha	Notificado
7	8	9	10
Elaborado por:	11	Fecha Ela.:	12
Revisado por:	13	Fecha Rev.:	14
Observaciones:	15		16
			Espacio para Firmas o Sellos

Tabla 13-23. Formato para Formulario de Control de Responsabilidades.



Instructivo de Llenado del Formulario de Control de Responsabilidades (Cada numeral a continuación, corresponde con el número de casilla del Formulario en la Tabla 13-23):

- 1) Identificar el número de página del formulario.
- 2) Anotar el número total de páginas del mismo formulario.
- 3) Escribir el nombre del empleado al cual le hará responsable de una determinada actividad.
- 4) Registrar el teléfono del empleado.
- 5) Registrar el correo electrónico del empleado.
- 6) Anotar el que desempeña el empleado en el proceso de instalación del Sistema (según los perfiles establecidos en la Sección 13.2.2.1).
- 7) Identificar el Código de la Actividad de la cual es responsable el empleado.
- 8) Escribir el Nombre de la Actividad de la cual es responsable el empleado.
- 9) Anotar Fecha en la que se estableció como responsable de la actividad.
- 10) Especificar si el empleado ha sido notificado de su responsabilidad con respecto a la tarea.
- 11) Anotar nombre de la persona quien elabora el formulario.
- 12) Fecha de elaboración del formulario.
- 13) Anotar nombre de la persona quien revisa y da valides al formulario.
- 14) Fecha de la revisión del formulario.
- 15) Agregar observaciones en caso de ser necesario.
- 16) Utilizar el espacio para Firma y Sello en caso de ser necesario.

Formulario de Control de Actividades

El propósito del Formulario de Control de Actividades, es conocer principalmente las fechas y el avance de cada una de ellas para una periódica revisión del progreso total de la implementación, y la recolección de datos para el cálculo de Índices de Control.

Este formulario poseerá la siguiente estructura (Tabla 13-24):



Instructivo de llenado del Formulario de Control de Actividades (Cada numeral a continuación, corresponde con el número de casilla del Formulario en la Tabla 13-24):

- 1) Identificar el número de página del formulario.
- 2) Anotar el número total de páginas del mismo formulario.
- 3) Especificar la Fecha de Inicio del Informe.
- 4) Especificar la Fecha de Finalización del Informe.
- 5) Escribir el nombre del empleado responsable de la actividad.
- 6) Identificar el Código de la Actividad a realizar.
- 7) Escribir el Nombre de la Actividad a realizar.
- 8) Si la actividad se ha finalizado marcar la casilla con una X.
- 9) Si la actividad aún no se ha finalizado marcar la casilla con una X.
- 10) Al finalizar, especificar la cantidad de actividades esperadas a la fecha.
- 11) Escribir el porcentaje de avance de las actividades esperadas.
- 12) Al finalizar, especificar la cantidad de actividades realizadas.
- 13) Escribir el porcentaje de avance que representan las actividades realizadas.
- 14) Al finalizar, especificar la cantidad de actividades pendientes.
- 15) Escribir el porcentaje que representan las actividades pendientes.
- 16) Anotar el total de actividades realizadas y pendientes.
- 17) Anotar el total del porcentaje actividades realizadas y pendientes.
- 18) Anotar nombre de la persona quien elabora el formulario.
- 19) Fecha de elaboración del formulario.
- 20) Anotar nombre de la persona quien revisa y da validez al formulario.
- 21) Fecha de la revisión del formulario.
- 22) Agregar observaciones en caso de ser necesario.
- 23) Utilizar el espacio para Firma y Sello en caso de ser necesario.



Formulario de Control de Costos

Para llevar un Control más efectivo de los costos en que se incurre al realizar las tareas, se utiliza el Formulario de Control de Costos, esto servirá como medida para determinar los costos reales, versus los costos estimados. Si hay demasiados costos, se deberán realizar los ajustes respectivos de reducción de costos, para evitar sobrepasar el presupuesto asignado.

Este formulario poseerá la siguiente estructura (Tabla 13-25):

No: ISIM-FCC-0000			
Ministerio de Salud de El Salvador - MINSAL Dirección de Tecnologías de Información y Comunicación - DTIC Implementación del Sistema Informático de Mantenimiento - SIM			
Formulario de Control de Costos			Pág.: 1
Periodo:	Fecha Inicio:	3	Fecha Fin:
		4	De: 2
Responsable de la Tarea:	5		
Código de la Actividad	Nombre de la Actividad	Fecha	Costo (\$)
6	7	8	9
Total Costos Incurridos:			10
Elaborado por:	11		Fecha Ela.:
		12	Fecha Rev.:
Revisado por:	13		14
Observaciones:	15		16
			Espacio para Firmas o Sellos

Tabla 13-25. Formato para Formulario de Control de Gastos.



Instructivo de llenado del Formulario de Control de Costos (Cada numeral a continuación, corresponde con el número de casilla del Formulario en la Tabla 13-25):

- 1) Identificar el número de página del formulario.
- 2) Anotar el número total de páginas del mismo formulario.
- 3) Especificar la Fecha de Inicio del informe.
- 4) Especificar la Fecha de Finalización del informe.
- 5) Escribir el nombre del empleado responsable de la tarea.
- 6) Identificar el Código de la Actividad a la que pertenece la Tarea.
- 7) Escribir el Nombre de la Actividad a realizar.
- 8) Anotar la Fecha en la que se incurrió en el costo.
- 9) Agregar el costo de la tarea.
- 10) Especificar la sumatoria de los Costos Incurridos.
- 11) Anotar nombre de la persona quien elabora el formulario.
- 12) Fecha de elaboración del formulario.
- 13) Anotar nombre de la persona quien revisa y da valides al formulario.
- 14) Fecha de la revisión del formulario.
- 15) Agregar observaciones en caso de ser necesario.
- 16) Utilizar el espacio para Firma y Sello en caso de ser necesario.

13.6.1.2 Índices de Control de Avance

Los cálculos de los índices de avance del proyecto son sumamente beneficiosos, pues al ser realizados, se puede extraer conclusiones de la efectividad de la ejecución del plan, mediante los cuales será posible la toma de decisiones que permitan realizar ajustes, tomar medidas correctivas o cambiar la dirección de las actividades del plan de implementación, según sea requerido.

La descripción de los índices que se recomienda utilizar y su forma de cálculo se presentan a continuación:

1. Índice de Actividades Programadas Ejecutadas (IAPE)

Este índice permite conocer el grado de avance del plan de implementación.

Cálculo:

$$IAPE = \frac{\sum \text{Duración de las Actividades Programadas Ejecutadas}}{\sum \text{Duración de Todas las Actividades Programadas}}$$

**Objetivo:**

- Si el resultado del índice es menor que 1 se determina que el grado de avance del proyecto se encuentra en estado aceptable, aunque debe evaluarse el número de actividades programadas ejecutadas contra el número total de actividades.
- Si el resultado es mayor que 1 deben tomarse medidas correctivas.

2. Índice de Duración de Actividades (IDA)

Índice que permite conocer el grado de desviación entre el tiempo real de una actividad cualquiera y el tiempo que se tenía programado para dicha actividad.

Cálculo:

$$IDA = \frac{\sum \text{Duración Real de la Actividad}}{\sum \text{Tiempo Programado para la Actividad}}$$

Objetivo:

- Si el resultado del índice es menor que 1, se determina que el grado de avance de dicha actividad se encuentra en estado aceptable.
- Si el resultado es mayor que 1 deben tomarse medidas correctivas.

3. Índice de Actividades Programadas Retrasadas (IAPR)

Índice que permite conocer el grado de retraso con respecto a la duración programada de la implementación.

Cálculo:

$$IAPR = \frac{\sum \text{Tiempo de Retraso de las Actividades}}{\sum \text{Duración de Todas las Actividades}}$$

Objetivo:

- Si el resultado es cercano a cero, se determina que una actividad se encuentra en niveles aceptables.
- Si el resultado es cercano a 1 o mayor que 1, se deben tomar medidas correctivas.

4. Índice de Rendimiento de las Actividades (IRA)

Permite conocer el grado de variación entre el costo real de la actividad y el costo planificado para la actividad.



Definición:

$$IRA = \frac{\sum \text{Costo Real de la Actividad}}{\sum \text{Costo Programado para la Actividad}}$$

Finalidad:

- Si el resultado es menor que 1, se establece que se encuentra en estado aceptable, de lo contrario, debe mejorarse la aplicación de los gastos que está teniendo la actividad específica.

Se recomienda que se le apliquen los índices respectivos a las actividades que se ejecutaran para la implementación, principalmente a las actividades que se realizarán antes que la Capacitación se lleve a cabo, es decir, aquellas relacionadas con la Instalación y Configuración del Servidor, del Sistema y Maquinas Cliente, así como las Relacionadas con las Pruebas del Sistema y la Preparación de la Capacitación; esto permitirá solventar inconvenientes para las actividades posteriores y evitar que se dé un retraso acumulado. Se debe tener en cuenta las actividades mencionadas, ya que por su naturaleza pueden tener gran influencia en el desarrollo de las siguientes actividades; por ello se debe ir verificando el cálculo de cada uno de los índices, para extraer conclusiones acertadas, de manera que el monitoreo sea el correcto.



14 Conclusiones

- El entendimiento del entorno de negocio junto con los procesos que esto conlleva, es de importancia para el equipo de desarrollo ya que esto permitirá un mejor análisis y diseño de la solución propuesta en el desarrollo del proyecto.
- La definición de una correcta Metodología es importante para el desarrollo del proyecto informático dado que esta brindará los pasos a seguir durante el desarrollo de este. La Metodología es la guía que expresa la forma de llevar a cabo el proyecto, desde el inicio del mismo hasta su finalización
- Una correcta y adecuada comunicación, así como el establecimiento de un compromiso mutuo es de gran importancia para el éxito del proyecto; debido a esto y al apoyo de la Dirección de Tecnología de Información y Comunicaciones junto con el Área de Mantenimiento General y el Equipo de Trabajo, se han establecido los lineamientos necesarios para el desarrollo del proyecto.

A esto se suma una adecuada utilización de estándares, tanto aquellos definidos por cuenta propia, como aquellos que han sido proporcionados como requerimientos por la DTIC. Esto da uniformidad al trabajo completo, que incluye aspectos de documentación, así como aspectos técnicos (estándares de codificación, de base de datos, etc.); también es indispensable para el trabajo en equipo, pues sirven de referente para el desarrollo del trabajo realizado por cada integrante.

- La utilización de estándares internacionales, probados y recomendados por diferentes organizaciones reconocidas favorece la culminación exitosa del proyecto. Estos estándares van desde detalles, como la utilización del estándar APA (estándar para elaborar citas y referencias en un documento) hasta estándares que rigen las mismas actividades del proyecto, como el estándar *IEEE 830-1998* para la Especificación de Requerimientos de Software.
- La selección de una adecuada Metodología de Desarrollo de Software es crucial en todo proyecto de sistemas informáticos. Esto debe realizarse mediante el estudio detallado de cada Modelo o Enfoque de Desarrollo de Software. Cada uno de los Modelos o Enfoques existentes es adecuado para determinados tipos de proyectos, es decir, que la elección de un Modelo o Enfoque se debe basar en diversas consideraciones técnicas y en características del mismo proyecto, así como del equipo que va a ejecutar el proyecto.
- Uno de los principales resultados de la etapa de diseño es la Arquitectura del Sistema, cuyo desarrollo está lleno de diversos puntos de vista, opiniones y consejos de cómo realizarlo. Sin embargo, no hay una guía concreta y correcta de cómo hacerlo y cómo saber cuándo la arquitectura está verdaderamente completa. Por tanto, esta debe hacerse a partir de una recopilación, investigación y análisis de diversas fuentes



- El beneficio social que aportara el desarrollo del proyecto será de alto impacto, ya que beneficia de forma indirecta a la población salvadoreña que acude a los Establecimientos de Salud del MINSAL, en cuanto a calidad de atención y de prestación de los servicios necesarios que garanticen la conservación y recuperación de la Salud.

Así mismo proporcionara un beneficio para las diferentes dependencias del MINSAL, permitiendo el control del personal de mantenimiento, control de costos, información oportuna y de utilidad, y la preservación de los diferentes equipos, instalaciones e infraestructura perteneciente.



15 Recomendaciones

Al estudiante y equipos de desarrollo:

- Seguir la metodología de desarrollo establecida, para una correcta coordinación, y estandarización de las herramientas y métodos que se utilizarán en las actividades del ciclo de vida del desarrollo del proyecto.
- Establecer una correcta comunicación entre los miembros del equipo de desarrollo y los usuarios directos e indirectos, obteniendo con ello la correcta determinación de los requerimientos del sistema y las funcionalidades que este brindará, despejando dudas y llegando a consensos en caso de ser necesarios. Conviene seguir los manuales de procedimientos y normas establecidas en la organización, sin embargo, no está demás realizar observaciones y propuestas de mejora de los mismos.
- Seguir lo más preciso que sea posible el cronograma de actividades establecido, respetando los hitos de entrega, para evitar atrasos en los tiempos de finalización de cada actividad, lo que evitará incurrir en más gasto de tiempo, dinero y recurso humano.
- Realizar una correcta delimitación del proyecto y establecer que funcionalidades se han de desarrollar y cuáles no, a través de la creación del Acta de Constitución del Proyecto, en concordancia con los usuarios directos, quienes brindarán observaciones y aprobarán dicha Acta, logrando así un claro entendimiento de las funcionalidades a desarrollar en el proyecto. Esto brinda formalidad al proyecto y se crea un compromiso entre las partes que colaboraran para llevarlo a cabo, y constituye a la vez un marco de referencia.
- Prever con anticipación los riesgos que puedan generar impacto en el desarrollo del proyecto y de esta manera poder minimizarlos o mitigarlos, creando con ello la Gestión de Riesgos del Proyecto.

A las partes interesadas:

- A la Dirección de Tecnologías de Información y Comunicaciones (DTIC), mantener esa buena comunicación e interés que han mostrado sobre la realización y el seguimiento del proyecto, facilitando y propiciando las condiciones para una finalización exitosa.
- El personal del Área de Mantenimiento General, mantener esa claridad de las necesidades del área y la visión de mejorar los servicios de mantenimiento a través de la automatización, para poder obtener requerimientos correctos en posteriores mejoras o adición de características al Sistema Informático de Mantenimiento.
- Dialogar, coordinar y organizar las actividades necesarias entre la DTIC y el Áreas de Mantenimiento General, para dar inicio a la implementación de tan importante proyecto, siendo conscientes del gran beneficio social que presenta.



16 Glosario Técnico

	Termino	Definición
B	Base de Datos	Es una colección de datos lógicamente coherente con algún tipo de significado inherente.
	Bundle	<p>Un Bundle es un conjunto estructurado de archivos dentro de un directorio. Es un “paquete”, que presta una característica única o funcional basado en el Framework Symfony2.</p> <p>Todo es un Bundle en Symfony2, incluyendo el núcleo de la funcionalidad del Framework, inclusive el código creado en el desarrollo del SIM, el cuál funcionará a través de él. Para una mejor comprensión se podría entender un bundle como un paquete que contiene un conjunto estructurado de directorios y archivos que permiten brindar una funcionalidad, estos directorios, y archivos comprenden desde Clases tipos Controllers, Vistas, hasta archivos JavaScript, CSS, entre otros.</p> <p>Para más información vea la Sección 11.1.10 en la cual se define la Estructura General de Bundles.</p>
C	Controlador Frontal	Un controlado frontal (Front Controller), es un script corto de PHP, que por lo general maneja todas las peticiones del cliente en el lado del servidor y cuyo trabajo, en Symfony, consiste en arrancar la aplicación.
	CSS	Lenguaje de hoja de estilo usado para la descripción de la semántica de presentación (aparición y formato) de un documento comúnmente escrito en HTML.
D	DBAL	Es una interfaz de programación que unifica la comunicación entre una aplicación y uno o varios Gestores de Bases de Datos como SQL Server, PostgreSQL, Oracle o SQLite.
	DMZ (Zona Desmilitarizada)	<p>Es una red local que se ubica entre la red interna de una organización y una red externa, generalmente en Internet. El objetivo de una <i>DMZ</i> es tener una red separada a la que se pueda acceder tanto desde la red interna como por vía externa sin correr el riesgo de comprometer la seguridad de la compañía.</p> <p>Por lo general, la política de seguridad para la DMZ es la siguiente:</p> <ul style="list-style-type: none">• El tráfico de la red externa a la DMZ está autorizado• El tráfico de la red externa a la red interna está prohibido• El tráfico de la red interna a la DMZ está autorizado• El tráfico de la red interna a la red externa está autorizado• El tráfico de la DMZ a la red interna está prohibido• El tráfico de la DMZ a la red externa está denegado



	Termino	Definición
D	Doctrine	Es un mapeador de objetos-relacional (ORM) escrito en PHP que proporciona una capa de persistencia para objetos PHP. Es una capa de abstracción que se sitúa justo encima de un SGBD. Para una definición más detallada vea la sección 8.8 y sección 11.3.4 del presente documento.
F	Framework	Palabra en ingles que significa marco de referencia. Es un conjunto estandarizado de conceptos, prácticas, criterios, librerías, programas y/o tecnologías para enfocar un tipo de problemática particular, que sirve como referencia para enfrentar y resolver nuevos problemas de índole similar.
G	Grupo	En el contexto de Privilegios de Usuarios se entiende como: Clasificación mediante la cual se definen distintos roles a los usuarios del sistema (Vea la definición de Rol en esta sección).
H	HTML	<i>Hipertext Markup Language</i> , o Lenguaje de Marcado de Hipertexto, el cual es el estándar utilizado para la elaboración de páginas web y presentación de contenido.
I	Instrucción SQL	Comando o instrucción escrita en SQL (Lenguaje Estructurado de Consulta), que permite llevar a cabo las tareas de consulta o manipulación de los datos de la base de datos.
J	JavaScript	Es un lenguaje de programación interpretado que se define como orientado a objetos. Permite mejoras en la interfaz de usuario y páginas web dinámicas.
L	LAN	Red de Área Local (del inglés, Local Area Network). Es una red de computadoras en la cual los usuarios se interconectan dentro de un área limitada como en un hogar, una escuela, una universidad o una oficina.
M	Máquina Virtual	Es un software que simula una computadora física y puede ejecutar programas como si fuese una computadora real.
	Modelo Vista Controlador	Es un patrón de arquitectura de software que separa los datos, la lógica de negocio y la interfaz de usuario de una aplicación. Para más
N	Navegador Web	Es un software que permite el acceso a Internet, interpretando la información de archivos y sitios web para que éstos puedan ser leídos.
O	Open SSL	Consiste en un robusto paquete de herramientas de administración y bibliotecas relacionadas con la criptografía, que suministran funciones criptográficas a otros paquetes como <i>OpenSSH</i> y navegadores web
	ORM	Mapeo Objeto Relacional, es una técnica utilizada en lenguajes de programación para eliminar la incompatibilidad de bases de datos relacionales, con las técnicas orientadas a objetos. En esencia, esto permite tener una "base de datos virtual" orientada a objetos.
P	Página Web	Es el nombre de un documento o información electrónica adaptada para la World Wide Web y que puede ser accedida mediante un navegador
	PDO	Es una extensión que provee una capa de abstracción de acceso a datos para PHP 5, con lo cual se consigue hacer uso de las mismas funciones para hacer consultas y obtener datos de distintos manejadores de bases de datos
	PHP	Es un lenguaje de programación de uso general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico.



	Termino	Definición
P	Puerto	Es una forma genérica de denominar a una interfaz a través de la cual los diferentes tipos de datos se pueden enviar y recibir. Dicha interfaz puede ser de tipo físico, o puede ser a nivel de software.
R	Request - Response	Es uno de los métodos básicos que utilizan los ordenadores para comunicarse entre sí. Cuando se utiliza request-response, una computadora solicita determinados datos y la computadora de destino responde a la solicitud.
	Rol	Un rol es un privilegio de acceso a cierta información del sistema y/o la realización de operaciones dentro del mismo.
S	Servidor	En informática, es un nodo (computadora) que forma parte de una red, el cual provee servicios a otros nodos denominados clientes.
	Servidor Web	Servidor que almacena documentos HTML, imágenes, archivos de texto, escrituras, y demás material web compuesto por datos (conocidos colectivamente como contenido), y distribuye este contenido a clientes que lo solicitan en la red.
	SGBD	Sistema de Gestión de Bases de Datos. Es un conjunto de programas que permiten el almacenamiento, modificación y extracción datos en una base de datos, además de proporcionar herramientas para añadir, borrar, modificar y analizar los datos.
	Sistema Informático	Conjunto de elementos interrelacionados entre sí, tales como hardware, software y de recurso humano que permite almacenar y procesar datos, generando información útil.
	Sonata Bundle.	Es un conjunto de paquetes que tiene por objetivo proporcionar un Framework que brinda ciertas características que facilitan la administración de los elementos del sistema, basados en Symfony 2, integrándose con éste y permitiendo así facilitar la construcción de sitios o aplicaciones web. Para más información vea la Sección 13.2.1.2 en la cual se describe el Bundle Sonata a mayor detalle.
	SSH	Del inglés Secure Shell, es el nombre de un protocolo que sirve para acceder a máquinas remotas a través de una red. Además de la conexión a otras computadoras, SSH permite manejar por completo la computadora mediante un intérprete de comandos y copiar datos de forma segura. SSH usa técnicas de cifrado que hacen que la información que viaja por el medio de comunicación vaya de manera no legible, evitando que terceras personas puedan descubrir el usuario y contraseña de la conexión, ni lo que se escribe durante toda la sesión.
	Symfony	Es un completo Framework diseñado para optimizar el desarrollo de las aplicaciones web basado en el patrón Modelo Vista Controlador. Para una definición más completa y detallada vea la sección 11.3.4 del presente documento.
U	URL	Localizador Uniforme de Recursos, es una secuencia de caracteres, de acuerdo a un formato modélico y estándar, que se usa para nombrar recursos en Internet para su localización o identificación, como por ejemplo documentos textuales, imágenes, vídeos, presentaciones digitales.



	Termino	Definición
W	WAN	Red de Área Ampla (del inglés Wide Area Network). Es una red de computadoras que abarca varias ubicaciones físicas, proveyendo servicio a una zona, un país, incluso varios continentes. Es cualquier red que une varias redes locales, llamadas LAN, por lo que sus miembros no están todos en una misma ubicación física.
X	XML	Siglas en inglés de <i>eXtensible Markup Language</i> es un lenguaje de marcas desarrollado por el World Wide Web Consortium (W3C) utilizado para almacenar datos en forma legible.



17 Bibliografía

17.1 Libros

- Project Management Institute Inc. (2008). *Guía de los Fundamentos para la Dirección de Proyectos*. Pennsylvania EE.UU.: PMI Book Service Center.
- Sommerville, I. (2011). *Ingeniería de Software* (9a ed.). Naucalpan de Juárez, Mexico: Pearson Education.
- Elmasri, R., & Navathe, S. B. (2007). *Fundamentos de Sistemas de Base de Datos*. Madrid: Pearson Education S.A.
- Silberschatz, A., Korth, H. F., & Sudarshan, S. (2002). *Fundamentos de Base de Datos* (4 ed.). Aravaca Madrid: McGraw Hill Interamericana de España, S.A. U.
- Rob, P., & Coronel, C. (2004). *Sistema de Base de Datos Diseño, Implementación y Administración* (5 ed.). CI, Mexico: International Thomson Editores S.A. de C.V.
- E. Kendal, K., & E. Kendal, J. (2005). *Análisis y Diseño de Sistemas* (6 ed.). Mexico: Perason Educación.
- Hernandez, C. R. (1991). *Metodología de la Investigación*. Naucalpan de Juárez: McGraw Hill Interamericana de México, S.A. de C.V.
- Rodríguez Monguel, E. A. (3003). *Metodología de la Investigación* (5 ed.). Mexico: Colección Héctor Merino Rodriguez.
- Larman, C. (1999). *UML y Patrones, Introduccion al Análisis y Diseño Orientado a Objetos* (1a ed.). Pearson Education.
- Pressman, R. S. (2005). *Ingeniería del Software, Un Enfoque Practico* (6a ed.). México: McGraw-Hill.

17.2 Páginas Web

- Ministerio de Salud. (2012). *Manual de Organización y Funciones de la Gerencia General de Operaciones. Unidad de Conservación y Mantenimiento*. Obtenido de http://asp.salud.gob.sv/regulacion/pdf/manual/manual_organizacion_y_funciones_gerencia_general.pdf



- Ministerio de Salud. (2012). *Manual de Organización y Funciones de las RIIS*. Obtenido de http://asp.salud.gob.sv/regulacion/pdf/manual/Manual_de_Organizacion_y_Funciones_de_las_RIIS.pdf
- Awad, M. (2005). *A comparison between Agile and Traditional Software Development Methodologies*. Obtenido de http://pds10.egloos.com/pds/200808/13/85/A_comparison_between_Agile_and_Traditional_SW_development_methodologies.pdf
- CMS Information Technology. (2008). *Selecting Development Approach*. Obtenido de Center for Medicare & Medicaid Services: <https://www.cms.gov/Research-Statistics-Data-and-Systems/CMS-Information-Technology/XLC/Downloads/SelectingDevelopmentApproach.pdf>
- Faidani, H. (2011). *A Guide to Selecting SW Development Methodologies*. Obtenido de GT ISLIG: http://www.gtislig.org/HamidFaridani_GuideToSelectingSWMethodologies_SOC_PDD_2011_0305.pdf
- Faridani, H. (Marzo de 2011). *A Guide to Selecting SW Development Methodologies*. Recuperado el 26 de Marzo de 2013, de GT ISLIG: http://www.gtislig.org/HamidFaridani_GuideToSelectingSWMethodologies_SOC_PDD_2011_0305.pdf
- Baker, B. (2001). *Business Modeling with UML: The Light at the End of the Tunnel*. Recuperado el 1 de Abril de 2013, de IBM: <http://www.ibm.com/developerworks/rational/library/content/RationalEdge/dec01/BusinessModelingwithUMLDec01.pdf>
- Fundación Wikipedia, Inc. (12 de marzo de 2013). *Mapeo objeto-relacional*. Obtenido de Wikipedia: http://es.wikipedia.org/wiki/Mapeo_objeto-relacional
- Fundación Wikipedia, Inc. (11 de abril de 2013). *PHP*. Obtenido de Wikipedia: <http://es.wikipedia.org/wiki/PHP>
- Fundación Wikipedia, Inc. (3 de abril de 2013). *Programación Orientada a Objetos*. Obtenido de Wikipedia: http://es.wikipedia.org/wiki/Programaci%C3%B3n_orientada_a_objetos
- Fundación Wikipedia, Inc. (12 de marzo de 2013). *Symfony*. Obtenido de Wikipedia: <http://es.wikipedia.org/wiki/Symfony>
- Ministerio de Salud. (4 de marzo de 2013). *Desarrollo web Symfony2 parte 2*. Obtenido de wiki de la DTIC - MINSAL: http://wiki.salud.gob.sv/wiki/Desarrollo_web_Symfony2_parte_2



- Purcell, J. (s.f.). *Comparison of Software Development Lifecycle Methodologies*. Obtenido de Global Information Assurance Certification: <http://www.giac.org/cissp-papers/217.pdf>
- Universidad de Oriente - Venezuela. (11 de julio de 2012). *Metodologías para el desarrollo de Software*. Obtenido de Manogas Universidad de Oriente: http://wiki.monagas.udo.edu.ve/index.php/Metodolog%C3%ADas_para_el_desarrollo_de_software
- ABDALA, J. A. (s.f.). *APUNTES DE PLANEACIÓN DEL M.I.* Recuperado el 25 de Marzo de 2013, de Facultad de Ingeniería UNAM: http://www.ingenieria.unam.mx/~jkuri/Apunt_Planeacion_internet/TEMAII.5.pdf

17.3 Otros Documentos.

- Salud, M. d. (2011). *Manual de Procedimientos de la Unidad de Conservación y Mantenimiento*. San Salvador.
- Social, M. d. (2006). *Manual de Organización y Funciones de la Región de Salud - Manual de Organización y Funciones del Sistema Básico de Salud Integral*. San Salvador.
- Ministerio de Salud. (1998). *Manual de Procedimientos Estandarizados*. San Salvador.
- Ministerio de Salud Pública y Asistencia Social. (2006). *Manual de Organización y Funciones de la Región de Salud*. San Salvador.
- Ministerio de Salud y Asistencia Social. (14 de Junio de 1996). Reglamento General de Hospitales del Ministerio de Salud Pública y Asistencia Social. *Diario Oficial N. 110*.
- Molina, A., A. A., Ramirez, O., R. E., Orellana, Z., & de los Angeles, G. (2008). Propuesta de un sistema de mantenimiento para el Hospital Rosales. Tesis Ingeniería Industrial, Universidad de El Salvador.
- Asamblea Legislativa de la República de El Salvador. (1 de mayo de 1988). Código de Salud. *Diario Oficial N. 86*.
- IEEE 729-1983. (1982). *Glossary of Software Engineering Terminology*.