

UNIVERSIDAD DE EL SALVADOR

FACULTAD DE INGENIERIA Y ARQUITECTURA

ESCUELA DE INGENIERIA DE SISTEMAS INFORMATICOS



**SISTEMA DE INFORMACIÓN PARA EL CENSO Y EVALUACIÓN
DE EDIFICIOS EN LA FACULTAD DE INGENIERÍA Y
ARQUITECTURA DE LA UNIVERSIDAD DE EL SALVADOR
APLICANDO TECNOLOGÍAS GIS**

PRESENTADO POR:

JOSÉ GIOVANNI CRUZ CORDERO

OSCAR RAÚL PLEITEZ TRUJILLO

STEFHANIE JESZERELITA RIZO RODRÍGUEZ

PARA OPTAR AL TITULO DE:

INGENIERO DE SISTEMAS INFORMATICOS

CIUDAD UNIVERSITARIA, AGOSTO DE 2018

UNIVERSIDAD DE EL SALVADOR

RECTOR:

MSC. ROGER ARMANDO ARIAS ALVARADO

SECRETARIO GENERAL:

MSc. CRISTOBAL HERNAN RIOS BENITEZ

FACULTAD DE INGENIERIA Y ARQUITECTURA

DECANO:

ING. FRANCISCO ANTONIO ALARCON SANDOVAL

SECRETARIO:

ING. JULIO ALBERTO PORTILLO

ESCUELA DE INGENIERIA SISTEMAS INFORMATICOS

DIRECTOR:

ING. JOSE MARIA SANCHEZ CORNEJO

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

Trabajo de Graduación previo a la opción al Grado de:
INGENIERO DE SISTEMAS INFORMATICOS

Título:

**SISTEMA DE INFORMACIÓN PARA EL CENSO Y EVALUACIÓN
DE EDIFICIOS EN LA FACULTAD DE INGENIERÍA Y
ARQUITECTURA DE LA UNIVERSIDAD DE EL SALVADOR
APLICANDO TECNOLOGÍAS GIS**

Presentado por:

JOSÉ GIOVANNI CRUZ CORDERO

OSCAR RAÚL PLEITEZ TRUJILLO

STEFHANIE JESZERELITA RIZO RODRÍGUEZ

Trabajo de Graduación Aprobado por:

Docente Asesor:

ING. ARNOLDO INOCENCIO RIVAS MOLINA

SAN SALVADOR, AGOSTO DE 2018

Trabajo de Graduación Aprobado por:

Docente Asesor:

ING. ARNOLDO INOCENCIO RIVAS MOLINA

AGRADECIMIENTOS

Agradezco a mi familia por el apoyo desde el inicio de mi formación, y en general a lo largo de mi vida. Gracias a Ph.D. Edgar Armando Peña Figueroa por su entrega en el desarrollo del proyecto. De igual manera un agradecimiento especial a Alejandra Rivera Paz, por su acompañamiento y apoyo en mi crecimiento personal y profesional.

Giovanni Cruz

Agradezco el apoyo y esfuerzo de mis padres a lo largo de mi carrera. Gracias a Stefhanie por ayudarme a salir adelante cuando quise dejar todo de lado.

Raúl Pleitez

Le agradezco a Dios por haberme acompañado y guiado a lo largo de mi vida y carrera. Agradezco a mis padres Francisco y Guadalupe por su amor, esfuerzo, dedicación y apoyo incondicional a lo largo de mi carrera, por confiar y creer en mí. Gracias a mi madre por acompañarme en todo momento y llenar mi vida con la sabiduría de sus consejos e impulsarme a perseguir mis sueños jamás encontraría las palabras para expresar lo mucho que Te Amo. Agradezco a Raúl por motivarme a ser mejor cada día.

Stefhanie Rizo

Contenido

Introducción	9
Objetivos.....	10
Objetivo General	10
Objetivos Específicos	10
Alcances	11
Limitaciones.....	12
Importancia.....	13
Justificación	14
CAPÍTULO I: Estudio preliminar.....	16
1.1 Marco teórico.....	16
1.1.1 Sismología e impacto en el país.....	16
1.1.2 Evaluaciones post-sísmicas	16
1.1.3 Dificultades más comunes en las evaluaciones de daños	17
1.1.4 Fases de la evaluación de daños.....	19
1.1.5 Sistemas de información geográfica.	20
1.2 Antecedentes	21
1.2.1 Evaluación de daños producidos por el sismo de 1965.....	22
1.2.2 Evaluación de daños por el sismo de 1986	22
1.2.3 Evaluación de daños por los sismos del 2001	23
1.3 Planteamiento del problema	24
1.3.1 Diagnóstico del problema.....	25
1.4 Determinación de factibilidades	30
1.4.1 Factibilidad técnica.....	30
1.4.2 Factibilidad económica.....	33
1.4.3 Factibilidad operativa	37
1.4.4 Resumen de factibilidades	39
1.5 Resultados esperados.....	39
1.6 Metodología de desarrollo	40
1.6.1 Análisis	41
1.6.2 Formulación del problema.....	45
1.6.3 Diseño.....	46
1.6.4 Codificación	49

1.6.5 Pruebas	52
1.6.6 Documentación	53
1.6.7 Implementación.....	53
CAPÍTULO II: Situación Actual.....	54
2.1 Objetivo de la Situación Actual	54
2.2 Descripción de la Situación Actual.....	54
2.2.1 Inspecciones presísmicas.	54
2.2.2 Inspecciones post sísmicas.....	62
2.3 Descripción con enfoque de sistemas de la situación actual.	67
2.4 Análisis y diagnóstico del problema.....	69
2.5 Conclusión de la situación actual.....	70
CAPÍTULO III: Determinación de Requerimientos.....	71
3.1 Requerimientos informáticos	71
3.1.1 Requerimientos Funcionales.....	71
3.1.2 Requerimientos No Funcionales	74
3.2 Requerimientos de Desarrollo	76
3.2.1 Tecnológicos.....	76
3.2.2 Legales	80
3.2.3 Recurso humano.....	81
3.3 Requerimientos operativos	82
3.3.1 Tecnológicos.....	82
3.3.2 Recurso humano.....	84
3.4 Solución propuesta.....	86
3.4.1 Enfoque de Sistemas de Sistema Propuesto	86
3.4.2 Modelos de Casos de Uso Propuesto	89
3.4.3 Modelo de Clases	103
CAPÍTULO IV: Diseño	105
4.1 Introducción	105
4.2 Diseño de Datos.....	105
4.2.1 Diagrama Entidad Relación.....	107
4.2.2 Modelo Físico de la Base de Datos.....	108
4.2.3 Diccionario de datos.....	109
4.3 Diseño Arquitectónico.....	114
4.4 Diseño de Interfaces.....	117

4.4.1 Diagrama de navegación vistas web	117
4.4.2 Diagrama de navegación vistas móviles	118
4.4.3 Vistas	119
CAPÍTULO V: Implementación.....	135
5.1 Tecnologías a utilizar.....	135
5.2 Definición de estándares	135
5.2.1 Programación.....	135
5.2.2 Base de datos	143
5.2.3 Reportes	144
5.2.4 Pantallas de Entradas	145
5.2.5 Pantallas de Salidas.....	146
5.3 Documentación durante el desarrollo	146
5.3.1 Documentación interna	146
5.3.2 Documentación externa	147
CAPÍTULO VI: Plan de implementación.....	149
6.1 Introducción.....	149
6.2 Objetivos	149
6.2.1 Objetivo general.....	149
6.2.2 Objetivos específicos	149
6.3 Marco Referencial del Plan de Implementación	150
6.3.1 Nombre del proyecto.....	150
6.3.2 Ubicación	150
6.3.3 Descripción del Proyecto.....	150
6.3.4 Recursos.....	150
6.4 Actividades	153
6.5 Administración de riesgos.....	154
6.6 Planes de contingencia	154
6.6.1 Objetivo.....	154
6.6.2 Descripción	154
6.6.1 Plan de Respaldo y restauración de copia de seguridad.....	154
6.6.2 Plan de contratación de nuevo personal	155
6.6.2 Plan de Gestión de pago de servidor.	155
Conclusiones	157
Recomendaciones	158

Glosario	159
Bibliografía.....	161
Anexos.....	162
Anexo A: Formulario CASALCO.....	162
Anexo B: Formulario ASIA.....	167
Anexo C. Visus.....	172
Anexo D: Cálculos de costos.....	174
Anexo E: Cálculo de beneficios	175
Anexo F: Software Libre	176
Anexo G: Ley de Fomento y Protección de la Propiedad Intelectual.....	178
Anexo H: JavaScript.....	178
Anexo I: MongoDB	181
Anexo J: Node.js	201
Apéndices.....	205
Apéndice A: Instrumentos de recolección de datos.	205
Apéndice B: Plan de capacitación para administradores.	213
Apéndice C: Plan de capacitación para evaluadores.....	215

Índice de tablas

Tabla 1: Análisis FODA.....	26
Tabla 2: Formulación de estrategias	27
Tabla 3: Requerimiento de recursos humanos.....	30
Tabla 4: Requerimiento de hardware.	31
Tabla 5: Características técnicas del hardware.....	32
Tabla 6: Requerimiento de software.....	33
Tabla 7: Costos de recursos de hardware.....	33
Tabla 8: Costos de recursos de software.	34
Tabla 9: Costos de recurso humano.	35
Tabla 10: Costos fijos mensuales.	35
Tabla 11: Costos totales del proyecto.	35
Tabla 12: Costo actual de una inspección.....	36
Tabla 13: Costo de una inspección con el uso del sistema.	36
Tabla 14: Requerimiento de hardware y software para el uso del sistema web.	38
Tabla 15: Requerimientos de hardware y software para el uso de la aplicación móvil.	38
Tabla 16: Requerimientos hardware y humano para la puesta en producción del sistema.....	39
Tabla 17: Análisis y diagnóstico del problema.....	69
Tabla 18: Requerimientos funcionales generales.....	71
Tabla 19: Requerimientos funcionales del módulo de inspección.	72
Tabla 20: Requerimientos funcionales de capacitaciones.....	72
Tabla 21: Requerimientos funcionales de inspecciones.....	73
Tabla 22: Requerimientos funcionales de mapas.....	73
Tabla 23: Requerimientos de hardware de desarrollo.....	76
Tabla 24: Descripción de roles y responsabilidades.....	81
Tabla 25: Descripción de los perfiles.....	82
Tabla 26: Descripción de los requerimientos operativos del sistema.....	82
Tabla 27: Requerimientos operativos de software para dispositivos móviles.	83
Tabla 28: Descripción de requerimientos operativos de software en el servidor.....	83
Tabla 29: Requerimientos operativos de hardware del cliente web.....	83
Tabla 30: Requerimientos operativos de hardware para dispositivos móviles.....	84
Tabla 31: Requerimientos operativos de hardware del servidor.....	84
Tabla 32: Requerimiento operativo de gestor de base de datos.....	84
Tabla 33: Requerimientos operativos de servidor web.....	84

Tabla 34: Requerimientos operativos de recurso humano.	85
Tabla 35: Descripción de nomenclatura de casos de uso.	89
Tabla 36: Descripción de los actores del sistema.....	91
Tabla 37: Descripción de caso de uso para la gestión de módulos de inspección.....	93
Tabla 38: Descripción de caso de uso para la gestionar la programación de inspecciones.....	94
Tabla 39: Descripción de caso de uso para la gestión de instituciones.	95
Tabla 40: Descripción de caso de uso para la gestión de catálogos de capacitaciones.	96
Tabla 41: Descripción de caso de uso para gestionar el perfil de las edificaciones.....	97
Tabla 42: Descripción de caso de uso para la gestión de usuarios.	98
Tabla 43: Descripción de caso de uso para la gestión de capacitaciones recibidas.	99
Tabla 44: Descripción de caso de uso para la gestión de alertas y notificaicones.....	100
Tabla 45: Descripción de caso de uso para realizar una inspección.	101
Tabla 46: Descripción de caso de uso para la visualización de inspecciones realizadas.	101
Tabla 47: Descripción de caso de uso para visualizar detalles de la inspección.	102
Tabla 48: Descripción de caso de uso para visualización del mapa.	102
Tabla 49: Descripción de caso de uso para visualizar el histórico de una edificación.	102
Tabla 50: Descripción de caso de uso para visualizar notificaciones y alertas.	103
Tabla 51: Costos iniciales de recursos informáticos para la implementación del sistema.....	150
Tabla 52 Costos fijos de recursos informáticos para la implementación del sistema.....	151
Tabla 53: Costos iniciales de recurso humano para la implementación del sistema.....	152
Tabla 54 Costos fijos de recurso humano para la implementación del sistema.	152
Tabla 55: Costos inicial total de la implementación del sistema.	152
Tabla 56: Costos fijos totales de la implementación del sistema.	152
Tabla 57: Actividades a desarrollar para la implementación del sistema.	154
Tabla 58: Lista de riesgos contemplados en la implementación del sistema.	154
Tabla 59: Plan de contingencia de respaldo y restauración de copia de seguridad.....	155
Tabla 60: Descripción del plan de contingencia de contratación de nuevo personal.	155
Tabla 61: Descripción del plan de contingencia de gestión de pago de servidor.	156

Índice de ilustraciones

Ilustración 1. Características de sobreposición de un SIG.	21
Ilustración 2. Diagrama de Ishikawa.....	28
Ilustración 3. Ciclo de desarrollo en cascada.	41
Ilustración 4. Narrativa de caso de uso.	44
Ilustración 5. Organización de grupos de inspección.	54
Ilustración 6. Representación de subregiones a evaluar.	57
Ilustración 7. Organización durante evaluaciones.	58
Ilustración 8. Flujo de trabajo.	60
Ilustración 9. Diagrama de sistema, situación actual.....	67
Ilustración 10. Diagrama de sistema, propuesta.....	86
Ilustración 11. Diagrama de casos de uso.....	90
Ilustración 12. Modelo de clases.	104
Ilustración 13. Diagrama entidad relación.	107
Ilustración 14. Modelo físico de base de datos.....	108
Ilustración 15. Arquitectura física.	114
Ilustración 16. Arquitectura lógica.	116
Ilustración 17. Diagrama de navegación web.....	117
Ilustración 18. Diagrama de navegación móvil.	118
Ilustración 19. Inicio de sesión, vista web.	119
Ilustración 20. Mapa, vista web.	119
Ilustración 21. Edificios, vista web.....	120
Ilustración 22. Usuarios, vista web.....	120
Ilustración 23. Instituciones, vista web.	121
Ilustración 24. Inspecciones, vista web.	121
Ilustración 25. Maquetar módulo, vista web.	122
Ilustración 26. Catalogo de capacitaciones, vista web.	122
Ilustración 27. Capacitaciones programadas, vista web.	123
Ilustración 28. Inspecciones programadas, vista web.	123
Ilustración 29. Inspecciones realizadas, vista web.	124
Ilustración 30. Detalles de inspección.	124
Ilustración 31. Gestión de notificaciones, vista web.	125
Ilustración 32. Panel de notificaciones, vista web.....	125
Ilustración 33. Estado de edificación, vista web.	126

Ilustración 34. Inicio de sesión, vista móvil.....	127
Ilustración 35. Mapa, vista móvil.	127
Ilustración 36. Gestión de usuarios, vista móvil.....	128
Ilustración 37. Nuevo usuario, vista móvil.	128
Ilustración 38. Nueva inspección, vista móvil.	129
Ilustración 39. Código QR, vista móvil.....	129
Ilustración 40. Realizar inspección, vista móvil.	130
Ilustración 41. Editor de polígonos, vista móvil.....	130
Ilustración 42. Entrada de imágenes, vista móvil.....	131
Ilustración 43. Histórico, vista móvil.	131
Ilustración 44. Reporte de inspecciones.....	132
Ilustración 45. Listado de capacitaciones.....	132
Ilustración 46. Listado de módulos.....	133
Ilustración 47. Listado de edificaciones.....	133
Ilustración 48. Usuarios registrados.....	134
Ilustración 49. Modelo reporte.....	145

Introducción

A continuación, se presentan los elementos desarrollados a lo largo de la ejecución del trabajo de graduación titulado “Sistema de información para el censo y evaluación de edificios en la facultad de ingeniería y arquitectura de la universidad de el salvador aplicando tecnologías gis”.

Inicialmente se realiza un estudio preliminar que sienta las bases para el planteamiento del problema y el diagnóstico del mismo. Se determinan las factibilidades económicas, operativas y técnicas. Y se detalla la metodología de desarrollo a emplear durante la construcción de la solución.

Se establece la situación actual del proceso de evaluación de edificios, se analiza y diagnostica el problema llegando a la conclusión de implementar una herramienta de software cuyo desarrollo detalla este proyecto.

Posteriormente se procede a la determinación de requerimientos, estableciendo así los requerimientos informáticos, operativos y de desarrollo, haciendo uso de herramientas de recolección de datos, como lo son documentación y reuniones con la contraparte técnica; brindando así una solución propuesta.

Luego se realiza el diseño de la solución, y se lleva a cabo el desarrollo del sistema informático haciendo uso de estándares de desarrollo que brindan un código legible, mantenible y escalable; garantizando así una solución libre de errores y de calidad.

Se proporciona un apartado que describe la documentación de uso del sistema, así como documentación para su extensibilidad por parte de usuarios técnicos. Se proporciona además un plan de implementación para su puesta en marcha en producción.

Finalmente se brindan conclusiones generadas en la realización del proyecto, recomendaciones y un glosario y anexo de documentos e información de apoyo al contenido presentado.

Objetivos

Objetivo General

Desarrollar un sistema informático para el censo y evaluación de edificios en la Facultad de Ingeniería y Arquitectura de la Universidad de El Salvador, aplicando tecnologías GIS, el cual servirá como apoyo a las jornadas de inspección que realizan evaluadores del área de Ingeniería Civil.

Objetivos Específicos

- Realizar una investigación documental del proceso para la realización de inspecciones pre-sísmicas y post-sísmicas.
- Describir la situación actual del sistema aplicando el enfoque de sistemas para describir los elementos y sus interrelaciones.
- Identificar las causas de la problemática actual y plantear una solución óptima.
- Determinar y validar los requerimientos funcionales y no funcionales para el desarrollo del sistema.
- Realizar el análisis, diseño y programación de un sistema informático que apoye en las jornadas de inspección que realizan evaluadores del área de Ingeniería Civil.
- Elaborar el plan de implementación y la documentación necesaria para poner en marcha el sistema propuesto.

Alcances

- El sistema apoyará la realización de inspecciones para la evaluación de edificaciones, utilizando dispositivos móviles para la recolección de los datos.
- El sistema permitirá administrar la información recolectada de las inspecciones realizadas.
- El sistema permitirá ingresar diferentes módulos de evaluación mediante un constructor de formularios el cual proveerá diferentes tipos de entradas tanto manuales como calculadas.
- El sistema permitirá configurar valores calculados por medio de un sistema de nodos.
- El sistema permitirá registrar las edificaciones las cuales serán georreferenciadas y categorizadas, para ser inspeccionadas posteriormente.
- El sistema permitirá administrar el recurso humano que haya sido capacitado (Evaluador).
- El sistema permitirá coordinar y programar inspecciones periódicas y en caso de emergencia por la ocurrencia de un desastre natural; la coordinación y programación de inspecciones se llevará a cabo por medio de un sistema de notificaciones y avisos integrado en el portal web y una aplicación móvil.
- La información recolectada por medio de las inspecciones permitirá orientar esfuerzos para realizar inspecciones detalladas con módulos de evaluación específicos y determinar el estado del edificio.
- En caso de que existan problemas con las redes de comunicación en el momento de una inspección, los dispositivos móviles, que servirán para recolectar datos, podrán pausar una inspección y realizar sincronizaciones con el resto del sistema cuando las redes de comunicación vuelvan a reestablecerse.
- En el momento de una inspección el sistema mostrará por medio de indicadores visuales cuales son las edificaciones que no han sido inspeccionadas y de igual manera mostrará las edificaciones que se están evaluando actualmente.
- La información recolectada a través del sistema será puesta a disposición por medio de una API por lo tanto nuestro sistema se podrá integrar con sistemas de monitoreo existentes en otras instituciones.
- La Escuela de Ingeniería Civil será la entidad que proporcionará información para el desarrollo del sistema.
- El sistema generará mapas donde podrá visualizarse las edificaciones y su estado, además mostrará información sobre ellas y el histórico de las inspecciones.

Limitaciones

Las siguientes limitaciones se tomarán en cuenta para el correcto desarrollo del Sistema Informático:

- Todos los módulos de inspección que se ingresen al sistema deberán de tener una lógica de generación de banderas en base a dos índices primarios.
- La clasificación de edificaciones se basa en banderas de tres diferentes colores: verde, amarillo y rojo.
- Se debe contar con conexión a internet para poder ubicar una edificación en el mapa e ingresarla con la aplicación móvil. A partir de ello, puede llevarse a cabo la inspección sin requerir de conexión de datos, y requerirla solamente cuando se desee subir al servidor la inspección realizada.
- Para evitar el gasto de la cuota de datos del móvil, las fotografías relacionadas a cada inspección se subirán de manera asíncrona con respecto a la inspección realizada.

Importancia

En el área de construcción existen diversos estándares y normas que rigen los procedimientos, materiales y técnicas a utilizar para garantizar la calidad y seguridad de las construcciones los cuales se han desarrollado con base en la experiencia. En nuestro país se encuentran muchos edificios que no cumplen con dichos estándares debido al tiempo en que se construyeron, y a pesar de ello siguen en funcionamiento lo cual representa un riesgo tanto para las personas que utilizan dichos edificios como las edificaciones aledañas.

Teniendo en cuenta los desastres naturales, sismos y el paso del tiempo es de suma importancia conocer cuales edificaciones presentan daños que pongan en peligro las vidas humanas.

Con el desarrollo del proyecto se obtendrán los siguientes beneficios:

- Apoyo en la coordinación de recurso humano para las jornadas de evaluación de edificios: El sistema almacenará los usuarios capacitados para la evaluación de edificios, además permitirá el envío de notificaciones en caso de sismo o eventos de capacitación.
- Facilitar el proceso de evaluación de edificios: Actualmente el proceso de evaluación de edificaciones se vuelve engorroso por el número de ítems que contiene el formulario utilizado, con la implementación del sistema dichos ítems serán clasificados previamente mostrando sólo los que corresponden al tipo de estructura o técnica utilizada.
- Inventario de estado actual de edificaciones: Conocer el estado en que se encuentran las edificaciones actualmente para hacer llegar la información a las entidades correspondientes y tomar las medidas necesarias en caso de que representen un peligro.
- Priorización de edificios con base en evaluación previa: Conociendo el estado en que se encuentran las edificaciones, se dará prioridad de evaluación en caso de sismos a todas aquellas que por su uso o estado de daño representen mayor peligro para la población.
- Centralización de información sobre edificaciones: la información recolectada en las jornadas previas y post-sísmicas será almacenada en la Universidad El Salvador, estando a disposición de la entidad central para que ésta pueda gestionarlo de la manera que legalmente definan.

Justificación

Debido a la alta actividad sísmica del país, desastres naturales, técnicas de construcción utilizadas en edificios antiguos y el paso del tiempo, conocer el estado de las edificaciones es imperativo ya que de esta manera se pueden tomar las medidas necesarias dependiendo del nivel de riesgo que éstas representen hacia la población.

Actualmente el país no cuenta con un registro de las edificaciones dañadas durante el sismo de octubre de 1986 y los sismos de enero y febrero de 2001. Se crearon de emergencia las comisiones de evaluación de edificaciones MOP-ASIA-CASALCO y MOP-ASIA-FESIARA, sin embargo, la poca organización provocó que el trabajo realizado en dichas evaluaciones no tuviera seguimiento. Para las evaluaciones realizadas por dichas comisiones se creó un formulario para la evaluación post-sísmica de las edificaciones, el cual brinda una clasificación basándose en los daños que presentan las estructuras y de esta manera determinar si es seguro habitar o usar dicha edificación, sin embargo, la información recolectada durante la evaluación post sísmica solamente queda archivada, sin que se tomen las medidas correspondientes para cada caso, permitiendo que edificaciones que deberían ser demolidas o restauradas representen un peligro para las vidas de los habitantes o usuarios de los edificios. El inventario de daños entregado por el comité antes mencionado al MOP después del sismo de 1986 no está disponible, y repetidos esfuerzos por obtener copias de los documentos del MOP han sido fallidos. La explicación dada por ahora es que los documentos han sido perdidos.

Es por tal motivo que nace la necesidad de crear un sistema que permita: construir múltiples estructuras de inspección de edificios y utilizarlas para la evaluación de los mismos, organizar personal capacitado durante una emergencia o previo a ellas y centralizar los datos recolectados para su procesamiento y fácil acceso a la información. Basado en ello, tomar las medidas necesarias para salvaguardar a la población. Además, el análisis sobre los datos sirva como insumo para la creación de nuevas normas o para implementar mejoras al Reglamento para la Seguridad Estructural de las Construcciones¹.

Tomando como punto de partida la Facultad de Ingeniería y Arquitectura de la Universidad El Salvador la cual cuenta con diversas edificaciones que han sufrido daños, ya sea por el paso del tiempo, desastres naturales y los sismos ocurridos durante su vida útil, el sistema planteado brindará a los ingenieros civiles la plataforma para el registro del estado actual de las edificaciones con el apoyo de fotografías y el ingreso de la información referente a estos; también permitirá la evaluación de las edificaciones después de la ocurrencia de un sismo; así como la posibilidad de crear cualquier tipo de estructura de inspección.

Con la utilización del sistema los tiempos para la realización de la evaluación de edificios se verán reducidos, hasta en un 33% del tiempo necesario si no se utilizara el sistema², incidiendo así en el número de evaluaciones realizadas por persona permitiendo un mayor alcance por cada jornada o durante una crisis sísmica, siendo esto de especial importancia debido a que durante el tiempo de inspección pueden originarse réplicas que fuerzan la reevaluación de edificios, debido a que su nivel de seguridad ya habrá cambiado, y es por ello que el número de inspecciones puede acrecentarse y

¹ D.E N° 105, del 23 de octubre de 1996, publicado en el D.O. N° 204, Tomo 333, del 30 de octubre de 1996.

² Ver Anexo D: Cálculo de beneficios

requerir de muchos evaluadores; siendo así que la gestión de toda esta información, que podría repetirse múltiples veces, debe ser eficiente y sistematizada.

El sistema permitirá recolectar la información de las edificaciones a través de formularios previamente construidos; la cual será almacenada en la base de datos permitiendo que el acceso a la información recolectada sea rápida y segura desde una plataforma web para su análisis posterior, también se prevee que el sistema mostrará el histórico de evaluaciones y brinde la calificación de los edificios con base en los datos ingresados por los usuarios, y poder así determinar las acciones a tomar y salvaguardar las vidas de los estudiantes y personal que labora en la Facultad de Ingeniería y Arquitectura.

CAPÍTULO I: Estudio preliminar

1.1 Marco teórico

1.1.1 Sismología e impacto en el país

Los sismos se manifiestan como vibraciones en la corteza terrestre que pueden ser clasificados como entre-placas e intra-placas. Los primeros son originados en los límites de las placas tectónicas como resultado de su movimiento relativo, un ejemplo de este tipo de sismos son los que se originan en el sistema de fallas de Guatemala. Los segundos son los que ocurren lejos de los límites de las mismas y probablemente se deban a una reducción local de la resistencia del material de la litosfera que a un aumento del esfuerzo, siendo un ejemplo los sismos que ocurren en la Depresión de Honduras (Bommer, Salazar y Samayoa, 1998).

El concepto o parámetro que actualmente se utiliza para cuantificar el nivel de actividad sísmica de una región, a través del registro de los sismos en el espacio y en el tiempo, es decir que se determina al identificar donde ocurren los sismos, qué magnitud tienen y con qué frecuencia ocurren, se conoce como sismicidad.

El Salvador es un país con una alta sismicidad, afectado principalmente por 5 fuentes generadoras de sismos:

1. La fosa de subducción, localizada a unos 125 km. de la costa, que se forma por la interacción de la placa de Cocos que se introduce debajo de la placa del Caribe.
2. La cadena volcánica en Centro América la cual corre paralelamente a la fosa de subducción, debido a que es la subducción la que produce el magma que ha dado origen a esta cadena y es por ello que corre paralela a ésta con una longitud total de aproximadamente 1,060 km.
3. La frontera entre la placa del Caribe y la de Norte América.
4. La depresión de Honduras, la cual contiene pequeños segmentos de fallas normales.
5. La zona de extensión tectónica que está limitada al suroeste por la cadena volcánica, por el sistema de fallas de Guatemala al norte y por la depresión de Honduras al este.

La alta sismicidad y crecimiento desordenado, genera un grave problema desde el punto de vista de seguridad para las personas que habitan las edificaciones que son construidas en lugares y con tecnologías que son altamente vulnerables a los eventos sísmicos.

1.1.2 Evaluaciones post-sísmicas

Ante la alta sismicidad de El Salvador es importante que las entidades de atención de emergencia tengan la capacidad de realizar evaluaciones post-sísmicas en las primeras horas después de que ocurra un sismo, con el fin de tener la información esencial para tomar decisiones a corto plazo, determinar la extensión geográfica de las zona afectadas, definir el grado relativo de daño y el tipo de

infraestructura involucrada, así como generar dictámenes de habitabilidad de los edificios para evitar la pérdida de vidas humanas.

Además, la existencia de un censo de daños post-sísmicos permite actualizar las normativas de construcción de edificios analizando los daños típicos, los cuales son aquellos daños que presentan un patrón definido y que se repiten en numerosas situaciones. Los daños típicos pueden catalogarse como leves, moderados, graves o severos, dependiendo de la pérdida de resistencia o estabilidad que experimente el elemento en el que se dé dicho daño.

Las inspecciones de edificios posteriores a los sismos son la mejor manera para identificar los daños típicos y establecer las causas principales de los mismos. Por ejemplo, en base a observaciones de sismos pasados, se pueden establecer tres causas principales de daños típicos en edificios:

1. Daños por configuración estructural inadecuada.
2. Daños por defectos de resistencia, rigidez y ductilidad.
3. Daños por defectos de diseño.

Cada nuevo episodio sísmico tiene sus propias características y genera nuevos datos de daños en las edificaciones que dejan al descubierto otros daños típicos no contemplados anteriormente.

1.1.3 Dificultades más comunes en las evaluaciones de daños

Con base en la experiencia y conocimiento de las evaluaciones de daños realizadas en el mundo y las experiencias basadas en la aplicación de diferentes procedimientos, se han observado algunas dificultades en los procesos de evaluación post-sísmica, que son similares en diferentes países.

Los más relevantes en términos generales son los siguientes:

1.1.3.1 Falta de entrenamiento

Para llevar a cabo un buen proceso de evaluación de daños, es importante contar con evaluadores con amplia experiencia. Sin embargo, cuando ocurre un sismo de gran magnitud, los daños en la zona pueden ser tan generalizados, que no es posible que los expertos se encarguen de hacer la totalidad de las evaluaciones. Este problema hace necesario que gran parte de las evaluaciones sean realizadas por profesionales con poca o ninguna experiencia, que posiblemente no están familiarizados con los daños causados por movimientos sísmicos.

Usualmente, para los neófitos, el impacto al ver los daños es tan grande que tienden a calificarlos de manera más grave de lo que realmente son, y en contraste, en muchas ocasiones subestiman casos graves que aparentemente no lo son. A pesar de que los métodos de evaluación existentes cuentan con buenas descripciones para los diferentes niveles de daño que utilizan, cuando no existe una buena preparación previa, la tendencia de los evaluadores inexpertos de agravar o subestimar el nivel de daño es una constante.

Sin duda, la información que interviene en la evaluación es altamente subjetiva y depende de la concepción y la impresión que tenga el evaluador en cada caso. Es posible que por la inexperiencia de los evaluadores se cometan errores como demoler edificios que probablemente no se encontraban en condiciones tan graves o que se evacuen edificios sin necesidad, lo que sería especialmente grave en el caso de edificios indispensables. También es posible que se pasen por alto daños en un edificio que comprometan su estabilidad, poniendo así en peligro la vida de sus ocupantes. Por eso es tan importante garantizar que realice las evaluaciones detalladas el personal más experimentado y con un mejor conocimiento sobre el comportamiento de las estructuras y patología de edificios.

1.1.3.2 Falta de cualificación de los evaluadores.

De lo anterior se concluye que es fundamental que se realice un entrenamiento previo de los profesionales que van a participar en el proceso, así como la selección de un grupo más especializado para la toma de las decisiones más difíciles. Se recomienda que en una fase posterior a este trabajo se realice una guía o manual detallado para unificar los criterios de evaluación de daños y se desarrollen unos módulos para el entrenamiento de los futuros evaluadores.

El entrenamiento debe desarrollarse en los siguientes aspectos: procedimiento de movilización, información y ayudas sobre cómo ubicarse en el terreno y cómo manejar la nomenclatura y la información catastral de los inmuebles, organización de las comisiones, uso de los formularios, procedimiento de informe, determinación en el sitio del sistema estructural, evaluación de la calidad de los materiales, evaluación del daño estructural y arquitectónico, identificación del peligro que presentan los elementos no estructurales y los edificios adyacentes, clasificación de los daños y de definición de la ocupación temporal del edificio. Para tal fin se recomienda organizar programas de entrenamiento continuo a través de convenios con las asociaciones de ingenieros y arquitectos, universidades, etc.

1.1.3.3 Subjetividad en las evaluaciones.

Los niveles de daño son definidos en la mayoría de los métodos de evaluación mediante calificaciones lingüísticas como leve, menor, moderado, medio, severo, grave o fuerte, conceptos que pueden tener una notable variación en su significado según la persona y experiencia de quien los utilice. Por esta razón se puede decir que no existe un límite claramente definido entre estas valoraciones. Lo que para una persona es moderado, para otra puede ser severo, así como puede estar en medio de los dos conceptos para otra. Por esto es necesario intentar unificar el sentido de estos conceptos y volver la evaluación lo más cuantitativa posible, determinando porcentaje de elementos afectados, tamaño y tipo de grietas.

1.1.3.4 Problemas en la ubicación de los edificios.

Otro problema muy común es la falta de estandarización de las direcciones, lo que genera no sólo que se repitan en muchas ocasiones las visitas, sino que también dificulta su correlación con las fichas catastrales o su ubicación sobre un mapa.

1.1.4 Fases de la evaluación de daños.

Las metodologías de evaluación están divididas en dos etapas, una conocida como evaluación de emergencia y otra a la que llamaremos como evaluación detallada. La evaluación de emergencia del nivel de daño de la edificación tiene como finalidad, identificar en el menor tiempo posible las edificaciones o las zonas de éstas, que de manera visual se muestran seguras (habitables), las inseguras (no habitables y en peligro de colapsar) y de las que se tienen dudas (uso restringido), para determinar las edificaciones que requieren una evaluación más exhaustiva para evitar la pérdida de vidas humanas debido a réplicas.

En el caso de la evaluación detallada de las edificaciones, ésta tiene como finalidad evaluar de una manera visual, pero basada en criterios técnicos razonables, la seguridad de las edificaciones que en la evaluación de emergencia se determinó que debían evaluarse más detalladamente, para dar un dictamen sobre las medidas de protección a tomar posteriormente, las cuales pueden ir desde la reparación hasta considerar la posible demolición de la edificación.

Estas evaluaciones se realizan con instrumentos diferentes, teniendo secciones más específicas los de una evaluación detallada. En las evaluaciones se toman en cuenta criterios como la inestabilidad global de la estructura, los problemas geotécnicos, los daños en elementos estructurales y los no estructurales. Estos criterios tienen pesos según el nivel de gravedad que los mismos representan para la edificación.

Además, existe un manual donde se presenta una descripción detallada de cada ítem del instrumento, para evaluar y clasificar el nivel de daños en elementos estructurales, elementos no estructurales, etc. con el propósito de llegar a un dictamen, lo más objetivo posible.

Hay que aclarar que aunque los instrumentos para una evaluación de emergencia y una evaluación detallada son diferentes estos tienen coherencia entre sí, ya que no puede estarse dando duplicidad de datos con secciones repetidas, al punto que a veces se recomienda utilizar un mismo formulario, que incluya todas las secciones para ambas evaluaciones, pero en el caso que se realice una evaluación de emergencia llenar sólo las secciones pertinentes a esta y si posteriormente se hará una evaluación detallada se llenará sobre el mismo formulario el resto de secciones, esto surge a raíz de que en ocasiones se ha generado confusión al no relacionar el formulario de una evaluación de emergencia y con el de una evaluación detallada, lo cual genera desconfianza acerca de la capacidad de los equipos evaluadores e implica mayor desgaste a las personas que conforman estos equipos cuando almacenan los instrumentos.

En El Salvador no existe una metodología unificada para la permanente evaluación de edificaciones a raíz de fenómenos sísmicos, aunque podemos rescatar los esfuerzos de diferentes investigaciones destinadas a generarla, una de estas investigaciones es la “Metodología para la evaluación de daños en edificaciones post-sismo”, presentada por Carlos Vladimir Najarro Gálvez en el año 2008, en esta metodología el autor sugiere 3 fases se puede dar respuesta a un levantamiento de datos post-sísmicos ordenado y actualizado.

1.1.4.1 Fase I: Organización.

En esta fase se propone la formación del Comité Interinstitucional Permanente para la Evaluación de Edificaciones Dañadas por Sismos, el cual será la entidad encargada de toda la logística para organizar a los profesionales de la ingeniería civil y la arquitectura, administrar el material y equipo necesarios para realizar las evaluaciones.

1.1.4.2 Fase II: Obtención de datos en campo.

En esta fase se obtendrá toda la información necesaria para identificar, evaluar y clasificar cada una de las edificaciones que se inspeccionen. La evaluación de los daños de las edificaciones se desarrollará en dos etapas, la evaluación de emergencia y la evaluación detallada. Para realizar las evaluaciones se proponen la conformación de brigadas de inspección de campo y el uso del formulario único de inspección.

1.1.4.3 Fase III: Procesamiento de datos.

En esta fase se propone la creación de una base de datos, en la que se pueda almacenar y procesar los datos obtenidos en los levantamientos de campo, la información debe estar georreferenciada, para una mejor consulta. Esta tarea estará a cargo del Comité Interinstitucional Permanente para la Evaluación de Edificaciones Dañadas por Sismos.

1.1.5 Sistemas de información geográfica.

El concepto de sistema de información geográfica (También conocido con los acrónimos SIG en español o GIS en inglés, en el presente documento se referirá a este término como SIG) no es nuevo. Primero, fue conceptualmente aplicado para identificar cambios al hacer análisis simultáneo de mapas producidos en diferentes fechas sobre el mismo tema. El concepto de SIG estuvo también ya en uso, cuando mapas con diferentes tipos de información para una misma área, fueron superpuestos como transparencias para ubicar sus interrelaciones.

Lo que es nuevo, y progresa rápidamente, es la tecnología avanzada de las computadoras, que permite el examen frecuente de grandes áreas, a bajo costo y con una creciente cantidad de datos. La digitalización, manipulación de información, interpretación y reproducción de mapas, son pasos en la generación de un SIG que ahora se pueden dar rápidamente, casi en tiempo real.

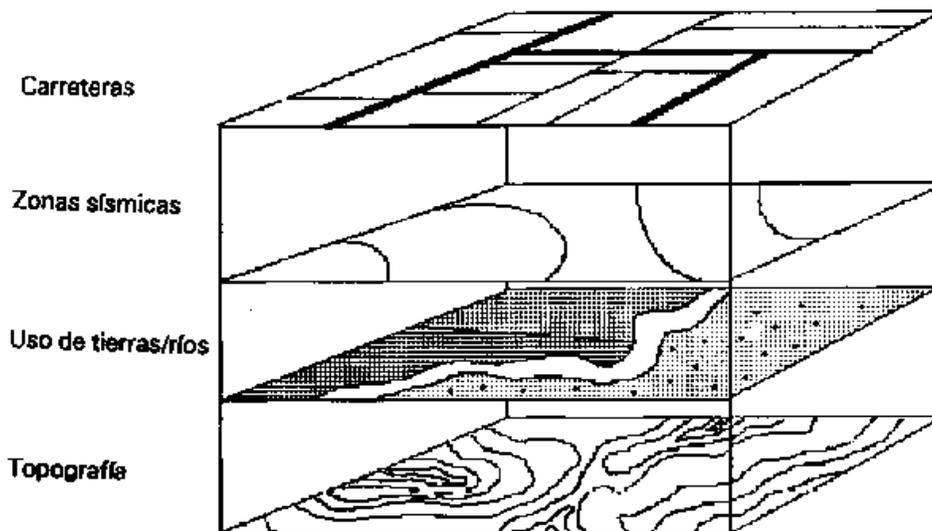


Ilustración 1. Características de superposición de un SIG.

Los datos manejados por un SIG en computadora son ordenados, sea por técnicas de "raster" o de vectores. El modelo "raster" utiliza un cuadrículado para referir y almacenar la información. Un área de estudio es dividida en pequeñas áreas o matriz de células cuadradas (a veces rectangulares) idénticas en tamaño, y la información -los atributos presentados con códigos numéricos- es almacenada en cada compartimento para cada estrato o atributo en la base de datos.

Un compartimento puede mostrar bien el rasgo dominante que se encuentra en esa unidad, o la distribución porcentual de todos los atributos que se encuentran en la misma unidad. Los sistemas basados en raster definen las relaciones espaciales entre variables más claramente que los basados en vectores, pero la inferior resolución por causa de la estructura celular reduce la exactitud espacial.

Los datos de vectores son una traducción más aproximada al mapa original. Estos sistemas refieren toda la información como puntos, rayas o polígonos y asignan un conjunto único de coordenadas X, Y a cada atributo. Generalmente, los programas de cómputo del sistema vector tienen capacidad para ampliar una pequeña porción del mapa y mostrar mayor detalle, o para reducir un área y mostrarla en el contexto regional. Los datos de vectores pueden ofrecer gran número de opciones posibles para una más fácil sobre posición de transparencias con estratos de datos.

El modelo de vectores presenta las áreas graficadas de manera más exacta que un sistema raster pero, porque cada estrato está definido de manera singular, es considerablemente más difícil analizar la información de diferentes estratos.

1.2 Antecedentes

A raíz de los fenómenos naturales como sismos, inundaciones o deterioro por falta de mantenimiento se vuelve necesario la evaluación del estado de las edificaciones, las cuales deben realizarse de manera rápida, eficiente y sencilla, con base a una metodología estandarizada, con el fin de determinar objetivamente las condiciones de habitabilidad o inseguridad de las edificaciones; además se vuelve

indispensable contar con profesionales de ingeniería sísmica, estructural y civil que sean capaces de identificar las razones por las que una edificación presenta daños ante las fuerzas de un sismo ya que en ello puede influir la calidad del diseño, materiales y construcción .

En el desarrollo de este proyecto nos centraremos en los sismos ya que nuestro país está ubicado cerca del borde oeste de la placa del Caribe la cual está en constante interacción con las placas Cocos, norteamericana, Sudamericana y Nazca. Este escenario tectónico da origen, al menos a cinco fuentes generadoras de sismos en nuestro país: la zona de subducción de la fosa mesoamericana, fallas asociadas a la cadena volcánica, las fallas de Guatemala, la depresión de Honduras y los volcanes; debido a todos estos fenómenos las edificaciones estarán sometidas a las excitaciones de sismo al menos una vez en su vida útil, además se estima que cada año en El Salvador se registran unos 5,500 sismos. Estadísticamente cada 12, 15 y máximo 20 años ocurre un sismo de gran magnitud.

A lo largo del tiempo se han realizado esfuerzos para ejecutar evaluaciones en las edificaciones dañadas por sismos de manera formal y con criterios técnicos, las cuales fueron en los siguientes años:

1.2.1 Evaluación de daños producidos por el sismo de 1965

La evaluación de los daños en edificios causados por el sismo de 1965 fue realizada por la UNESCO a través de una comisión encabezada por Rosenblueth, E (Rosenblueth, E, 1965) a petición del Gobierno Salvadoreño, no se sabe sobre los criterios que utilizaron para cuantificar los daños. En ese tiempo existieron deficiencias tales como:

1. El país no contaba con profesionales capacitados para evaluar los daños causados en edificaciones debido a sismos.
2. No se contaba con ninguna institución encargada de capacitar a los profesionales ni de llevar un registro de las edificaciones dañadas y el seguimiento del cumplimiento de los dictámenes.
3. No se contaba con una metodología para evaluar las edificaciones dañadas, ni con manuales o formularios.

1.2.2 Evaluación de daños por el sismo de 1986

Según información consultada se formó la comisión MOP-ASIA-CASALCO, que se encargó de coordinar las evaluaciones y que tenía su sede en ASIA. Antes de formar esta comisión existía una descoordinación completa ya que cada una de las instituciones realizaba sus propias evaluaciones con sus criterios y su formulario. Dentro de las deficiencias que se encontraron en este proceso están:

1. No existió un entrenamiento continuo y previo al sismo. Se realizaron 4 cursos de entrenamiento de emergencia y de manera improvisada a los profesionales (aproximadamente 100 por cada curso) que se ofrecieron como voluntarios, en un periodo de 2 semanas.
2. No existían formularios para hacer el levantamiento de daños y la evaluación de las edificaciones (el Dr. Héctor David Hernández Flores elaboró uno basado en sus conocimientos y experiencia).
3. No existía una metodología para realizar la evaluación (todo se realizó de manera improvisada).

4. Las evaluaciones se realizaron hasta dentro de 2 semanas (periodo en el cual se capacitó a los voluntarios).
5. No se permitió el ingreso de los evaluadores a establecimientos de uso público de propiedad privada.
6. No existió un procesamiento adecuado de la información, la cual se terminó extraviando.

Las observaciones positivas que se tienen de esta evaluación es que hubo continuidad de los profesionales que realizaron las evaluaciones, es decir que únicamente los que recibieron el entrenamiento realizaron las evaluaciones y no se permitió que profesionales que no habían recibido el entrenamiento participaran.

1.2.3 Evaluación de daños por los sismos del 2001

De acuerdo a la investigación realizada se formó la comisión MOP-ASIA-FESIARA que se encargó de coordinar las evaluaciones y que tenía su sede en ASIA. Dentro de las deficiencias que se encontraron en este proceso están:

1. No existió un entrenamiento continuo y previo al sismo. Se realizaron 3 cursos de entrenamiento de emergencia y de manera improvisada a los profesionales que se ofrecieron como voluntarios, en el periodo de 1 semana.
2. El formulario que se utilizó fue uno utilizado en una ciudad del sur de México y el problema que se presentó fue que en la capacitación no se explicó a los profesionales como rellenar dicho formulario, creando muchas dudas al momento de realizar las evaluaciones.
3. No existía una metodología para realizar la evaluación (todo se realizó de manera improvisada).
4. Las evaluaciones se realizaron hasta dentro de 1 semana (periodo en el cual se capacitó a los voluntarios).
5. No se permitió el ingreso de los evaluadores a establecimientos de uso público de propiedad privada.
6. No existió una continuidad de parte de los profesionales que fueron capacitados y se cometió el error de permitir que cualquier profesional (sin estar debidamente capacitado) realizará de manera particular las evaluaciones de las edificaciones.

Lo positivo de esta evaluación fue que se elaboró una base de datos de todos los edificios evaluados, de la que tiene copia ASIA, el VMVDU y OPAMSS, pero actualmente no existe una coordinación para actualizar la lista de las edificaciones dañadas que han sido reparadas o del seguimiento que se les dio ni se conoce el estado de dicha base de datos.

Debido a las experiencias de los sismos de 1965, 1986 y sismos de Enero y Febrero de 2001 mencionadas anteriormente, se ha identificado la necesidad de desarrollar una metodología estandarizada para evaluar el daño de las edificaciones afectadas. Ahora mismo los procedimientos para las evaluaciones post-sísmicas normalmente se aplican en dos niveles: la evaluación rápida (o de habitabilidad), que se basa en el nivel de riesgo o peligro que representa un edificio para la población, y el segundo nivel es la evaluación detallada, que describe el nivel de daño estructural donde se evalúa los elementos por los que está constituido la edificación como columnas, vigas, paredes o

combinaciones de los mismos; posterior a su evaluación las edificaciones son etiquetadas con colores: VERDE, AMARILLO y ROJO, dependiendo del daño observado. El color VERDE significa que la edificación ha sido inspeccionada y que su ocupación es permitida, el color AMARILLO significa que el uso de la edificación es restringido y el color ROJO que la edificación es insegura y que su ingreso u ocupación no es permitida.

En el terremoto del 2001, la Universidad de El Salvador resultó con varios edificios afectados, muchos de ellos ya deteriorados por sismos anteriores, el huracán Mitch y la guerra. Las facultades más dañadas fueron las de Humanidades, Ingeniería, Agronomía, Química y Farmacia.

Como resultado de la experiencia adquirida después de los terremotos del 2001 en 2008 CASALCO elabora con el apoyo del Dr. Héctor David Hernández Flores³ elabora el manual de Evaluación Post-Sísmica⁴ y el formulario de evaluación rápida⁵.

Actualmente se utiliza el Formulario de Evaluación de Edificaciones⁶ creado y aprobado por ASIA, la información obtenida de una evaluación es altamente subjetiva, dado que los niveles de daños están definidos con calificaciones lingüísticas como: leve, moderado, fuerte o severo, en términos de la metodología única establecida para El Salvador, que tienen una variación considerable en su significado dependiendo del profesional y del entrenamiento, experiencia y criterio que posea.

A lo largo del tiempo han surgido diversas iniciativas para evaluación de edificaciones una de ellas fue el proyecto piloto, llamado VISUS⁷, el cual con la ayuda de la UNESCO fue lanzado en agosto del 2013 e implementado en febrero de 2014 en la Facultad de Ingeniería y Arquitectura de la Universidad de El Salvador, en San Salvador. Su objetivo es promover un compromiso comunitario escolar en el proceso de toma de decisiones concernientes a la seguridad de escuelas en áreas de El Salvador propensas a desastres mediante la evaluación de instalaciones escolares y utilización de OpenStreetMap. El proyecto contribuye a establecer un inventario de data geoespacial de las escuelas del país.

1.3 Planteamiento del problema

Los desastres naturales como sismos, inundaciones, huracanes y falta de mantenimiento producen daños en las edificaciones por lo que se vuelve necesario un proceso de evaluación que permita identificar y determinar el estado de las edificaciones, estas evaluaciones deben ser dirigidas y realizadas por profesionales relacionados con el sector de la construcción de edificaciones, como ingenieros estructurales o civiles previamente capacitados en el área de ingeniería estructural.

Actualmente las edificaciones solo se evalúan en caso de emergencia, y si el daño en el área afectada es generalizado, los expertos locales en ingeniería estructural son insuficientes para hacer la totalidad

³ Evivienda.gob.sv. Evaluación post-sísmica de Edificios. Dr. Héctor Hernández Flores
http://www.evivienda.gob.sv/Emergencias/Descargas/MaterialCER/Evaluaci%C3%B3n_Post_S%C3%ADsmica.pdf

⁴ Evivienda.gob.sv Manual de Evaluación Post - Sísmica de Edificaciones de El Salvador abril 2008
http://www.evivienda.gob.sv/Emergencias/Descargas/MaterialCER/Manual/PARTE_1.pdf

⁵ Ver Anexo A: Formulario CASALCO

⁶ Ver Anexo B: Formulario ASIA

⁷ Ver Anexo C. Visus

de las evaluaciones, por lo que se hacen convocatorias donde se presentan voluntarios profesionales en Ingeniería Estructural y Civil sin verificar que estos últimos posean experiencia o estén capacitados en el área estructural, lo que puede causar que en los resultados haya sobreestimación o subestimación del daño, dado que se les hace difícil reconocer los daños estructurales, llevando así a la ocupación peligrosa o la demolición innecesaria de los edificios.

Para realizar las evaluaciones los profesionales examinan el exterior de la edificación, observan el suelo alrededor de la edificación, examinan la seguridad de elementos no estructurales y llenan el formulario de evaluación de edificios creado y aprobado por ASIA (Asociación Salvadoreña de Ingenieros y Arquitectos), donde especifican los datos de la edificación como el uso predominante, sistema estructural, año de construcción y posteriormente se detallan los daños en elementos arquitectónicos y estructurales. Cuando todos los ítems son llenados se procede a dar un estimado del porcentaje de daño de la edificación la cual es calculada por el evaluador ya que el formulario no posee pesos para cada ítem, esto permite que el dictamen del estado de una edificación sea subjetivo ya que depende del criterio y experiencia de cada evaluador.

Desafortunadamente los datos recolectados se guardan y la falta de interés en ellos y complejidad por la cantidad de evaluaciones realizadas no ha permitido que estos datos sean procesados adecuadamente; además no existe un sistema informático o un banco de información que permita consultar el estado de edificaciones específicas como escuelas, hospitales, etc.

1.3.1 Diagnóstico del problema

Para poder diagnosticar el problema se hará uso de dos técnicas, estas son: análisis FODA y diagrama de Causa-Efecto (Diagrama de Ishikawa).

1.3.1.1 Análisis FODA

Fortalezas	Debilidades
<ul style="list-style-type: none"> ● Existen instrumentos para la recolección de datos posteriores a una emergencia. ● Personal voluntario con conocimientos técnicos para la realización de evaluación de edificios. ● Existe información georreferenciada de la Facultad de Ingeniería y Arquitectura de la Universidad de El Salvador. 	<ul style="list-style-type: none"> ● No se tabula la información recolectada. ● Falta de capacitación previa a todo el personal involucrado en las evaluaciones ● Resultados subjetivos en evaluaciones realizadas. ● Llenar los actuales instrumentos de evaluación es lento y tedioso. ● Evaluaciones realizadas sólo en caso de emergencia ● Poca coordinación en las jornadas de evaluación.
Oportunidades	Amenazas
<ul style="list-style-type: none"> ● Muchas organizaciones que pueden aportar profesionales para las evaluaciones. ● Manejar asignación de puntajes de manera objetiva posterior al ingreso de los datos de los edificios. 	<ul style="list-style-type: none"> ● No existe institución oficial que maneja la información de los edificios que representan riesgos. ● Los resultados de las evaluaciones no clasifican el estado de la edificación. ● No se conoce el estado actual de las

<ul style="list-style-type: none"> • Procesamiento de los datos recolectados y generación de reportes. • Crear una interfaz que permita una recolección de datos más rápida y simple. • Construcción de un generador de formularios de inspección dinámico. 	edificaciones
--	---------------

Tabla 1: Análisis FODA

Formulación de estrategias

<p>Formulación de estrategias FODA</p>	<p>Fortalezas (F)</p> <ul style="list-style-type: none"> • Existen instrumentos para la recolección de datos posteriores a una emergencia. • Personal voluntario con conocimientos técnicos para la realización de evaluación de edificios. • Existe información georreferenciada de la Facultad de Ingeniería y arquitectura de la Universidad de El Salvador. 	<p>Debilidades (D)</p> <ul style="list-style-type: none"> • No se tabula la información recolectada. • Falta de capacitación previa a todo el personal involucrado en las evaluaciones • Resultados subjetivos en evaluaciones realizadas. • Llenar los actuales instrumentos de evaluación es lento y tedioso. • Evaluaciones realizadas sólo en caso de emergencia • Poca coordinación en las jornadas de evaluación.
---	---	--

<p>Oportunidades (O)</p> <ul style="list-style-type: none"> • Muchas organizaciones que pueden aportar profesionales para las evaluaciones. • Manejar asignación de puntajes de manera objetiva posterior al ingreso de los datos de los edificios. • Procesamiento de los datos recolectados y generación de reportes. • Crear una interfaz que permita una recolección de datos más rápida y simple. • Construcción de un generador de formularios de inspección dinámico. 	<p>FO (Maxi – Maxi)</p> <p>Estrategia para maximizar tanto las F como las O.</p> <ul style="list-style-type: none"> • Aprovechar los voluntarios que existen en las diferentes organizaciones para realizar capacitaciones a través de las mismas y obtener resultados más precisos de las inspecciones. • Utilizar la información existente para (Tanto formularios como información geográfica) para generar un banco de datos centralizado. • Sistematizar todo el proceso de recolección y procesado de información. • Sistematizar la construcción de instrumentos para inspecciones. 	<p>DO (Mini – Maxi)</p> <p>Estrategia para minimizar las D y maximizar las O.</p> <ul style="list-style-type: none"> • Utilizar formularios establecidos por las organizaciones para estandarizar las evaluaciones. • Sistematizar el procesamiento de los datos recolectados para acelerar el proceso. • Sistematizar la construcción de herramientas que permitan evaluar las edificaciones
<p>Amenazas (A)</p> <ul style="list-style-type: none"> • No existe institución oficial que maneja la información de los edificios que representan riesgos. • Los resultados de las evaluaciones no clasifican el estado de la edificación. • No se conoce el estado actual de las edificaciones. 	<p>FA (Maxi – Mini)</p> <p>Estrategias para maximizar las fortalezas y minimizar las amenazas.</p> <ul style="list-style-type: none"> • Utilizar herramientas que hagan el cálculo cuantitativo del estado de las edificaciones. • Realizar inspecciones con el personal voluntario para tener una base de datos actualizada del estado de las edificaciones. 	<p>DA (Mini – Mini)</p> <p>Estrategias para minimizar tanto las D como las A.</p> <ul style="list-style-type: none"> • Centralizar los voluntarios por institución para coordinar esfuerzos de recolección de datos. • Facilitar la recolección de la información del estado de los edificios mediante herramientas informáticas amigables para el usuario.

Tabla 2: Formulación de estrategias

Conclusión:

A pesar de que existen herramientas (Formularios) desarrollados por las diferentes instituciones como ASIA y CASALCO no existe un banco de datos que centralice esta información y así poder conocer el estado actual de las edificaciones; así también es notable que existen organizaciones que realizan esfuerzos para reunir a voluntarios y recolectar información sin embargos en muchos de estos casos el personal no está debidamente capacitado. Por tanto, se plantean estrategias enfocadas a Facilitar la recolección y el procesado de los mismos centralizando dicha información para conocimiento de los interesados. También se hace énfasis en la estrategia de capacitar al personal mediante las instituciones voluntarias que ya realizan esfuerzos en la temática.

1.3.1.2 Análisis de Ishikawa

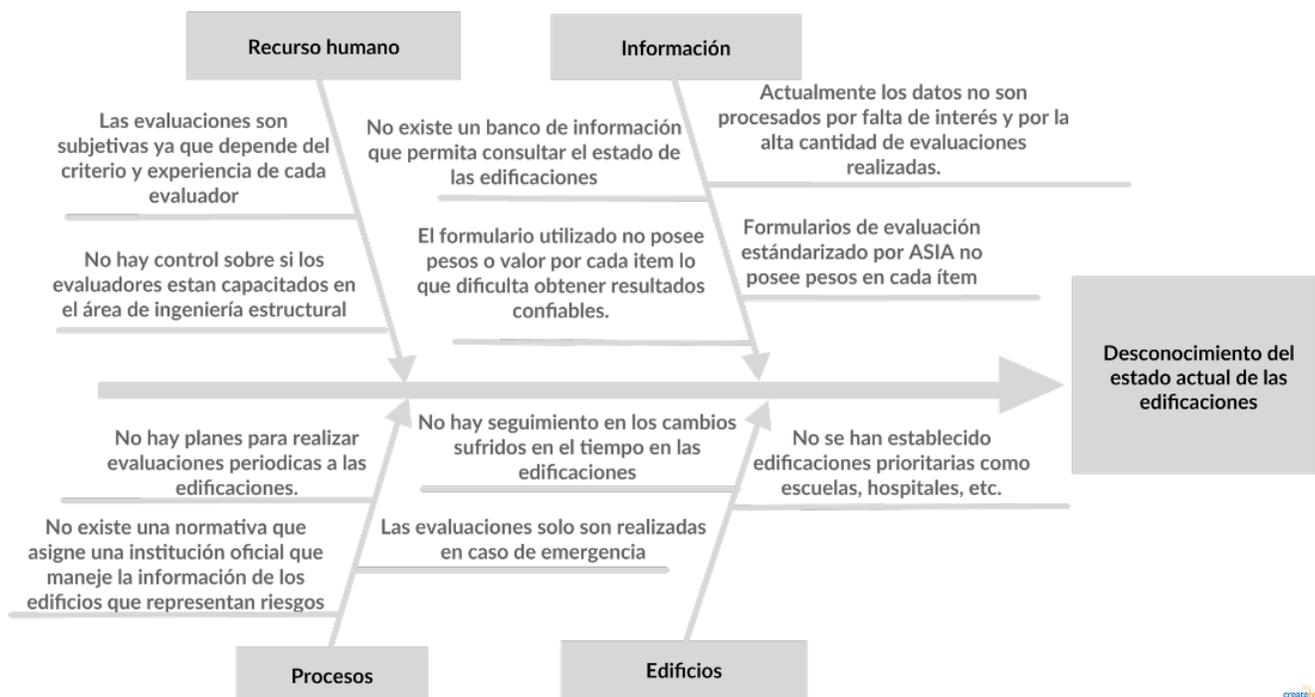


Ilustración 2. Diagrama de Ishikawa.

Elementos de la problemática:

- **Recurso humano:** Son los profesionales de Ingeniería Estructural y Civil autorizados que serán los responsables de realizar los trabajos de evaluación en campo, de la inspección de las edificaciones, recopilación de la información en campo, evaluación de daños, diligenciamiento de los formularios para inspección y señalización de las edificaciones con su respectiva clasificación de habitabilidad mediante la colocación de avisos y de colores. Se definen los siguientes roles dentro de la evaluación de edificaciones:

- Coordinadores de las brigadas de evaluación: son los encargados de entregar los formularios a los supervisores y recibirlos una vez haya sido llenados, revisados y según hayan sido clasificados por los supervisores dependiendo del dictamen de habitabilidad.
 - Supervisores: Son los encargados de distribuir al personal, de las brigadas de evaluación disponibles en cada zona y preparar las rutas de trabajo, entregarles y verificar que el material y equipo esté completo. Preparar el informe de los resultados que se obtengan cada día, reportes semanales y el reporte final de todas las edificaciones inspeccionadas en el municipio. Posteriormente entregar estos informes a los coordinadores departamentales.
 - Evaluadores: estos serán los responsables directos de realizar las inspecciones de campo, es decir inspeccionar las edificaciones, recopilar la información de campo, evaluar los daños, llenar el formulario de evaluación y colocar los avisos de habitabilidad a la que se llegue según la evaluación de emergencia y posteriormente dar el dictamen de la habitabilidad de la evaluación detallada (en el caso de las edificaciones en las que se realizó).
- Información: son el resultado de los datos recolectados a partir de las evaluaciones de edificaciones. La cual contiene detalles sobre el uso predominante, sistema estructural, año de construcción, daños en elementos arquitectónicos y elementos estructurales; los cuales permiten conocer el estado de la edificación.
 - Procesos: Procedimientos definidos para la evaluación de edificaciones, entre estos procesos están procedimiento de inspección que incluyen la observación del suelo, exterior de la edificación, colocación de avisos para la clasificación de las edificaciones, identificar los daños y el llenado de formulario de notificación a las autoridades pertinentes sobre el estado de la edificación, procesamiento de los datos y la planeación que permite organizar a los profesionales de ingeniería estructural y civil, administrar el material y equipo necesarios para realizar las evaluaciones.
 - Edificios: Elementos a evaluar para determinar el riesgo y clasificar el estado en que se encuentra.

1.3.1.3 Resultado del diagnóstico

Inexistencia de una herramienta flexible para la recolección de datos que facilite el cambio de enfoque en los formularios de evaluación. Es decir, una herramienta que le siga el paso a los constantes cambios en requisitos de lo que se desea recolectar, ya que las herramientas existentes (formularios en papel) se vuelven obsoletos en cuanto el enfoque con el que se desea llevar a cabo una inspección cambia, y que todos estos datos recolectados por los formularios cambiantes puedan centralizarse y procesarse, para el conocimiento de zonas de riesgo o daños típicos en edificaciones.

Los tipos de formularios actuales dependen demasiado de la subjetividad por parte del humano que lleva a cabo la evaluación, significando esto que los resultados podrían ser no fiables u honestos, por lo que se requiere que la misma herramienta pueda realizar un juicio (evaluaciones de reglas impuestas por el creador del formulario) de lo que se recolecta y brindar un resultado final automatizado, removiendo así buena parte de la subjetividad del evaluador.

Actualmente, debido a la descentralización de los esfuerzos realizados por las distintas entidades que velan por la seguridad e integridad humana, no existe institución central responsable de validar las herramientas de evaluación utilizadas para la inspección de edificaciones. Así mismo, no existe entidad que capacite y certifique a los evaluadores que llevan a cabo los censos. Por ello, un sistema que facilite la gestión de las herramientas de evaluación, evaluadores, capacitaciones y entidades certificadas se vuelve requerido para la coordinación humana y gestión de información.

1.4 Determinación de factibilidades

1.4.1 Factibilidad técnica

1.4.1.1 Recursos humanos

Área de negocio

Vicedecanato de facultad de ingeniería y arquitectura y Escuela de Ingeniería Civil.

Área de desarrollo

Cantidad	Descripción
1	Administrador de proyecto/Programador
1	Administrador de Bases de Datos/Programador
1	Analista de Sistemas/Programador

Tabla 3: Requerimiento de recursos humanos.

Durante las diferentes etapas del proyecto los roles serán asumidos con base a la experiencia del equipo de trabajo.

1.4.1.2 Conocimientos con los que se cuenta

Área de negocio

El usuario de negocio posee amplios conocimientos técnicos en materia de edificaciones, proceso de evaluación de edificios, utilización de formulario para la evaluación de edificios posterior a un sismo y categorización de edificaciones según su uso.

Área de desarrollo

El equipo de desarrollo posee los siguientes conocimientos y habilidades:

- Desarrollo de sistemas informáticos
- Administración de proyectos informáticos
- Conocimientos sobre sistemas de información geográfica
- Desarrollo de aplicaciones móviles
- Administración y programación de bases de datos.

1.4.1.3 Recursos materiales

Para el desarrollo de este proyecto se requiere como mínimo tener los siguientes recursos de hardware y software:

Recursos de Hardware

Recurso requerido	Descripción	Requerido	Se posee
Estaciones de trabajo	Procesador Pentium 4 o superior Memoria RAM 2GB Disco duro 20GB	3	3
Dispositivo móvil	Dispositivo móvil de gama media	1	1
Servidor de desarrollo	Procesador Xeon E7 o superior Memoria RAM 8GB Disco duro 300GB Tarjeta de red de 10/100 Mbps	1	1

Tabla 4: Requerimiento de hardware.

El equipo de desarrollo posee recursos de hardware con las siguientes características técnicas:

Recurso	Cantidad	Características técnicas
Estaciones de trabajo	1	Procesador: Intel Core i7 3.60GHz Memoria RAM: 16 GB Disco Duro: 1 TB Sistema Operativo: Windows 10
	1	Procesador: Intel Core i5 3.10GHz Memoria RAM: 8 GB Disco Duro: 500 GB Sistema Operativo: Windows 10
	1	Procesador: Intel Core i5 3.20GHz Memoria RAM: 16 GB Disco Duro: 300 GB Sistema Operativo: Windows 10
Dispositivo móvil	1	Samsung Galaxy S6 Edge+ Ram: 4 GB Memoria interna:32 GB Núcleos: 8 núcleos de 2.1 Ghz

Servidor de desarrollo	1	Linode 8GB Ram: 8 GB CPU: 4 CPU Cores Disco duro: 500GB Gbps Network: in 40 GB / out 1000 GB
-------------------------------	---	--

Tabla 5: Características técnicas del hardware.

Recursos de Software

Recurso	Descripción	Software que se posee
Sistema operativo servidor / cliente web	<ul style="list-style-type: none"> • Soportar aplicaciones en redes • Gestión de errores • Multiprocesos • Seguridad • Estabilidad 	<ul style="list-style-type: none"> • Debian 9 Stretch • Windows 10
Sistema operativo cliente móvil	<ul style="list-style-type: none"> • Multitarea • Multiproceso • Sistema de Posicionamiento Global GPS • Conectividad Inalámbrica • Administración del Hardware • Administración de Aplicaciones 	<ul style="list-style-type: none"> • Android 5
Servidor web	<ul style="list-style-type: none"> • Multiplataforma. • Soporte de operaciones asíncronas y E/S sin bloqueo. • Manejo de errores. • Alto rendimiento. 	<ul style="list-style-type: none"> • NodeJS/ Express
Gestor de base de datos	<ul style="list-style-type: none"> • Alto rendimiento en escritura/lectura de data. • Modelos de datos flexibles. • Almacenamiento de datos geoespaciales. • Ejecución de queries geoespaciales. • Escalabilidad horizontal y vertical. 	<ul style="list-style-type: none"> • Mongo DB
Herramienta de control de versiones	<ul style="list-style-type: none"> • Flujo de trabajo centralizado. • Control de versiones de repositorios locales distribuido. • Creación de ramas con roles. • Integridad criptográfica de la data. • Procesamiento veloz de las operaciones. • Áreas intermedias para revisión de commits. 	<ul style="list-style-type: none"> • Git / Github

Herramientas utilitarias	<ul style="list-style-type: none"> • Suit Ofimática • Navegador web 	<ul style="list-style-type: none"> • Microsoft Office • Google Chrome
Herramientas de diseño	<ul style="list-style-type: none"> • Modelado de bases de datos • Creación de mockups • Diagramador UML 1.4 	<ul style="list-style-type: none"> • ERDPlus • Draw.io • Astah Comunity

Tabla 6: Requerimiento de software.

1.4.1.4 Conclusión

Por lo anterior podemos concluir que se tiene el recurso necesario para el desarrollo del proyecto a nivel técnico: conocimientos, equipos, software y herramientas necesarias, por lo que el desarrollo del proyecto es factible técnicamente.

1.4.2 Factibilidad económica

Con base a los recursos necesarios para el desarrollo del proyecto se estiman necesarios los siguientes recursos para el funcionamiento del proyecto.

1.4.2.1 Recursos de Hardware

A continuación, se detallan los gastos por depreciación de equipos a utilizar⁸.

Recurso	Descripción	Cantidad	Costo/Mes	No. Meses	Total
Estación de trabajo	Procesador Core i3 o superior Memoria RAM 8GB Disco duro 500GB	3	\$ 33.33	9	\$900.00
Dispositivo móvil	Dispositivo móvil de gama media	1	\$ 4.16	9	\$37.50
Servidor de desarrollo	Procesador Xeon E7 o superior Memoria RAM 8GB Disco duro 300 GB Tarjeta de red de 10/100 Mbps	1	\$ 50.00	9	\$450.00
Total					\$ 1387.50

Tabla 7: Costos de recursos de hardware.

1.4.2.2 Recursos de Software

A continuación, se detallan los costos del software a utilizar⁹.

⁸ Ver #Anexo D: Cálculos de costos.

⁹ Ver #Anexo D: Cálculos de costos.

Categoría	Software	Cantidad de licencias	Costo/ Mes	No. Meses	Total
Sistema operativo servidor / cliente web	Debian 9 Stretch	1	\$ 00.00	9	\$ 00.00
	Windows 10	3	\$ 06.04	9	\$ 163.08
Sistema operativo cliente móvil	Android 5	1	\$ 00.00	9	\$ 00.00
Servidor web	NodeJS/ Express	1	\$ 00.00	9	\$ 00.00
Gestor de base de datos	Mongo DB	1	\$ 00.00	9	\$ 00.00
Herramienta de control de versiones	Git	3	\$ 00.00	9	\$ 00.00
Herramientas utilitarias	Microsoft Office	3	\$ 06.99	9	\$ 188.73
	Google Chrome	3	\$ 00.00	9	\$ 00.00
Herramientas de diseño	ERDPlus	3	\$ 00.00	9	\$ 00.00
	Draw.io	3	\$ 00.00	9	\$ 00.00
	Astah Comunity	3	\$ 00.00	9	\$ 00.00
Total					\$ 351.81

Tabla 8: Costos de recursos de software.

1.4.2.3 Costos de desarrollo

A continuación, se detallan los salarios del equipo de desarrollo¹⁰.

Descripción	cant.	Horas semanales	Costo/ Hora	Costo/ Mes	No. Meses	Total
DESARROLLO DEL SOFTWARE						
Administrador de proyecto	1	20	\$ 5.00	\$ 400.00	9	\$ 3,600
Analista de Sistemas/Programador	2	20	\$ 3.33	\$ 266.64	9	\$ 2,399.76

¹⁰ Ver #Anexo D: Cálculos de costos.

Administrador de Base de datos/ Programador	1	20	\$ 4.85	\$ 388.00	9	\$ 3,492
			Total			\$ 9,491.76

Tabla 9: Costos de recurso humano.

1.4.2.4 Costos fijos mensuales

Descripción	Total / mes
Local	\$ 240
Papelería	\$ 10
Servidores de desarrollo	\$ 60
Transporte y alimentación	\$ 75
Imprevistos	\$ 38.5
TOTAL	\$ 423.5
TOTAL A 9 MESES	\$ 3,811.5

Tabla 10: Costos fijos mensuales.

- El costo de local incluye energía eléctrica, agua e internet.
- El costo de imprevistos se calculó como el 10 % de los costos fijos.

1.4.2.5 Costo total del proyecto

Costo depreciación de hardware	\$ 1,387.50
Costo de software	\$ 351.81
Costo de desarrollo	\$ 9,491.76
Costo fijo	\$ 3,811.5
Total	\$ 15,042.57

Tabla 11: Costos totales del proyecto.

1.4.2.6 Beneficio del proyecto

Para cuantificar el beneficio del proyecto evaluaremos el costo de una evaluación de edificios antes y después de la utilización del sistema informático.

Para la comparación se toma como base la experiencia de evaluaciones realizadas utilizando la herramienta de VISUS que se implementa en el 2014 como un esfuerzo en conjunto de la UNESCO y la Facultad de Ingeniería y Arquitectura de la Universidad El Salvador para realizar la evaluación en escuelas y generar un inventario georreferenciado de las instituciones, el uso de dicha herramienta ayudó a minimizar los tiempos de las inspecciones a un tercio del sin la utilización de VISUS, con la implementación del sistema se esperan los mismos resultados con respecto a la reducción de los tiempos de evaluación, ahora en un contexto de edificaciones generales. Nuestra edificación base

constará de una estructura con 5 niveles. La estimación de los costos de inspección previos a la utilización del sistema se detalla en la siguiente tabla¹¹.

Actividad	Cantidad/ personas	Costo/hora	Cantidad de horas	Costo total
Evaluador especialista	2	\$6.25	90	\$1,125
Evaluador de apoyo	3	\$2.5	90	\$675
Totales				\$1,800

Tabla 12: Costo actual de una inspección.

La estimación de los costos de inspección posterior a la utilización del sistema se detalla en la siguiente tabla.

Actividad	Cantidad/ personas	Costo/hora	Cantidad de horas	Costo total
Evaluador especialista	2	\$6.25	30	\$375
Evaluador de apoyo	3	\$2.5	30	\$225
Totales				\$600

Tabla 13: Costo de una inspección con el uso del sistema.

Por tanto, al evaluar una edificación base tendremos un ahorro de \$ 1,200.

1.4.2.7 Conclusión

El desarrollo del sistema genera un costo de \$15,042.57, el beneficio que se obtiene con la utilización del sistema para la evaluación de las edificaciones radica en la reducción de un 66.66% del costo de evaluación de un edificio, por lo tanto, se concluye que la inversión que genera el desarrollo de dicho proyecto se recupera con la inspección de 13 edificios, por lo que su implementación es factible económicamente hablando¹².

¹¹ Ver Anexo E: Cálculo de beneficios.

¹² Ver Anexo E: Cálculo de beneficios.

1.4.3 Factibilidad operativa

1.4.3.1 Uso del sistema

Para poder asegurar que el sistema sea usado como ha sido planeado se han considerado cuatro aspectos:

1. Nivel de complejidad del sistema: El sistema de información para el censo y evaluación de edificios brindará una interfaz amigable con el usuario de manera que el manejo e ingreso de datos dentro de la aplicación se realice de una manera intuitiva al igual que la plataforma de administración.
2. Resistencia al cambio por parte de los usuarios: El uso de la aplicación para la evaluación de edificios permitirá una disminución en los tiempos de ingreso de datos y número de personas requeridas para dicha actividad, por lo tanto, se prevé una aceptación del sistema por parte de los usuarios
3. Nivel de adaptación al nuevo sistema: Los usuarios están en constante capacitación para el uso de los instrumentos de evaluación que utilizan, la utilización del sistema de información para el censo y evaluación de edificios se incorporará como parte de dichas capacitaciones y dado que está basado en los instrumentos existentes la adaptación no conllevará mayor esfuerzo.
4. Nivel de obsolescencia: El sistema será construido con tecnologías de bajo nivel de obsolescencia, utilizando en su mayoría lenguaje JavaScript en conjunto con librerías como React.js para el cliente web o React Native para el cliente móvil, lo cual nos asegura que futuras versiones de navegadores web o actualizaciones a sistemas Android puedan trabajar perfectamente con el sistema.

Los requisitos mínimos para el uso del sistema se detallan a continuación:

Cliente web

Características	Requerimientos mínimos
CPU	Pentium 4
Memoria RAM	2GB
Disco Duro (Espacio Disponible)	20GB
Interfaz de Red	Tarjeta de red de 10/100 Mbps
Monitor	<ul style="list-style-type: none">• VGA o HDMI• 14"
Periféricos	<ul style="list-style-type: none">• Teclado• Mouse
SO	Cualquiera de los siguientes: <ul style="list-style-type: none">• Windows 7

	<ul style="list-style-type: none"> • Debian 8 • Mac OS X
Software	<ul style="list-style-type: none"> • Navegador web <ul style="list-style-type: none"> • Google Chrome. • Mozilla Firefox

Tabla 14: Requerimiento de hardware y software para el uso del sistema web.

Cliente móvil

Características	Requerimientos mínimos
CPU	Procesador Snapdragon 801 2.5GHz o superior
Memoria RAM	2GB
Almacenamiento interno	8GB
Pantalla	4.7", 1080 x 1920 pixels
Cámara	16 MP
SO	Android 5.0+
Extra	GPS

Tabla 15: Requerimientos de hardware y software para el uso de la aplicación móvil.

El equipo requerido para el uso del sistema ya lo poseen los usuarios o es fácil de adquirir en caso necesario.

1.4.3.2 Puesta en producción

A continuación, se detalla los recursos requeridos para la puesta en producción del sistema, con su respectivo detalle de existencia.

Recurso requerido	Descripción	Requerido	Se posee
Técnico para implementación del sistema	Persona responsable de la instalación, ejecución y correcta configuración del sistema informático. Conocimientos necesarios: <ul style="list-style-type: none"> • Conocimientos sólidos de redes • Conocimiento de configuración de servidores • Conocimiento de bases de datos NoSQL • Conocimientos de servidor Node.js 	1	1 (Esta persona será puesta a disposición por la contraparte)
Servidor	Características:	1	1 (Será puesto a

	<ul style="list-style-type: none"> • CPU: Xeon E7 o superior. • RAM: 8GB. • Disco Duro: 500GB hábiles. Con redundancia distribuida con paridad y discos de reserva (RAID 5E). • 2 Fuentes de alimentación eléctrica redundantes. • 2 Interfaces de Red: Tarjeta de red de 10/100 Mbps. 		disposición por la contraparte)
--	---	--	---------------------------------

Tabla 16: Requerimientos hardware y humano para la puesta en producción del sistema.

1.4.3.3 Conclusión

Se cuenta con los recursos necesarios para la correcta operatividad del sistema informático y se prevee que tenga una aceptación positiva por parte de los usuarios, la adaptación será fácil a la hora de observar los beneficios que se otorgan con la implementación de las nuevas herramientas.

1.4.4 Resumen de factibilidades

A continuación, se detallan las razones por las cuales se concluyó que la implementación del “Sistema de Información para el Censo y Evaluación de Edificios en la Facultad de Ingeniería y Arquitectura de la Universidad de El Salvador con la Aplicación de Tecnologías GIS” es factible, tomando como base los resultados obtenidos en el análisis de las factibilidades técnica, económica y operativa:

- Desde el punto de vista técnico, se cuenta con las herramientas tecnológicas y el conocimiento necesario para llevar a cabo el desarrollo e implementación del proyecto, además en caso de requerir nuevo equipo para la implementación se tiene la disposición por parte de los usuarios para la adquisición de servicios de Hosting para el sistema, por lo que es factible técnicamente llevar a cabo el sistema.
- En cuanto a la factibilidad económica, si bien el sistema presenta un costo de \$18,102.78 el ahorro en la evaluación de cada edificio mediante el uso del sistema es de \$1200 por lo que la inversión del proyecto se recupera con el ahorro obtenido al evaluar 16 edificios, por lo tanto, es factible la implementación del sistema.
- La Factibilidad Operativa se garantiza ya que el sistema informático podrá ser utilizado sin mayores dificultades por parte de los usuarios, y con el uso del sistema se prevé que haya una disminución del 66.66% en los tiempos de evaluación de edificaciones lo que implica mayor productividad al momento de las evaluaciones.

1.5 Resultados esperados

- Reportes dinámicos y parametrizable tomando como base los siguientes reportes:

- Reporte de inspecciones realizadas a edificios.
 - Reporte de detalle de inspección realizada.
 - Reporte de perfil de edificaciones.
 - Reporte de estado de edificación.
 - Reporte de histórico de edificación.
 - Reporte de listado de módulos.
 - Reporte de usuarios registrados en el sistema.
 - Reporte de capacitaciones disponibles.
 - Reporte de capacitaciones recibidas por usuarios.
- Evaluación de edificaciones por medio de diferentes módulos de inspección creados en el sistema: el sistema a desarrollar permitirá la creación de nuevos instrumentos de evaluación a los cuales se les podrá incluir elementos calculados y una lógica de generación de banderas.
 - Coordinación de jornadas de inspecciones: el sistema a desarrollar permitirá programar jornadas de inspecciones y dar aviso a los evaluadores capacitados, igualmente permitirá llevar un control de las edificaciones inspeccionadas en tiempo real, de manera que si hay edificaciones que hayan quedado pendientes de inspeccionar, se pueda dar aviso rápidamente para que los inspectores las cubran.
 - Sistema de avisos: El sistema tendrá un mecanismo de avisos sobre desastres naturales u otro tipo de información vital para los evaluadores de edificaciones.
 - Gestionar el catálogo de evaluadores capacitados: El sistema permitirá gestionar las capacitaciones recibidas por los evaluadores de edificaciones, para poder tener actualizado un catálogo de evaluadores vigente. Con base a las capacitaciones recibidas los evaluadores tendrán acceso a diferentes módulos de evaluación de acuerdo a sus competencias.
 - Gestión de instituciones evaluadoras: El sistema dará soporte para registrar instituciones que puedan brindar las capacitaciones de los diferentes módulos de evaluación.

1.6 Metodología de desarrollo

Para resolver un problema se hace necesaria la aplicación de un método que permita obtener una solución de manera eficiente, la metodología es la ciencia que aplica el método para la resolución del problema.

El método de cascada es considerado como el enfoque clásico para el ciclo de vida del desarrollo de sistemas, por lo tanto, para este proyecto se aplicará dicho método cumpliendo con las etapas que se mencionan a continuación:

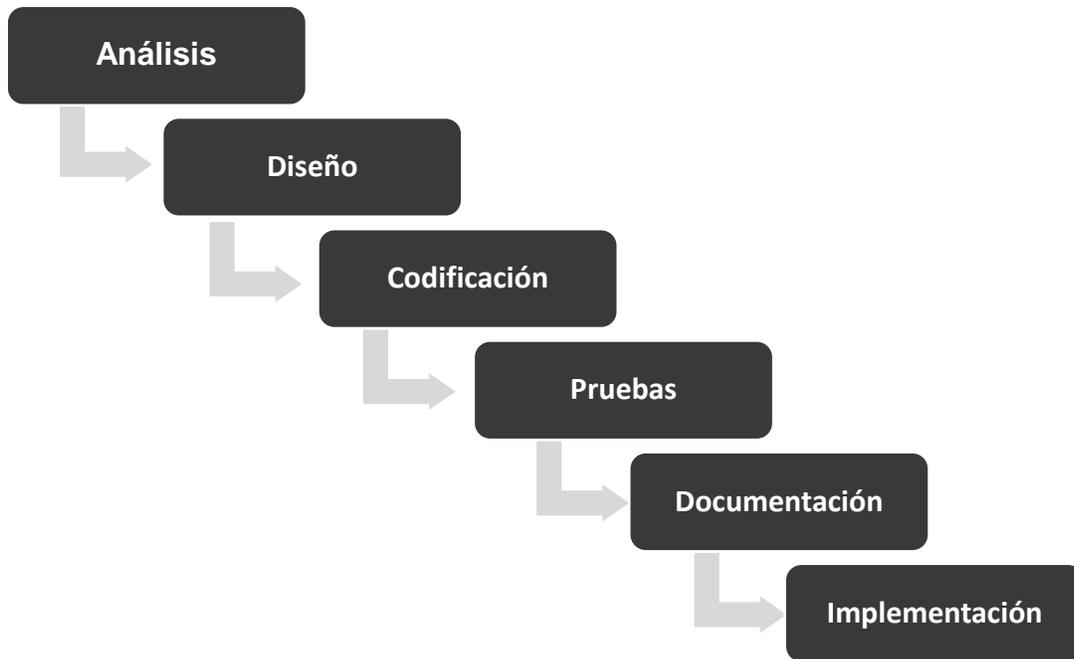


Ilustración 3. Ciclo de desarrollo en cascada.

1.6.1 Análisis

Esta etapa tiene por objeto realizar un análisis global del sistema, involucra la identificación de las características que nos guían para determinar las funcionalidades del software de acuerdo al medio donde se pretende implementar, el análisis involucra las siguientes etapas:

- **Análisis de la situación actual**

Es importante conocer la forma en que se realizan las evaluaciones de edificios previa y post-sísmica en la actualidad, si hay documentación previa sobre el tema o precedentes en cuanto a sistemas de evaluación de edificaciones, y la forma en que se utiliza la información recolectada, y de esta manera identificar oportunidades de mejora en los procesos existentes.

Para llevar a cabo este análisis nos apoyaremos de los siguientes recursos:

- Entrevista: esta herramienta es una forma específica de interacción con los usuarios que permite la realización de preguntas específicas que permiten recolectar información relevante que ayude a entender las necesidades o requerimientos del sistema a desarrollar.
- Investigación: consiste en realizar una investigación acerca de proyectos que se relacionen con la evaluación de edificaciones en el país, así como conocer si se posee un inventario de las edificaciones que han sido dañadas por los diversos fenómenos naturales, esto como base para conocer la forma de proceder después de cada evaluación.

- **Determinación de Requerimientos**

Los requerimientos del sistema se obtienen mediante entrevistas que permiten conocer cuáles son los requerimientos funcionales, no funcionales, de desarrollo, y de producción del proyecto, dichos requerimientos deben ir encaminados con el cumplimiento de los objetivos que se establecieron en el perfil del proyecto.

Como apoyo a la determinación de los requerimientos se realizarán los diagramas de casos de uso de UML.

Técnicas de Análisis

A continuación, se describen algunas de las técnicas que se proponen para la realización del análisis del sistema.

- **Diagrama de sistemas:** El enfoque sistémico, es un método que nos permite comprender aspectos de la realidad, pero desde una vista de conjunto, de totalidad, y haciendo uso del concepto de sistema, para ello se utiliza un diagrama denominado Diagrama de entrada-proceso-salida o Diagrama de Sistemas. y consta de los siguientes elementos:
 - **Entradas:** Son todos los datos que hay que ingresar para obtener la solución del problema.
 - **Procesos:** Son los diferentes procedimientos en los cuales se usarán los datos proporcionados por el usuario para resolver el problema.
 - **Salidas:** Es la solución del problema o resultados esperados.
 - **Medio Ambiente:** el conjunto de objetos exteriores al sistema, pero que influyen decididamente a éste, y a su vez el sistema influye, aunque en una menor proporción.
 - **Frontera:** Es el límite del sistema.
 - **Control:** Permite el control de un sistema y que el mismo tome medidas de corrección en base a la información retroalimentada.
- **Ishikawa:** Consiste en una representación gráfica sencilla en la que puede verse de manera relacional una especie de espina central, que es una línea en el plano horizontal, representando el problema a analizar, que se escribe a su derecha.
Este diagrama causal es la representación gráfica de las relaciones múltiples de causa-efecto entre las diversas variables que intervienen en un proceso.

La estructura del Diagrama de Ishikawa es intuitiva: identifica un **problema** o **efecto** y luego enumera un conjunto de **causas** que potencialmente explican dicho comportamiento. Adicionalmente cada causa se puede desagregar con grado mayor de detalle en **subcausas**.
- **FODA:** La matriz FODA es una herramienta de análisis que puede ser aplicada a cualquier situación, individuo, producto, empresa, etc, que esté actuando como objeto de estudio en un momento determinado del tiempo.

El análisis FODA es una herramienta que permite conformar un cuadro de la situación actual del objeto de estudio (persona, empresa u organización, etc.) permitiendo de esta manera obtener un diagnóstico preciso que permite, en función de ello, tomar decisiones acordes con los objetivos y políticas formulados.

El objetivo primario del análisis FODA consiste en obtener conclusiones sobre la forma en que el objeto estudiado será capaz de afrontar los cambios y las turbulencias en el contexto, (oportunidades y amenazas) a partir de sus fortalezas y debilidades internas.

- **Narrativa de casos de Uso**

En el Lenguaje de Modelado Unificado (UML), un caso de uso es una secuencia de interacciones entre un sistema y alguien o algo que usa alguno de sus servicios, describe la secuencia de eventos de un actor que utiliza un sistema para completar un proceso.

Para la realización de la narrativa de los casos de uso se sigue el siguiente formato:

- **Caso de uso:** nombre del caso de uso.
- **Actores:** lista de actores en la que se indica quién inicia el caso de uso.
- **Tipo:** que puede ser Primario, Secundario, Opcional, entre otros.
- **Descripción:** repetición del caso de uso de alto nivel o alguna síntesis similar.
- **Propósito:** intención del caso de uso.
- **Referencias Cruzadas:** casos de uso y funciones relacionadas con el sistema.
- **Curso Normal de los Eventos:** es la parte medular del formato expandido; describe los detalles de la conversión interactiva entre los actores y el sistema. Un aspecto esencial de la sección es explicar la secuencia más común de eventos: la historia normal de las actividades y el término exitoso de un proceso. No incluye situaciones alternativas.
- **Cursos Alternativos:** describe importantes opciones o excepciones que pueden presentarse en relación al curso normal. Si son éstas son complejas, se pueden expandir y convertir en nuevos casos de uso

Ejemplo

Caso de Uso	Comprar Productos	
Actores	Cliente (iniciador), Cajero	
Propósito	Capturar una venta y su pago	
Tipo	Principal y esencial	
Descripción	Un Cliente llega a una caja con productos que desea comprar. El Cajero registra los productos y obtiene el pago. Al terminar la transacción, el Cliente se marcha con los productos.	
Referencias Cruzadas	Casos de Uso: el Cajero debe haber terminado el caso de uso llamado <i>Registrar</i>	
Curso Normal de los Eventos		
	Acción de los Actores	Respuesta del Sistema
	1.- Este caso de uso comienza cuando un Cliente llega a la caja con productos que desea comprar.	
	2.- El Cajero registra los productos. Si hay más de un producto, también puede introducir la cantidad.	3.- Determina el precio del producto y agrega la información sobre él a la actual transacción de venta.
	4.- Al terminar el registro de los productos, el Cajero indica al sistema que terminó dicho proceso.	5.- Calcula y presenta el total de la venta.
	6.- El Cajero le indica el total al Cliente.	
	7.- El Cliente escoge la forma de pago: a) Si paga en efectivo, ver la sección <i>Pago en Efectivo</i> b) Si paga con tarjeta con crédito, ver la sección <i>Pago con Tarjeta de Crédito</i> c) Si paga con cheque, ver la sección <i>Pago con Cheque</i>	8.- Registra la venta terminada.
		9.- Actualiza los niveles de inventario.
		10.- Genera un recibo.
	11.- El Cajero entrega el recibo al cliente.	
	12.- El Cliente se marcha con los productos comprados.	
Sección: Pago en Efectivo		
Curso Normal de los Eventos		
	Acción de los Actores	Respuesta del Sistema
	1.- El Cliente da un pago en efectivo, posiblemente mayor que el total de la venta.	
	2.- El Cajero registra el efectivo recibido.	3.- Entrega la diferencia al Cliente.
	4.- El cajero guarda el efectivo recibido y saca la diferencia. Luego, le entrega el vuelto al Cliente.	
Cursos Alternativos – Sección Pago en Efectivo		
- Línea 1: el Cliente no tiene suficiente efectivo. Puede cancelar o iniciar otro método de pago.		
- Línea 4: la caja no tiene suficiente efectivo para pagar la diferencia. El Cajero pide más efectivo al supervisor o le pide al Cliente otro billete de menor valor u otra forma de pago.		

Ilustración 4. Narrativa de caso de uso.

- **Diagrama de casos de Uso**

Un 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 principales. Los requisitos no se describen en detalle, ya que esto puede hacerse en otros diagramas o en documentos que pueden vincularse a cada caso de uso.

Permite a desarrolladores y usuarios entender mejor los requerimientos, determinar cuáles son indispensables y cuáles deseables, e identificar riesgos de forma temprana.

1.6.2 Formulación del problema

Actualmente el proceso de evaluación de edificios se ve afectado por los instrumentos utilizados en la evaluación de edificios, como el “Formulario de Evaluación de edificios”, además los evaluadores que no están capacitados. Cabe destacar que el procesamiento de los datos no se realiza.

Actualmente no se tiene conocimiento de cuáles edificios están dañados en nuestro país, ni existe una institución que posea datos centralizados de evaluaciones, lo que implica que se desconoce el riesgo de manera cuantitativa, y como ya se mencionó, nuestro país al estar en una zona de alta actividad sísmica posee una alta vulnerabilidad a nivel estructural debido al desconocimiento de esta información.

El proyecto propuesto consiste en desarrollar un sistema informático que permita crear instrumentos de inspección y que apoye a los ingenieros civiles a recolectar la información de los edificios y determinar si estos representan riesgo para la población por su nivel de daño, por tanto considerando los motivos planteados, surge la necesidad de una herramienta que apoye el proceso, permitiendo reducir el tiempo de recolección de información y evaluación sin afectar la calidad y objetividad de la evaluación, priorizando los recursos, así como reduciendo el desgaste físico y mental que produciría en los evaluadores el evaluar la misma cantidad de edificios en la misma cantidad de tiempo.

El sistema funcionará para la Facultad de Ingeniería y Arquitectura con escalabilidad para cualquier otro sector deseado que proporcione información válida para el sistema. Brindando el sistema la alternativa visual de representar las zonas geográficas a evaluar en un mapa 2D para mejor ubicación del evaluador.

El sistema facilitara las inspecciones ya que permitirá configurar cualquier instrumento de inspección generado por cualquier institución a través del constructor de formulario, permitiendo visualizar los resultados de los datos recolectados a través de mapas, que mostraran los estados de la edificación.

Facilitará concertar una organización de emergencia entre los profesionales aptos y avalados por el encargado del sistema en caso de ocurrido un siniestro, permitiendo evaluar las edificaciones con los instrumentos de inspecciones construidos previamente.

1.6.2.1 Problema general

Inexistencia de información sobre el estado actual de los edificios.

1.6.2.2 Problemas específicos

- No se ha elaborado un plan de evaluaciones periódicas para determinar el estado de las edificaciones, por lo que no se cuenta con un histórico de la variación del estado de las edificaciones.
- Cuando ocurre una emergencia se requiere la inmediata evaluación de las edificaciones para determinar el daño, por lo que se realizan convocatorias a ingenieros estructurales y civiles los

cuales podrían carecer de experiencia en el área estructural por lo que sus evaluaciones se vuelven subjetivas y puede crear una sobreestimación o subestimación del estado de la edificación lo que pondría en riesgo la vida de las personas que la habitan.

- Las evaluaciones realizadas durante los terremotos del 13 de enero y 13 de febrero del año 2001 fueron archivadas y debido a que toda esta información fue tomada en papel se volvió difícil su procesamiento por la alta cantidad de evaluaciones realizadas, además no existen datos estadísticos sobre el número de edificaciones dañadas o en buen estado.
- Al momento de realizar evaluaciones no se han establecido edificaciones prioritarias como hospitales, escuelas, etc., La priorización de las edificaciones es importante porque dependiendo del tipo de edificación se determinará la urgencia de evaluación.

1.6.3 Diseño

Esta etapa una vez aprobados los requerimientos consiste en el diseño de la solución, utilizando herramientas y conocimientos de diseño de aplicaciones, diseño de modelos de la solución y tecnologías utilizadas para satisfacer los requerimientos planteados, esta etapa se divide en las siguientes fases:

1. Diseño preliminar. Se centra en la transformación de requisitos en los datos y la arquitectura del software y comprende:
 - Diseño de datos: Transforma el modelo del campo de la información en las estructuras de datos que se van a requerir para implementar el software.
 - Diseño arquitectónico: Define las relaciones entre los principales elementos estructurales del programa.
 - Diseño de interfaces: Establece la disposición y los mecanismos para la interacción Hombre-Máquina. En esta etapa se hará énfasis en la experiencia del usuario por medio del prototipado, para asegurar interfaces amigables según los usuarios meta e identificar posibles mejoras.
2. Diseño detallado: Se ocupa del refinamiento de la representación arquitectónica que lleva a una estructura de datos detallada y a las representaciones algorítmicas del software.

Como apoyo a esta etapa del desarrollo se realizarán los diagramas de clases, el modelo conceptual, modelo lógico y físico de la base de datos y demás que sean de ayuda para el diseño del sistema.

Una vez definido el diseño se presentará al usuario de negocio para su aprobación, en caso de requerir ajustes se deberá hacer por escrito y proceder a la modificación y posterior aprobación.

Técnicas de Diseño

Las técnicas de diseño dependen del paradigma que se utilice para el desarrollo de los sistemas, en nuestro caso se utiliza la programación asíncrona para el backend.

Para el diseño de la solución se tomaron en cuenta las siguientes técnicas:

- **Diagrama de clases:**

Un diagrama de clases describe la estructura estática de un sistema en términos de clases y de relaciones entre estas clases, mostrando los atributos y operaciones que caracterizan cada clase de objetos.

- La clase define el ámbito de definición de un conjunto de objetos.
- Cada objeto pertenece a una clase.
- Los objetos se crean por instanciación de las clases.

Los diagramas de clases son interacciones entre nodos y arcos, que generalmente representan interacciones, relaciones, interfaces y colaboraciones entre las clases, interfaces, notas, restricciones, paquetes y demás elementos que conforman un programa orientado a objetos.

- **Modelo de datos:**

- **Modelo Entidad Relación:**

El modelo entidad-relación ER es un modelo de datos que permite representar cualquier abstracción, percepción y conocimiento en un sistema de información formado por un conjunto de objetos denominados entidades y relaciones, incorporando una representación visual conocida como diagrama entidad-relación. Se compone por:

Entidad: La entidad es cualquier clase de objeto o conjunto de elementos presentes o no, en un contexto determinado dado por el sistema de información o las funciones y procesos que se definen en un plan de automatización.

Atributos: Son las características, rasgos y propiedades de una entidad, que toman como valor una instancia particular. Es decir, los atributos de una tabla son en realidad sus campos descriptivos

Relación: Vínculo que permite definir una dependencia entre los conjuntos de dos o más entidades. Esto es la relación entre la información contenida en los registros de varias tablas.

Entidades fuertes: Lo constituyen las tablas principales de la base de datos que contienen los registros principales del sistema de información y que requieren de entidades o tablas auxiliares para completar su descripción o información.

Entidades débiles: Son entidades débiles a las tablas auxiliares de una tabla principal a la que completan o complementan con la información de sus registros relacionados.

Clave: Es el campo o atributo de una entidad o tabla que tiene como objetivo distinguir cada registro del conjunto, sirviendo sus valores como datos vinculantes de una relación entre registros de varias tablas.

- **Modelo Lógico:**

Un modelo lógico de datos es un modelo que no es específico de una base de datos que describe aspectos relacionados con las necesidades de una organización para recopilar datos y las relaciones entre estos aspectos. Un modelo lógico contiene representaciones de entidades y atributos, relaciones,

identificadores exclusivos, subtipos y supertipos y restricciones entre relaciones. Un modelo lógico también puede contener objetos de modelo de dominio o referirse a uno o varios modelos de dominio o de glosario.

Una vez definidas las relaciones y los objetos lógicos en un modelo lógico de datos, utilice el área de trabajo para transformar el modelo lógico en una representación física específica de la base de datos en forma de modelo físico de datos.

- **Modelo Físico:**

Un modelo de datos físicos es un modelo específico de bases de datos que representa objetos de datos relacionales (por ejemplo, tablas, columnas, claves principales y claves externas) y sus relaciones. Un modelo de datos físicos se puede utilizar para generar sentencias DDL que, después, se pueden desplegar en un servidor de base de datos.

- **Diseño de arquitectura general del sistema SOA (RESTful)**

SOA (Arquitectura orientada a servicios) es un marco de trabajo conceptual que establece una estructura de diseño para la integración de aplicaciones, que permite a las organizaciones unir los objetivos de negocio, en cuanto a flexibilidad de integración con sistemas legados y alineación directa a los procesos de negocio, con la infraestructura de TI.

Los servicios Web son tecnologías que utilizan un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Permiten también la intercomunicación entre sistemas de cualquier plataforma y se utilizan en una gran variedad de escenarios de integración, tanto dentro de las organizaciones como con socios comerciales.

La Transferencia de Estado Representacional (REpresentation State Transfer - REST) describe un estilo arquitectónico de sistemas en red como, por ejemplo, aplicaciones Web.

REST está comprendida por una serie de limitaciones y principios arquitectónicos. Si una aplicación o diseño cumple con esas limitaciones y principios, se considera RESTful.

Uno de los principios REST de mayor importancia para las aplicaciones Web es que la interacción entre el cliente y el servidor no tiene estado entre solicitudes. Cada solicitud del cliente al servidor debe contener toda la información necesaria para comprender la solicitud. El cliente no se dará cuenta si el servidor debe reiniciarse en ningún momento entre las solicitudes. Asimismo, las solicitudes sin estado pueden ser respondidas por cualquier servidor disponible, lo cual resulta apropiado en un entorno como la computación en nube.

Otro principio REST importante es el de sistema por capas, el cual implica que un componente no puede ver más allá de la capa inmediata con la cual interactúa. Al restringir el conocimiento del sistema a una sola capa, se impone un límite en la complejidad del sistema en general, promoviendo así la independencia de los sustratos.

El enfoque RESTful de los servicios web surge como una alternativa popular por su naturaleza liviana y a la capacidad de transmitir datos directamente sobre HTTP. Los clientes se implementan usando una gran variedad de lenguajes como programas Java, Perl, Ruby, Python, PHP y Javascript (inclusive Ajax). Generalmente se accede a los servicios web RESTful a través de un cliente automatizado o una aplicación que actúa en representación del usuario. Sin embargo, la simplicidad de estos servicios posibilita la interacción humana directa pudiéndose construir una URL GET con el navegador Web y leer el contenido devuelto.

En un servicio web de estilo REST, cada recurso tiene una dirección. Los recursos en sí son los objetivos de las llamadas de los métodos y todos los recursos comparten una misma lista de métodos. Los métodos son estándar; se soportan los métodos HTTP GET, POST, PUT, DELETE, y, pueden soportarse los métodos HEADER y OPTIONS.

- **Diagrama del patrón MVC**

Es un patrón de arquitectura de software, que separa los datos y la lógica de negocio de una aplicación de la interfaz de usuario y el módulo encargado de gestionar los eventos y las comunicaciones. Para ello MVC propone la construcción de tres componentes distintos que son el modelo, la vista y el controlador, es decir, por un lado, define componentes para la representación de la información, y por otro lado para la interacción del usuario.

- **El Modelo:** Es la representación de la información con la cual el sistema opera, por lo tanto, gestiona todos los accesos a dicha información, consultas como actualizaciones, implementando también los privilegios de acceso que se hayan descrito en las especificaciones de la aplicación (lógica de negocio). Envía a la 'vista' aquella parte de la información que en cada momento se le solicita para que sea mostrada. Las peticiones de acceso o manipulación de información llegan al 'modelo' a través del 'controlador'.
- **El Controlador:** Responde a eventos e invoca peticiones al 'modelo' cuando se hace alguna solicitud sobre la información. También puede enviar comandos a su 'vista' asociada si se solicita un cambio en la forma en que se presenta el 'modelo' (por ejemplo, desplazamiento o scroll por un documento o por los diferentes registros de una base de datos), por tanto, se podría decir que el 'controlador' hace de intermediario entre la 'vista' y el 'modelo'.
- **La Vista:** Presenta el 'modelo' (información y lógica de negocio) en un formato adecuado para interactuar (usualmente la interfaz de usuario) por tanto requiere de dicho 'modelo' la información que debe representar como salida.

1.6.4 Codificación

Esta etapa consiste en el desarrollo de la solución planteada en la etapa de diseño, en pocas palabras es la traducción del diseño a código, es aquí donde se definen el lenguaje de programación, los estándares de programación y las tecnologías adecuadas a utilizar para el cumplimiento de los requerimientos del proyecto en el tiempo establecido.

Técnicas de codificación

- **Programación Funcional**

Es un paradigma de programación declarativa basado en el uso de funciones matemáticas, en contraste con la programación imperativa, que enfatiza los cambios de estado mediante la mutación de variables.

En el paradigma de la programación funcional, un programa se considera una función matemática, la cual describe una relación entre una entrada y una salida y donde el concepto de estado o variable se elimina completamente, en programación funcional pura no existen variables, sólo existen constantes, parámetros y valores, aun cuando en la práctica la mayoría de los lenguajes de programación funcionales no son puros pues retienen algunas nociones de variables y asignaciones.

Como no hay variables ni asignación, tampoco existen los ciclos al estilo de los de los lenguajes tradicionales ya que los mismos trabajan con una variable de control que se va reasignando.

La programación funcional se basa en las siguientes técnicas:

- **Funciones puras.** Las funciones puramente funcionales no tienen efectos secundarios (memoria o E/S). Esto significa que las funciones puras tienen varias propiedades útiles, muchas de las cuales pueden ser utilizadas para optimizar el código, en programación las funciones puras son aquellas que cumplen con dos requisitos básicos:
 - Dado unos parámetros de entrada de idéntico valor, la función siempre devolverá el mismo resultado.
 - El cómputo de la función, su lógica, no implica ningún efecto observable colateral fuera de ella

Entre las ventajas de la utilización de funciones puras se pueden mencionar las siguientes:

- Si no se utiliza el resultado de una expresión pura, se puede eliminar sin afectar a otras expresiones.
 - Si una función pura se llama con parámetros que no causan efectos secundarios, el resultado es constante con respecto a la lista de parámetros (a veces llamada transparencia referencial), es decir, si la función pura se llama de nuevo con los mismos parámetros, el mismo resultado será devuelto (esto puede habilitar las optimizaciones de almacenamiento en caché).
 - Si no hay una dependencia de datos entre dos expresiones puras, entonces su orden puede ser invertido, o pueden llevarse a cabo en paralelo y que no pueda interferir con los otros.
 - Si el lenguaje no permite efectos secundarios, entonces cualquier estrategia de evaluación se puede utilizar, lo que da la libertad al compilador para reordenar o combinar la evaluación de expresiones en un programa (por ejemplo, usando la poda).
 - Las Funciones Puras, por definición, son fáciles de testear. Al no requerir de un contexto, podemos concentrarnos en evaluar el tándem entrada/salida de forma aislada
-
- **Funciones de orden superior.** Una de las principales características de los lenguajes funcionales más avanzados es que incluyen lo que se denomina orden superior. Básicamente, esto significa que las funciones podemos usarlas de la misma forma que el resto de tipos. Así, podemos pasar funciones como parámetros, devolver una función como resultado de otra, almacenar funciones en estructuras de datos, etc., para definir funciones de orden superior se utilizan exactamente las mismas técnicas que para definir cualquier otro tipo de función. La única diferencia está en que los parámetros pueden ser funciones.

Las funciones de orden superior permiten la aplicación parcial, una técnica en la que se aplica una función a sus argumentos uno a la vez, con cada aplicación devolver una nueva función que acepta el siguiente argumento.

Esto le permite a uno expresar, por ejemplo, la función sucesora como el operador de suma aplicada parcialmente al número natural uno.

- **Composición de funciones** Es un acto o mecanismo para combinar funciones simples en complejas. La composición de funciones es el acto de dirigir el resultado de una función, a la entrada de otra, creando una nueva función.

Tal como las composiciones de funciones en matemática, el resultado de cada función es pasada como argumento a la siguiente y el resultado de la última función es el de todas, se debe tomar en cuenta lo siguiente al momento de utilizar la composición de funciones:

- Las funciones se ejecutan de derecha a izquierda según se pasen a la función compone.
 - El tipo de dato que resulta de una función debe ser el mismo que acepta como entrada la siguiente función.
 - Una clave de la composición es que podemos agrupar las funciones como deseemos e ir componiendo funciones cada vez más complejas.
- **Asincronía** Establece la posibilidad de hacer que algunas operaciones devuelvan el control al programa llamante antes de que hayan terminado mientras siguen operando en segundo plano. Esto agiliza el proceso de ejecución y en general permite aumentar la escalabilidad, existen diversos modelos de asincronía:
 - **Modelo de paso de continuadores** Cada función recibe información acerca de cómo debe tratar el resultado –de éxito o error– de cada operación. Requiere orden superior.
 - **Modelo de eventos** Se utiliza una arquitectura dirigida por eventos que permite a las operaciones no bloqueantes informar de su terminación mediante señales de éxito o fracaso. Requiere correlación para sincronizar.
 - **Modelo de promesas** Se razona con los valores de retorno de las operaciones no bloqueantes de manera independiente del momento del tiempo en que dichos valores –de éxito o fallo– se obtengan.
 - **Modelo de generadores** Se utilizan generadores para devolver temporalmente el control al programa llamante y retornar en un momento posterior a la rutina restaurando el estado en el punto que se abandonó su ejecución.

- **Inmutabilidad** Es la propiedad que permite que algo no pueda ser modificado una vez creado, en el contexto de la programación, una variable es inmutable cuando su valor no se puede modificar, y un objeto lo es cuando su estado no puede ser actualizado tras la creación del objeto, esto nos permite asegurarnos que nuestro objeto no se modifica en lugares inesperados, afectando con ello la ejecución de nuestro programa. La inmutabilidad, en general, hace nuestro código mucho más predecible, porque somos más conscientes de dónde se producen los cambios de estado. Su mayor uso se da en aplicaciones multihilo, donde la inmutabilidad simplifica mucho el tratamiento de la concurrencia.

- **Arquitectura basada en componentes** Una arquitectura basada en componentes describe una aproximación de ingeniería de software al diseño y desarrollo de un sistema. Esta arquitectura se enfoca en la descomposición del diseño en componentes funcionales o lógicos que expongan interfaces de comunicación bien definidas. Esto provee un nivel de abstracción mayor que los principios de orientación por objetos y no se enfoca en asuntos específicos de los objetos como los protocolos de comunicación y la forma como se comparte el estado.

La arquitectura basada en componentes tiene las siguientes características:

- Es un estilo de diseño para aplicaciones compuestas de componentes individuales.
- Pone énfasis en la descomposición del sistema en componentes lógicos o funcionales que tienen interfaces bien definidas.
- Define una aproximación de diseño que usa componentes discretos, los que se comunican a través de interfaces que contienen métodos, eventos y propiedades.

1.6.5 Pruebas

Se realizarán pruebas para garantizar que se ha codificado la solución de acuerdo con las especificaciones de los usuarios, para la realización de las pruebas se utilizarán datos precargados para las capas y se ingresaran datos ficticios.

Se realizarán pruebas individuales para verificar el funcionamiento por separado de los módulos del sistema y pruebas integrales para verificar la integración de los diferentes módulos del sistema y su funcionamiento en general.

En caso de existir fallos durante las pruebas se realizarán las observaciones para realizar las correcciones respectivas.

Metodología de las pruebas

Pruebas del tipo Caja Negra: se prueba cada una de las funciones para determinar si son operativas, que la entrada se acepta de forma adecuada, que se produce un resultado correcto y que la integridad de la información se mantiene.

Los errores que intenta encontrar este método son los siguientes:

- Funciones incorrectas o ausentes.
- Errores en tablas de la Base de datos.
- Errores de inicialización de variables y construcción de objetos.
- Prueba de Documentación y Ayuda, se examina el documento para comprobar su claridad, utilizando el sistema junto con la documentación.

- Prueba de Validación y Verificación, se utiliza con el objeto de conocer si el software funciona de acuerdo a los requerimientos del usuario y cumple correctamente con una función específica.
- Prueba de Módulos, se prueba la interfaz del módulo para asegurar que la información fluye en forma adecuada.
- Prueba de Seguridad, se verifican los mecanismos de protección incorporados en el sistema, de accesos no permitidos, de tal forma de resguardar la información que contiene el sistema.

Diseño de Pruebas

El desarrollo de pruebas se realizará en dos fases:

La primera fase es la realización de las pruebas parciales, las cuales se realizarán durante la programación del software. Cada módulo programado será evaluado para comprobar que el objetivo del diseño se cumpla.

La segunda fase consiste en una prueba integrada del sistema, en la que se realizarán operaciones que involucran la interrelación de los módulos del sistema.

Para que el sistema se dé por concluido se deben haber aprobado los siguientes puntos durante la realización de las pruebas:

Funcional: Prueba desde el punto de vista de los requerimientos funcionales.

De Sistema: Prueba desde el punto de vista de los niveles de calidad del sistema y de desempeño.

De Integración: Prueba de interfaces.

De Aceptación Técnica: Prueba de manejo de condiciones extremas.

1.6.6 Documentación

La etapa de documentación contempla tanto la documentación interna del software como la documentación externa referente al sistema, dentro de la documentación externa se tiene el manual de usuario, manual de instalación y manual técnico.

1.6.7 Implementación

El Plan de Implementación se ocupa de planificar la forma en que se implantará el nuevo sistema de manera que se adapte a las características y necesidades de los usuarios y se alcance el éxito del proyecto.

El plan incluye los siguientes puntos:

- Marco de referencia de la implementación.
- Recursos necesarios para llevar a cabo el plan de implementación, divididos en recursos informáticos y recurso humano.
- Actividades a realizarse para la implementación, así un detalle de los riesgos posibles durante la etapa.

CAPÍTULO II: Situación Actual

2.1 Objetivo de la Situación Actual

Describir la situación actual del proceso de evaluación de edificios, para realizar un estudio en el cual se planteen soluciones informáticas para los elementos que constituyen la evaluación de edificios; y como resultado realizar el análisis y determinación de requerimientos para las oportunidades de mejora detectadas.

2.2 Descripción de la Situación Actual

2.2.1 Inspecciones presísmicas.

2.2.1.1 Recursos Humanos

Se crean brigadas de inspección conformadas por el personal de inspección, los cuales serán los encargados directos de recopilar la información de las edificaciones (personal de campo), y el personal de logística, quienes serán los encargados de coordinar los estudios en las edificaciones y la cantidad de los mismos, así como del tratamiento de la información y el establecimiento de los resultados.

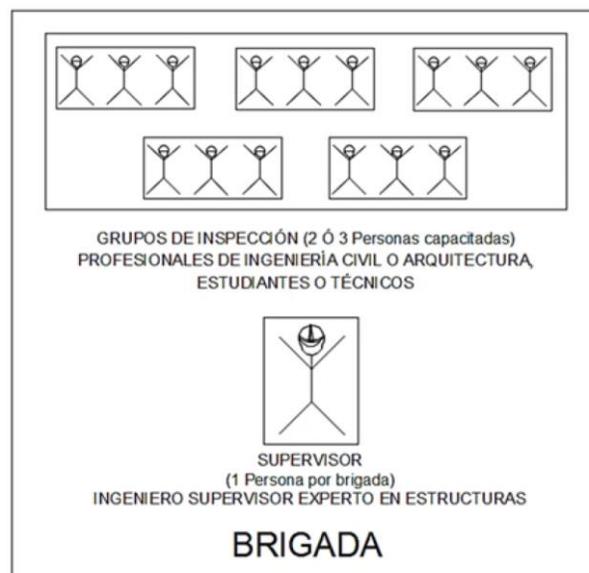


Ilustración 5. Organización de grupos de inspección.

Alvarado, J. & Javier, O. (2017). *PROPUESTA METODOLÓGICA PARA LA CREACIÓN DE UN SISTEMA DE GESTIÓN DE INFORMACIÓN DE ESTUDIOS DE VULNERABILIDAD ESTRUCTURAL*. Universidad de El Salvador, El Salvador.

Personal de Inspección

Para la toma directa de información en las inspecciones de campo es necesario conformar pequeños grupos de inspección, los cuales podrán estar conformados por 2 o 3 personas capacitadas que pueden ser profesionales de Ingeniería Civil o Arquitectura; técnicos o estudiantes de algunas de esas ramas. El personal de inspección deberá estar bajo supervisión directa del personal logístico.

Personal Logístico

Es el encargado de coordinar todos los grupos de inspección, las actividades y las áreas urbanas a inspeccionar durante cada jornada. Debe estar compuesto de al menos un Ingeniero supervisor experto en estructuras con amplia experiencia en el campo, que tenga un amplio conocimiento en el diagnóstico de estructuras y brinde soporte técnico a los inspectores de campo.

La cantidad establecida de personal para la inspección de campo podrá variar a consideración del criterio de los profesionales o consultores a cargo del proyecto, los cuales definirán cuantos grupos de inspección serán necesarios, de acuerdo a la cantidad de estructuras a evaluar, la extensión del área de estudio, el tiempo y los recursos disponibles. De igual manera, la cantidad del personal de logística debe ser definida en base a las condiciones ya mencionadas.

2.2.1.2 Recursos materiales

Los recursos materiales deben cumplir tres funciones esenciales: salvaguardar la seguridad de los inspectores de campo, realizar las mediciones en la estructura y servir de medios para la toma de datos. Cada grupo de inspección debe contar con los recursos que se mencionan a continuación, a menos que estos sean prescindibles.

Los recursos necesarios para salvaguardar la seguridad de las personas son:

- Chalecos.
- Cascos.
- Botas de seguridad.
- Tapabocas o mascarillas.
- Gafete de Identificación.

Todos los recursos mencionados en la lista anterior pueden ser prescindibles dependiendo de las condiciones en que se encuentren las estructuras a evaluar, a excepción de los gafetes de identificación, en los cuales se debe plasmar el nombre del proyecto (estudio), la organización a cargo, la institución contratante y la identificación de los inspectores de campo.

Recursos necesarios para realizar mediciones de campo:

- Escalera.
- Plomada.
- Cinta métrica.
- Binoculares.
- Nivel de mano.
- Marcador o lápices de colores.
- Linterna.

Recursos para la toma de datos:

- Cuaderno de anotaciones.
- Fichas de campo.
- Tabla.
- Cámaras fotográficas.
- Lápiz o lapicero.
- Lápices de colores.
- Grabadora.

2.2.1.3 Capacitación del personal para las inspecciones

Antes de iniciar los sondeos de campo, es necesario que el personal de logística capacite adecuadamente al personal para las inspecciones y les brinde además todos los recursos materiales que se consideren necesarios.

La capacitación debe estar basada en:

- La forma en que se debe tratar con los encargados, ocupantes o titulares de la edificación.
- La forma correcta en que deben utilizarse los recursos materiales.
- El entendimiento de todos los criterios técnicos en los cuales se basa la obtención del índice.
- La forma correcta en que deben realizarse las inspecciones.
- La forma correcta en que debe rellenarse el formulario de inspección.
- El procedimiento para la obtención del índice.
- El tiempo que debe tardarse cada inspección.
- La forma en que debe notificarse a los encargados, ocupantes o titulares de la edificación, del estado de seguridad de la misma, su calificación y el porqué de dicha calificación.

2.2.1.4 Seguridad para el personal de campo

Es responsabilidad del personal logístico y de la institución a cargo del estudio de vulnerabilidad el coordinar con las instituciones públicas para la colaboración conjunta en la realización de las inspecciones de campo, a manera de asegurar la seguridad e integridad de las personas involucradas

en el estudio, así como de los equipos y herramientas que se utilizarán. Cuando sea necesario, se deberá contactar con la Policía Nacional Civil o con el cuerpo de agentes de seguridad de la alcaldía del municipio o departamento en cuestión, en el que se realicen los estudios de vulnerabilidad.

2.2.1.5 Distribución del personal de inspección y programación de las visitas de campo

Antes de iniciar las inspecciones de campo, el personal de logística debe establecer los grupos de inspección necesarios para cubrir el área de estudio, así como el periodo de tiempo que durará dicho estudio.

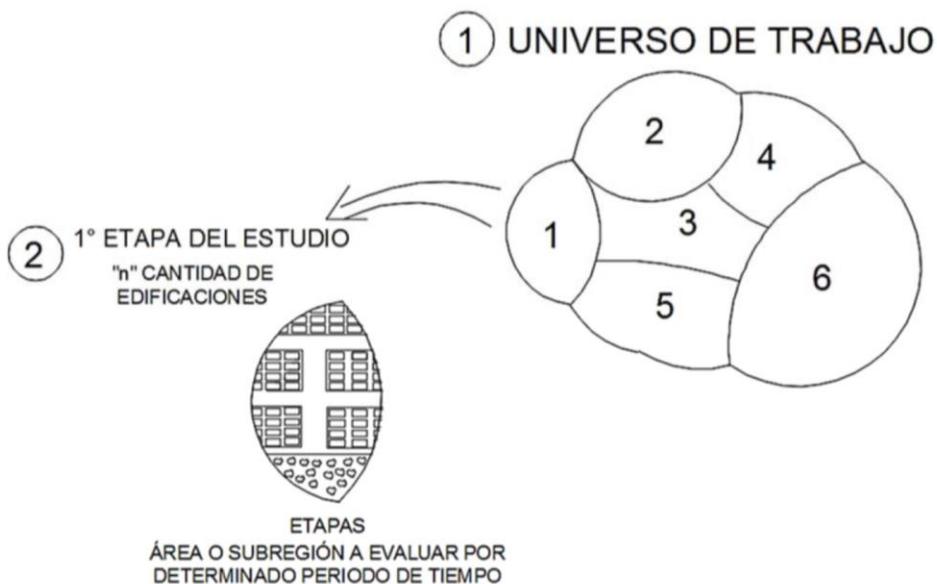


Ilustración 6. Representación de subregiones a evaluar.

Alvarado, J. & Javier, O. (2017). *PROPUESTA METODOLÓGICA PARA LA CREACIÓN DE UN SISTEMA DE GESTIÓN DE INFORMACIÓN DE ESTUDIOS DE VULNERABILIDAD ESTRUCTURAL*. Universidad de El Salvador, El Salvador.

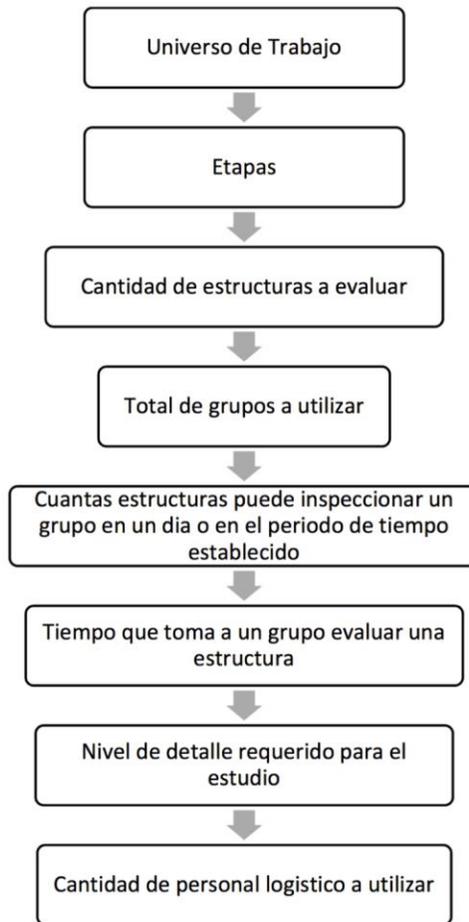


Ilustración 7. Organización durante evaluaciones.

Alvarado, J. & Javier, O. (2017). PROPUESTA METODOLÓGICA PARA LA CREACIÓN DE UN SISTEMA DE GESTIÓN DE INFORMACIÓN DE ESTUDIOS DE VULNERABILIDAD ESTRUCTURAL. Universidad de El Salvador, El Salvador.

Inicialmente, se debe definir el universo de trabajo, que comprende la región urbana sobre la cual se pretende realizar los estudios de vulnerabilidad y en un posible futuro, estudios de riesgo. Dependiendo de la extensión de dicha área de estudio, esta podrá dividirse en etapas o subregiones para su estudio. Cada etapa representará una subregión específica del universo de trabajo, que podrá ser evaluada por día, semana o cualquier otro periodo de tiempo que se estipule conveniente. Para cada etapa se define la cantidad de estructuras a evaluar, que comprenderá el total de edificaciones emplazadas dentro de la subregión definida para la etapa correspondiente. Se podrá definir entonces, el total de grupos de inspección a utilizar para una o todas las etapas del estudio. Esto en base al conocimiento de cuantas estructuras puede inspeccionar un grupo en un día o periodo de tiempo que se haya definido para cada etapa, que dependerá del tiempo que le tome a un grupo evaluar una estructura, el cual podrá ampliarse o reducirse, de acuerdo al nivel de detalle requerido en el estudio. El formulario podrá ser complementado con detalles adicionales, tales como: fotografías del diagnóstico del edificio, entrevistas a los ocupantes de la edificación, hojas de observaciones acerca del estado del edificio e inclusive aspectos adicionales a evaluar o elementos para la realización de inventarios de componentes. Finalmente, una vez establecida la cantidad de grupos a utilizar por etapa es posible definir la cantidad de personal logístico necesario para el estudio.

La programación de las visitas de campo, así como de las etapas de evaluación y el tiempo que durarán éstas, dependerá de los recursos disponibles para el estudio: económicos, humanos, materiales, tiempo, entre otros.

2.2.1.6 Realización de la inspección

Una vez se hayan contemplado y organizado adecuadamente todos los aspectos mencionados en los apartados anteriores se procede a la realización de las inspecciones de campo.

Preparativos

Antes de iniciar las inspecciones, los supervisores o encargados deben reunir a todo el personal de campo (antes de llegar al lugar donde se realizará el estudio) y tratar los siguientes puntos:

- Verificar la cantidad de personal de campo que realizará las inspecciones, así como la conformación de los grupos de inspección.
- Presentar a cada supervisor o experto ante los grupos de los cuales será responsable y hacer de conocimiento de estos últimos el área o conjunto de edificaciones que deberán evaluar.
- Establecer el horario de trabajo, los tiempos de descanso y el tiempo que deberá durar cada visita.
- Verificar que todos los grupos posean los recursos materiales necesarios para realizar las inspecciones, así como también la cantidad y la disponibilidad de los recursos compartidos.

Metodología de inspección

La secuencia lógica de cada una de las fases o etapas de la inspección se muestra en el siguiente esquema:

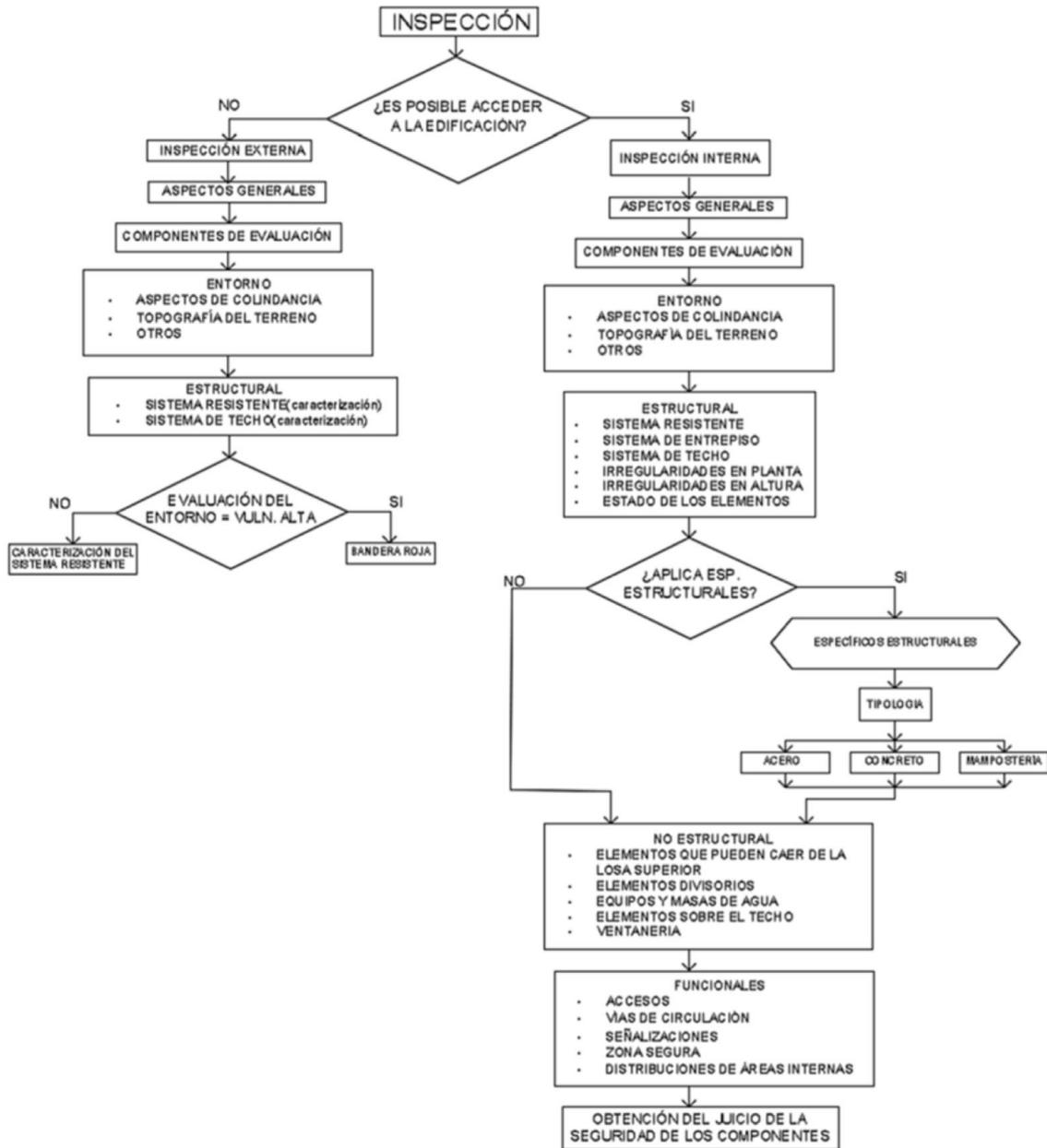


Ilustración 8. Flujo de trabajo.

Alvarado, J. & Javier, O. (2017). PROPUESTA METODOLÓGICA PARA LA CREACIÓN DE UN SISTEMA DE GESTIÓN DE INFORMACIÓN DE ESTUDIOS DE VULNERABILIDAD ESTRUCTURAL. Universidad de El Salvador, El Salvador.

Consentimiento del encargado u ocupante

Para poder proceder a la realización de la evaluación, es necesario dialogar con la persona o las personas a cargo del edificio, a fin de pedir su consentimiento para ingresar a la edificación. Este aspecto definirá el tipo de inspección a realizar en el edificio, inspección interna (simple) o inspección externa. Si el encargado brinda su consentimiento para la realización de la inspección, se deberá proceder a la inspección interna; En cambio, si éste no concede el permiso de entrar a la edificación, se procederá entonces a la inspección externa.

Inspecciones según consentimiento

Ante la posibilidad de que el equipo de campo no pueda ingresar a una o varias edificaciones, se establecen dos formas de realizar la inspección: Una inspección interna o una inspección externa.

La inspección interna es el tipo de inspección que debe realizarse en el caso más favorable, cuando el equipo de trabajo obtenga el permiso de ingresar a la edificación. En ella se deberán evaluar todos los parámetros del formulario que sean posibles, que puedan caracterizarse a simple vista o que requieran de alguna acción adicional para poder apreciarlos, siempre y cuando estos no dañen ni alteren la propiedad o bienes del ocupante o titular de la edificación. Al mismo tiempo, se deberá cuidar que dichas acciones no causen efecto nocivo alguno sobre la estructura en sí.

La inspección externa es aquella que deberá realizarse en caso de obtener una respuesta negativa para acceder a la edificación. Debido a que el equipo no podrá adentrarse en las instalaciones, se deberán evaluar únicamente los aspectos relacionados al entorno y los que sean posibles de caracterizar desde afuera de la edificación.

Secuencia de evaluación

En cualquiera de los dos escenarios posibles, a los que deberá enfrentarse el equipo de evaluación, se establece un orden común para la secuencia de evaluación de las clases de parámetros:

1. Aspectos generales.
2. Componentes de evaluación:
 - a. Entorno.
 - b. Estructural.
 - c. No estructural.
 - d. Funcional.

Los componentes específicos estructurales al estar relacionados con un determinado tipo de estructura serán procedentes únicamente para la inspección interna y serán evaluados en conjunto con los componentes estructurales, siempre y cuando sean aplicables (su evaluación corresponda a la estructura evaluada).

2.2.2 Inspecciones post sísmicas.

2.2.2.1 Recursos humanos.

Se consideran un evaluador y un auxiliar para la conformación de las comisiones para la inspección de daños en edificios, sin embargo, se debe considerar más personal para la inspección si la dimensión de la edificación así lo requiera.

Se tomará como evaluador un profesional relacionado con el sector de la construcción de edificaciones, como ingenieros civiles, arquitectos o técnicos en obras civiles, preferiblemente profesionales con 2 o más años de experiencia en diseño estructural o en construcción, con el fin de poder reconocer con facilidad los grados de daños estructurales y no estructurales.

Se considerarán como auxiliares a los profesionales con menos de 2 años de experiencia, estudiantes egresados o de último año de ingeniería o arquitectura.

Además de las personas que conformarán la comisión, se requerirá un supervisor y un coordinador, los cuales deberán tener conocimientos sobre logística. El supervisor será el encargado de coordinar la labor de las comisiones en una zona determinada. El coordinador será el encargado de coordinar la labor de los supervisores.

Funciones del personal

Función de los coordinadores:

- Los deberes del coordinador son entregar los paquetes de formularios a los supervisores y recibirlos una vez hayan sido revisados y clasificados por los diferentes supervisores en su área.
- Realizar un informe integrado de la localidad o zona.
- Arreglar todo lo pertinente al transporte, alimentación y acomodo del personal.
- Reportar a las autoridades pertinentes las acciones necesarias a ejecutar en su localidad como la protección de vías de tránsito, la remoción de escombros o peligros locales, el rescate de víctimas, la evacuación de edificaciones, etc.

Función de los supervisores:

- Distribuir el personal asignado a la zona.
- Repartir el material correspondiente.
- Verificar y asesorar el correcto y completo llenado de los formularios.
- Preparar las rutas de trabajo y los reportes.
- Es el responsable de la labor y seguridad de la comisión.

Función de los evaluadores:

- Evaluar de manera objetiva y profesional.
- Realizar el llenado del formulario.
- Definir el color de bandera según la puntuación obtenida en el formulario.

- Preparar los informes finales de cada edificación para ser enviados a las instituciones encargadas de su registro y análisis.

Funciones de auxiliares:

- Tomar fotografías de la edificación.
- Dibujar croquis de ubicación y esquemas de planta y elevación.
- Marcar y señalizar la edificación.
- Corroborar puntuación obtenida en el formulario.
- Colaborar en la evaluación de daños y elaboración de los informes.

Organización de comisiones.

La organización básica para las inspecciones debe llevarse a cabo en cada localidad siguiendo el Plan de Contingencias, en el cual se debe especificar el número de comisiones por sector, partiendo de que dicha labor debe desarrollarse en un corto plazo después de un terremoto.

Los formularios serán repartidos a las comisiones por un coordinador, que además será el encargado de recoger los formularios ya utilizados. Las comisiones de evaluación deben estar previamente asignadas a una zona por un supervisor y haber recibido la capacitación.

2.2.2.2 Recursos materiales

- Para los procedimientos de evaluación se recomienda contar con los siguientes elementos:
- Planos de la zona a inspeccionar.
- Manual de evaluación postsísmica.
- Formularios de inspección, avisos de clasificación o pinturas, grapas, cinta o brocha.
- Cinta con la inscripción PELIGRO para restringir el acceso a áreas inseguras.
- Libreta de notas, lápiz o bolígrafo.
- Linterna y baterías extra.
- Cámara fotográfica.
- Cinta métrica de 10 a 30 m.
- Nivel, destornillador o cincel ligero.
- Radio o teléfono celular.
- Nombres y números telefónicos de los coordinadores y supervisores de evaluación y de las entidades del sistema de prevención y atención de desastres
- Calculadora
- GPS
- Botiquín

2.2.2.3 Capacitación para el personal.

Todo el personal utilizado para la inspección de las edificaciones debe tener previamente una capacitación sobre:

- La forma de gestionar los formularios.
- Los conceptos utilizados en el formulario.
- La forma correcta de realizar la inspección externa e interna de daños.
- El procedimiento para la determinación de los índices de daño.

2.2.2.4 Seguridad para el personal de campo

- Identificación personal.
- Casco de seguridad.
- Botas de seguridad.
- Lentes de protección.
- Chaleco refractivo.

2.2.2.5 Realización de las inspecciones

Categoría de ocupación.

Las inspecciones post sismo tienen que realizarse de acuerdo al uso que estas tienen, la Norma Técnica para Diseño por Sismo, hace una clasificación a las edificaciones de la siguiente manera:

- Establecimientos Esenciales o Peligrosos: Son aquellas edificaciones que son necesarias para atender la emergencia después de un sismo.
- Edificios de ocupación especial: Son edificios que tengan niveles altos de ocupación o que requiera su uso inmediatamente después de un sismo.
- Edificios de ocupación normal: Son edificios con niveles bajos de ocupación.

Esta clasificación es el punto de partida para establecerse las prioridades de inspección.

Inspección simplificada

El procedimiento de la inspección simplificada se basa en observar las condiciones y características de los daños que individual o colectivamente sean suficientes para concluir que la edificación se encuentra en capacidad de tener un uso normal (habitable) o si el uso deber ser restringido o anulado (seguridad en duda o insegura). Esta conclusión depende de la ponderación adecuada de diversos aspectos relacionados a la capacidad de resistencia y estabilidad de la estructura.

La inspección simplificada puede derivarse en dos etapas:

Inspección externa: Durante la inspección externa se deberá determinar si la edificación posee colapso total, parcial o se encuentra sin riesgo de colapso. Cuando desde el exterior se determine que existe un colapso total o inminente, por seguridad de la comisión no se debe realizar la inspección interna, y se clasifica como bandera roja finalizando la inspección.

Inspección interna (Daños estructurales): Cuando se determina por medio de la inspección externa que no existe riesgo de colapso y se tenga el permiso del propietario o encargado, se procederá a realizar la inspección interna.

El procedimiento de la inspección simplificada se basa en tres tareas básicas:

Inspección visual:

El procedimiento de inspección debe iniciar con un reconocimiento del entorno y del exterior de la edificación, para determinar si es posible realizar la inspección interna. Se consideran tres etapas durante la inspección visual que deben ser realizadas:

- **Inspección del entorno:** consiste en observar el suelo alrededor de la edificación, para determinar la presencia de grietas, hundimientos, deslizamientos o cualquier anomalía del terreno. Se debe examinar el exterior de la edificación, llenar el formulario de evaluación con la identificación de la edificación, evaluar la calidad de la construcción, irregularidades y otros aspectos preexistentes evaluados desde el exterior.
- **Inspección interna:** Descripción cualitativa de las condiciones de seguridad de la edificación con base a los daños en elementos estructurales y no estructurales evaluados en el interior. En esta etapa puede auxiliarse con los planos estructurales de la edificación para facilitar la evaluación.
- **Esquemas o fotos:** Esquematación de los daños observados para una mejor comprensión en la siguiente etapa.

Evaluación objetiva:

Consiste en la ponderación de los datos obtenidos en la inspección visual. La ponderación debe hacerse de acuerdo al criterio de los expertos o personal previamente capacitado. En la evaluación del sistema estructural se debe analizar el grado de daño de los diferentes elementos estructurales de acuerdo la importancia de cada elemento en la estabilidad del edificio.

Calificación del edificio:

El estado de la edificación después de un sismo se representa mediante la asignación de un color de bandera de advertencia obtenida con los resultados de la fase de evaluación objetiva. En la calificación además de definir la bandera de advertencia se debe incluir el significado de ésta.

Cuando la clasificación de la edificación es naranja o roja generalmente será necesario realizar una evaluación detallada para determinar la capacidad de resistencia de los elementos y determinar las acciones posteriores de rehabilitación o demolición.

Inspección detallada

La inspección detallada determina el estado o la condición del edificio de una manera más detallada y rigurosa. Durante la revisión de los daños en los elementos estructurales, la evaluación no debe

necesariamente realizarse en toda la edificación sino en el nivel o niveles más dañados. De igual manera, se deben evaluar los elementos no estructurales que presenten un grado de daño importante o tengan el riesgo de caída o volcamiento.

La inspección detallada cubre un conjunto de acciones que deben seguirse de manera secuencial y programada, las acciones que deben realizarse son las siguientes:

- Investigación documental: recopilación del diseño arquitectónico, diseño estructural, estudio geotécnico o de suelos, memoria de cálculos, entre otros.
- Inspección visual detallada: se deben realizar anotaciones de las áreas afectadas, longitud del daño, tamaño de fisuras y otras características importantes de los daños.
- Levantamiento gráfico de daños: se trata de esquematizar o dibujar los daños en los elementos.
- Recuento fotográfico: las fotografías deben ir de acuerdo al levantamiento, éstas sustentarán el levantamiento gráfico de daños.
- Planeamiento y definición de ensayos: consiste en la selección del tipo de pruebas y ensayos que deben realizarse para proceder a formular la metodología de reparación o rehabilitación.
- Diagnóstico de daños: con base a los resultados de los ensayos, se determinará las zonas que presenten un alto nivel de riesgo para los ocupantes.
- Informe de la inspección.

Metodología de inspección.

Para realizar una inspección de daños se recomienda seguir los siguientes pasos:

1. Obtener la información de ítems de aspecto común desde el exterior de la edificación e información proporcionada por la persona de contacto.
2. Obtener la información de la descripción general de la edificación, desde el exterior de la edificación e información proporcionada por la persona de contacto.
3. Dibujar croquis de ubicación y esquema de elevación y planta del edificio.
4. Dentro de la inspección externa evaluar los ítems de aspectos del entorno: inclinación del edificio, falla en cimentación y la existencia de problemas geotécnicos.
5. Determinar el Índice de Condiciones Exteriores y del Suelo (IDCES).
6. Por medio de una inspección interna evaluar los ítems de aspecto estructural: sistema estructural y daño en el mismo. Se debe analizar grado de daño de los diferentes elementos estructurales de acuerdo con el tipo de sistema estructural y establecer el porcentaje de elementos afectados en el piso con mayores daños.
7. Determinar el Índice de Daño Estructural (IDE).
8. Examinar la seguridad de elementos no estructurales, identificar los grados de daño existentes en cielos rasos, paredes de fachada o divisorias, escaleras, luminarias e instalaciones y establecer el porcentaje de elementos afectados en el piso con mayores daños.
9. Determinar el Índice de Daño No Estructural (IDNE).
10. Determinar el Índice de daño (ID) del edificio por medio del IDE y IDNE.
11. Determinar el Estado del edificio por medio de ID y IDCES, y determinar la clasificación de habitabilidad.
12. Anotar la información de los inspectores y la fecha y hora de la inspección.

13. Llenar los avisos para clasificación de las edificaciones y colocarlos en cada una de las entradas.
14. Explicar verbalmente el significado de la clasificación a la persona de contacto de la edificación. También se debe restringir el acceso a las áreas determinadas como inseguras, colocando algún tipo de barreras, por ejemplo, las cintas que lleven la inscripción de PELIGRO.
15. Notificar a los coordinadores o supervisores la clasificación de habitabilidad de la edificación para que se realicen los procedimientos que correspondan.

2.3 Descripción con enfoque de sistemas de la situación actual.

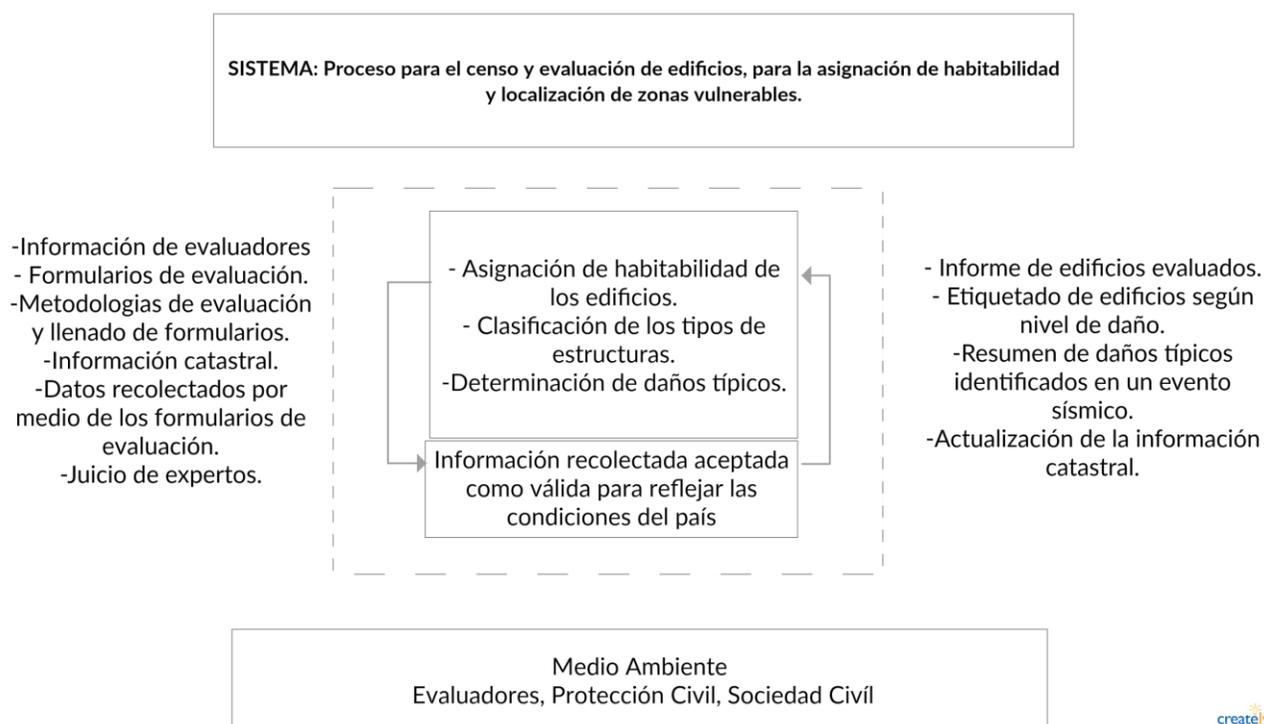


Ilustración 9. Diagrama de sistema, situación actual.

Salidas:

- Informe de edificios evaluados: Este tipo de informes se entrega a los coordinadores de las inspecciones y se archivan en la institución encargada de la jornada de inspección, normalmente ASIA.
- Etiquetado de edificios según nivel de daño: Posterior a las inspecciones los evaluadores generan una etiqueta que es colocada en un lugar visible de la edificación, además si el edificio tiene peligro de colapso o zonas a las cuales no se recomienda ingresar se tendrá que restringir el paso por medio de cintas con la inscripción de PELIGRO.

- Resumen de daños típicos identificados en un evento sísmico: Al realizar jornadas de evaluación de edificios después de un evento sísmico, los especialistas generan un resumen de daños típicos, al analizar los datos recolectados, esta información ayuda a actualizar normativas de construcción.
- Actualización de la información catastral: Luego de una evaluación se actualiza la información catastral debido a errores en inspecciones pasadas o colapsos causados por eventos sísmicos.

Entradas:

- Información de evaluadores: Esta información es recolectada durante las capacitaciones para poder tener los contactos a la hora de realizar jornadas de evaluación, la información de los evaluadores también es incluida como referencia cuando se llena un formulario de inspección.
- Formularios de evaluación: Para la realización de las inspecciones se trabaja con diversos formularios dependiendo del objetivo de la inspección y la institución que realizó la convocatoria.
- Metodologías de evaluación: Cada formulario posee un instructivo de la metodología de evaluación necesaria para el levantamiento de datos.
- Información catastral: Para el registro de los datos a las estructuras correspondientes los evaluadores necesitan información catastral con la cual emparejar los datos registrados.
- Datos recolectados por medio de los formularios de evaluación: Esta información es recolectada por medio de los formularios, que provee la institución que realiza la convocatoria, el llenado de dichos formularios se lleva a cabo manualmente.
- Juicio de expertos: Se necesita el juicio de expertos para la asignación correcta de habitabilidad en los edificios.

Procesos:

- Asignación de habitabilidad de los edificios: En base a los datos recolectados con los diferentes formularios los expertos deben de asignar un nivel de habitabilidad a cada edificio inspeccionado.
- Clasificación de los tipos de estructuras: Los expertos clasifican el tipo de estructura de las diferentes edificaciones censadas, de acuerdo a las características recolectadas en los formularios.
- Determinación de daños típicos: Posterior a las jornadas de evaluación de edificios los expertos agrupan los daños frecuentemente acaecidos, para determinar daños típicos.

Medio Ambiente:

- Evaluadores: personas capacitadas para poder realizar las inspecciones a las edificaciones; así también se consideran las instituciones encargadas de capacitar a dicho personal.

- Instituciones: Alcaldías, Protección Civil, OPAMSS, MOP y las gremiales afines; estas instituciones como entes interesados en el estado de los edificios, donde las personas que las integran hacen sugerencias sobre la consistencia y utilidad de los datos recolectados para determinar la habitabilidad de las estructuras.
- Sociedad civil: Usuarios de las edificaciones y principales beneficiarios del control de las condiciones de riesgo en dichas instalaciones.

Control:

- Información recolectada aceptada como válida para reflejar las condiciones del país:
 - Diferentes instituciones del área de la construcción y afines aceptan la información reflejada posterior a las inspecciones.
 - Los ingenieros estructurales o civiles internacionales ven la información recolectada durante las inspecciones como datos válidos.

Los niveles de riesgo se apegan a la realidad desde una visión objetiva tomando como marco de referencia las herramientas utilizadas por cada institución.

2.4 Análisis y diagnóstico del problema.

Para el análisis y diagnóstico del problema se han listado las diferentes situaciones que generan ineficiencia o errores en los procesos junto a las causas de dichas situaciones.

Listado de causas		
#	Causas	Detalle del problema
1	Las herramientas utilizadas por los evaluadores permiten el ingreso de información errónea.	Los resultados obtenidos son subjetivos.
2	Inadecuado control de los evaluadores calificados.	Dificultad para poder convocar a los evaluadores capacitados en el área estructural.
3	Alta cantidad de formularios y poco personal para su procesamiento.	<ul style="list-style-type: none"> - Desconocimiento del estado histórico de las edificaciones. - No existen estadísticas de las edificaciones. - Formularios de evaluación archivados.
4	Inexistencia de una institución oficial para el manejo de datos posterior a las evaluaciones.	Bases de datos fragmentadas o inexistentes.

Tabla 17: Análisis y diagnóstico del problema.

2.5 Conclusión de la situación actual

Los problemas identificados podrían solucionarse con la implementación de una herramienta que permita configurar cualquier tipo de instrumento de inspección de edificaciones y además sea confiable para la recolección de datos, ayudando a los evaluadores a realizar de manera más rápida su trabajo. Esta herramienta tiene que ser parte de un sistema informático que permita configurar cualquier formulario con su lógica de generación de banderas y permita centralizar y procesar la información en tiempo real, además esta data debe poder ser analizada con herramientas de procesamiento estadísticos.

CAPÍTULO III: Determinación de Requerimientos

3.1 Requerimientos informáticos

3.1.1 Requerimientos Funcionales

3.1.1.1 Generales

Código	Requerimiento
REQ-FG01	Permitir la creación de diferentes usuarios para dar soporte a los roles existentes.
REQ-FG02	Permitir crear instituciones y vincular representantes, para procurar un fácil canal de comunicación.
REQ-FG03	Crear canales de comunicación, tanto generales como particulares, para notificaciones y alertas relacionadas con estados de emergencia, jornadas de capacitaciones y jornadas de inspección.
REQ-FG04	Generar reportes parametrizados de los catálogos existentes.
REQ-FG05	Generar reportes parametrizados de las banderas obtenidas en una jornada de inspección.

Tabla 18: Requerimientos funcionales generales.

3.1.1.2 Módulos de inspección

Código	Requerimiento
REQ-FF01	Gestionar instrumentos de evaluación por medio de módulos, los cuales se aplicarán a cada edificación para poder realizar inspecciones.
REQ-FF02	Obtener resultados de la aplicación de módulos de evaluación.
REQ-FF03	Maquetar diferentes instrumentos de evaluación por medio de un constructor de formularios.
REQ-FF04	Cada elemento calculado de los instrumentos de inspección podrá configurarse con atributos de visibilidad para permitir o denegar su visualización.
REQ-FF05	Los instrumentos de evaluación deben soportar valores calculados u observados, y estos podrán ser a su vez cualitativos o cuantitativos
REQ-FF06	Configurar la lógica de cálculo de banderas de acuerdo a índices calculados.

Código	Requerimiento
REQ-FF07	Los módulos de inspección no podrán ser eliminados una vez hayan sido aplicados a un edificio.
REQ-FF08	A la hora de actualizar un módulo de inspección se guardará una copia completa, para no alterar los datos históricos.
REQ-FF9	Si se aplica un módulo de evaluación más de una vez a una edificación, se tiene que mostrar la última bandera obtenida.
REQ-FF10	Los instrumentos de inspección deberán poseer soporte para el almacenamiento de recursos multimedia.
REQ-FF11	Los instrumentos deberán poseer soporte para guardar datos georeferenciados.
REQ-FF12	El instrumento de inspección permitirá ser configurado con campos multivaluados, y se deben poder agregar elementos cualitativos y cuantitativos.
REQ-FF13	A cada edificación se le debe poder asociar responsables para informar de medidas obligatorias dado un resultado de inspección.

Tabla 19: Requerimientos funcionales del módulo de inspección.

3.1.1.3 Capacitaciones

Código	Requerimiento
REQ-FC01	Agregar, modificar, actualizar y eliminar capacitaciones.
REQ-FC02	Posterior a la creación o modificación de una capacitación, el sistema debe notificar a los usuarios sobre la disponibilidad de la misma.
REQ-FC03	Vincular capacitaciones recibidas a los usuarios que hayan participado en las correspondientes capacitaciones y hayan aprobado las mismas.

Tabla 20: Requerimientos funcionales de capacitaciones.

3.1.1.4 Inspecciones

Código	Requerimiento
REQ-FI01	Programar jornadas de inspecciones, describiendo recomendaciones a la hora de realizar la inspección, fecha de la jornada y área a cubrir.
REQ-FI02	Posterior a la programación o modificación de una jornada de inspección, el sistema debe notificar a los evaluadores.

Código	Requerimiento
REQ-FI03	Los evaluadores deben poder realizar inspecciones y visualizar en tiempo real las edificaciones que faltan por evaluar, así como las edificaciones que tienen una evaluación en curso.
REQ-FI04	Una vez se haya obtenido una bandera, la inspección no podrá ser editada ni eliminada.
REQ-FI05	Los administradores deben poder visualizar el detalle de las inspecciones realizadas y los evaluadores el detalle de las inspecciones en las que han participado.
REQ-FI06	Los evaluadores deben poder incluir asistentes a una inspección por un método de identificación como códigos QR.

Tabla 21: Requerimientos funcionales de inspecciones.

3.1.1.5 Mapas

Código	Requerimiento
REQ-FM01	Agregar, modificar, actualizar y eliminar datos geospaciales que correspondan a las edificaciones a evaluar.
REQ-FM02	Visualizar mapa con marcadores de las edificaciones inspeccionadas.
REQ-FM03	Proveer en el mapa vista satelital y vista de bloques y calles, las cuales podrán ser cambiadas en cualquier momento por el usuario.
REQ-FM04	Actualizar el mapa en tiempo real, cada vez que una edificación haya sido agregada o modificada.
REQ-FM05	Filtrar edificaciones por módulo de evaluación aplicado.
REQ-FM06	Visualizar en tiempo real por medio de indicadores de color, que edificaciones han sido evaluadas de acuerdo al módulo filtrado.
REQ-FM07	Visualizar en tiempo real por medio de indicadores de color, que edificaciones están siendo evaluadas actualmente de acuerdo a el módulo filtrado.
REQ-FM08	Visualizar en tiempo real por medio de indicadores de color, la bandera obtenida en cada edificación de acuerdo al módulo filtrado.
REQ-FM09	Desplegar el histórico de inspecciones de una edificación al seleccionarla desde el mapa.

Tabla 22: Requerimientos funcionales de mapas.

3.1.2 Requerimientos No Funcionales

3.1.2.1 Interfaz de usuario

El sistema deberá ser capaz de:

1. Brindar una interfaz intuitiva y de fácil navegación. La interfaz de la aplicación web y móvil deben proporcionar fácil entendimiento de cómo utilizar el sistema, tanto para usuarios acostumbrados a la manipulación de aplicaciones móviles y web, como para los que no. Proporcionar comodidad para el llenado de los formularios, así como la facilidad para navegar entre secciones ya llenadas para poder revisarlas o editarlas.
2. Proporcionar iconografía representativa en la aplicación web y móvil, con especial detalle en el formulario, esto para contextualizar al usuario en el significado los elementos del formulario a través de la representación gráfica, lo cual brinda un llenado más rápido.
3. Proporcionar interfaces móviles para realizar evaluaciones y gestiones básicas.

3.1.2.2 Requerimientos de Seguridad

Autenticación de Usuarios

El sistema deberá ser capaz de:

1. Validar que el sistema pueda ser utilizado solo por usuarios previamente registrados. Los usuarios podrán ser registrados por parte del administrador luego de que dichos usuarios hayan participado en una jornada de capacitación en la evaluación de estructuras y manipulación de la aplicación.
2. Cuando un usuario nuevo se registre, las credenciales deberán de ser entregadas por medio de su correo electrónico.
3. Autenticar por token con una expiración de 1 mes. El usuario utilizará identificador único y contraseña, almacenada en con una codificación SHA256, para la adquisición del token y respectivo ingreso al sistema.
4. Transferir la data cifrada. El servidor contará con un certificado TSL, brindando una comunicación cifrada entre las aplicaciones web y móvil, y el servidor.

Usuario Administrador

- El usuario administrador será capaz de gestionar todos los módulos del sistema; sin embargo, ningún usuario, ni siquiera el administrador, será capaz de alterar el flujo de replicación interna de los módulos dispuestos como formularios.

Cambio de Contraseñas

- El cambio de contraseña ostenta el ingresar la contraseña anterior para validar el cambio a la nueva contraseña.

Perfiles de Usuario para Acceso a Módulos

1. Administrador
2. Responsable de institución
3. Evaluador

3.1.2.3 Requerimientos de Desarrollo

A continuación, se detallan los recursos que deben de estar disponibles para el desarrollo de SICEVED, especificando aspectos legales, las técnicas que se utilizarán para el análisis, diseño, programación y documentación del sistema, el recurso humano necesario, la herramienta de desarrollo a utilizar para la programación, el Sistema Gestor de Base de Datos, y la plataforma operativa sobre la cual se ejecutará.

En el caso del Gestor de Base de Datos, Lenguaje de Desarrollo y Sistema Operativo en que se implantará el SICEVED, se ha considerado implementar Software Libre¹³, a solicitud del dueño del proyecto y por sugerencia del asesor de trabajo de graduación, solicitud que coincide con lo propuesto por el grupo que presenta este proyecto; específicamente con el siguiente software:

Sistema Operativo para Servidor:	Debian 8
Sistema Operativo para Móviles:	Android 5.0+
Gestor de Base de Datos:	MongoDB 3.6.x
Lenguaje de Programación:	JavaScript ES6
Servidor de aplicaciones:	Node.js + Express.js

Por lo mencionado anteriormente es que no se realizará una evaluación para la selección de ellos, en cambio, se analizarán las características de cada uno para asegurar que cumplen con los requerimientos mínimos para el funcionamiento del SICEVED.

3.1.2.4 Legales

El uso del software necesario para el desarrollo de SICEVED, está regido por la Ley de Derecho de Autor y Leyes y Tratados sobre la propiedad intelectual¹⁴. Por tal motivo SAF será utilizado bajo la licencia que requiera el propietario legal del mismo. Incluye las licencias de las herramientas que se utilizaran como: el Sistema Administrador de Base de Datos, el Lenguaje de Programación y el Sistema Operativo o plataforma sobre la cual se va a programar.

¹³ Ver Anexo F: Software Libre

¹⁴ Ver Anexo G: Ley de Fomento y Protección de la Propiedad Intelectual

De acuerdo al artículo 29 del Capítulo V del Reglamento General de Procesos de Trabajos de Graduación, los derechos de Autor sobre los trabajos de investigación elaborados en los procesos de graduación serán de propiedad exclusiva de la Universidad de El Salvador, la cual podrá disponer de los mismos de conformidad a su marco jurídico interno y legislación aplicable.

3.2 Requerimientos de Desarrollo

3.2.1 Tecnológicos

3.2.1.1 Equipo

Para el desarrollo del sistema se requiere que cumpla como mínimo las siguientes características:

Recurso requerido	Descripción	Requerido
Estaciones de trabajo	Procesador Pentium 4 o superior Memoria RAM 2GB Disco duro 20GB	3
Dispositivo móvil	Dispositivo móvil de gama media	1
Servidor de desarrollo	Procesador Xeon E7 o superior Memoria RAM 8GB Disco duro 300GB Tarjeta de red de 10/100 Mbps	1

Tabla 23: Requerimientos de hardware de desarrollo.

3.2.1.2 Lenguaje de programación

Características necesarias:

- Multiparadigma.
- Tipado dinámico.
- Basado en objetos utilizando prototipos.
- Funcionalidades asíncronas.
- Ejecución de lado del cliente y del servidor.

Lenguaje elegido: **JavaScript ECMAScript 6.**¹⁵

3.2.1.3 Gestor de Base de Datos

Para la construcción del sistema SICEVED es necesario que la base de datos a utilizar cumpla con los siguientes requerimientos:

¹⁵ Ver Anexo H: JavaScript

- Alto rendimiento en escritura/lectura de data.
- Modelos de datos flexibles.
- Almacenamiento de datos geoespaciales.
- Ejecución de queries geoespaciales.
- Escalabilidad horizontal y vertical.
- Gratis y open source.

Gestor de bases de datos elegido: **MongoDB**.¹⁶

3.2.1.4 Servidor Web

Características necesarias:

- Asíncrono y E/S sin bloqueo.
- Arquitectura basada en eventos.
- Servir contenido dinámico.
- Manejo de errores.
- Programación de tareas.

Servidor elegido: **Node.js + Express.js**.¹⁷

3.2.1.5 Sistema operativo

Un sistema operativo es el software principal o conjunto de programas de un sistema informático que gestiona los recursos de hardware y provee servicios a los programas de aplicación de software.

Características necesarias en el servidor:

- Soportar programas multiusuarios
- Soportar aplicaciones en redes
- Gestión de errores
- Multiprocesos
- Seguridad
- Estabilidad
- Alto rendimiento
- Facilidad de actualizaciones

Características necesarias en el cliente web:

- Multiprocesos
- Organizar datos para acceso rápido y seguro
- Manejo de comunicaciones en red
- Recuperación de errores
- Administrar el hardware

¹⁶ Ver Anexo I: MongoDB

¹⁷ Ver Anexo J: Node.js

- Facilitar las entradas y salidas
- Navegación Web

Características necesarias en el cliente móvil:

- Multitarea
- Multiproceso
- Sistema de Posicionamiento Global GPS
- Conectividad Inalámbrica
- Administración del Hardware
- Administración de Aplicaciones
- Navegación Web

Sistemas operativos elegidos:

Servidor: **Debian 8 o superior**¹⁸

Cliente: **Windows 7, Debian 8, Mac 10 o superiores**, debido a que el sistema SICEVED es Web es independiente del sistema operativo que posean las máquinas cliente.

Móvil: Android **5.0 o superior** ya que está sujeto al dispositivo Android que posea el evaluador.

3.2.1.6 Herramienta de control de versiones

Características necesarias:

- Flujo de trabajo centralizado.
- Control de versiones de repositorios locales distribuido.
- Creación de ramas con roles.
- Integridad criptográfica de la data.
- Procesamiento veloz de las operaciones.
- Áreas intermedias para revisión de commits.
- Gratis y open source.

Herramienta elegida: **Git**.

3.2.1.7 Herramientas utilitarias

Procesadores de Texto

El procesador de texto es un tipo de aplicación informática para la creación, edición, modificación y procesamiento de documentos de texto con formato (tal como el tipo y tamaño de la tipografía, adición de gráficos, etcétera), a diferencia de los editores de texto, que manejan solo texto simple.

Los procesadores de textos son una clase de software con múltiples funcionalidades para la redacción, con diferentes tipografías, tamaños de letras o caracteres, colores, tipos de párrafos, efectos artísticos y otras opciones.

¹⁸ Ver Anexo K: Ventajas de usar Debian

Características necesarias

- Insertar texto o imágenes nuevas en cualquier lugar del documento.
- Copiar o duplicar una sección indicada del documento.
- Borrar o eliminar caracteres, palabras, líneas, páginas o imágenes.
- Pegar o insertar material que fue removido o copiado de otras partes de un documento.
- Formato para el diseño del documento, especificando la página, el margen, el tamaño del margen y aplicando características de diseño específicas, como el tipo de fuente, el color, las negritas, itálicas, subrayado y lo que va remarcado.
- Buscar y restituir caracteres, palabras o frases específicas dentro del documento.
- Crear columnas y tablas, además manipular y dar formato a las columnas y tablas.
- Administrar archivos, almacenar, acceder, mover y eliminar los archivos de la computadora.
- Impresión, generar una copia en papel de un archivo almacenado electrónicamente en la computadora.

Herramienta elegida

- **Procesador de Texto**
Microsoft word: Word cuenta con amplias características y funciones como formatos, alineaciones, tablas, gráficos, colores de letras, estilos de letras, cortar y pegar texto, cambiar tamaño de las letras, imprimir, ingresar imágenes insertar vínculos o hipervínculos etc.

3.2.1.8 Diagramadores

Las herramientas de modelado se utilizan para el diseño de datos a manera de acercarnos a la comprensión y estudio de un modelo existente (en aquellas con características de ingeniería inversa). La capacidad de realizar un diseño de modelo visualizando de forma gráfica, a través de los Diagramas E-R, o de obtener un diagrama y documentación de una base de datos existente aumenta enormemente la eficiencia de nuestro trabajo reduciendo drásticamente el tiempo necesario con respecto a una realización manual de dichas tareas de diseño, análisis y estudio.

Características necesarias

- Debe estar disponible para Windows y Linux, como mínimo.
- Debe tener características de modelado de datos (no basta con una herramienta de diagramas o que sea sólo para UML).
- Debe tener características de ingeniería directa e inversa.
- Para el caso de bases de datos no relacionales como MongoDB permitir la representación gráfica de los documentos su lógica de estructura.
- Debe tener características de documentación: generar diagramas E-R, y documentación de la base de datos.
- Debe ser gratuita y, preferentemente, libre.

Herramientas elegidas

- **ERDPlus**

Es una herramienta de modelado de bases de datos basada en web que permite crear rápida y fácilmente los siguientes diagramas:

- Entity Relationship Diagrams (ERD)
- Esquemas Relacionales (Diagramas Relacionales)
- Esquemas de estrellas (modelos dimensionales)

- **Draw.io**

Como Draw.io está basado en el navegador, cualquier computadora con acceso a Internet puede usar la aplicación. Los dispositivos móviles pueden abrir el sitio web real, pero de otro modo no puede utilizar la aplicación de una manera práctica. Sin embargo, los archivos se pueden cargar desde Google Drive y se pueden ver al utilizar un dispositivo móvil dentro de Draw.io.

- **Astah Community**

Ofrece las siguientes funcionalidades:

- Soporte de UML 1.4 (parte de la expresión UML 2.0 en la versión comercial/ Professional)
- Diagrama de la clase (los diagramas del objeto, del paquete, del subsistema y de la robustez se incluyen)
- Use el diagrama del caso
- Diagrama de secuencia
- Diagrama de colaboración
- Diagrama de estado
- Diagrama de actividad
- Diagrama de implementación
- Diagrama de componentes
- Generar código fuente Java 1.4 del modelo
- Importar archivos de origen de Java 1.4 para crear un modelo

3.2.2 Legales

En el presente proyecto se respeta y se hace cumplir la ley de los derechos de autor cumpliendo con todas las prerrogativas que dicha ley establece, con la finalidad de evitar multas y demandas en el momento de implementar el sistema.

La Universidad de El Salvador es la propietaria del sistema a desarrollar. Una vez desarrollado el Sistema Informático la escuela de ingeniería civil, o cualquier otra institución con fines no lucrativos, tendrá derecho a solicitarlo mediante una carta dirigida a la Dirección de la Escuela de Ingeniería de Sistemas Informáticos.

3.2.3 Recurso humano

El recurso humano involucrado para el desarrollo del proyecto debe estar equilibrado en cuanto a la contribución del aporte a la construcción del proyecto y que este sea finalizado con éxito, por lo cual se establece una estructura de roles que tendrá cada integrante del grupo y a la vez esto contribuirá a que exista una mejor comunicación entre los miembros; considerando que el líder del proyecto será quien tomará iniciativas y podrá tomar decisiones finales cuando no exista un consenso general del grupo.

3.2.3.1 Roles y responsabilidades

A continuación, se presentan los roles y las responsabilidades del equipo de desarrollo.

Rol	Responsabilidad
Administrador de proyectos	Se encarga de dirigir y coordinar la ejecución del proyecto en conjunto con el equipo de trabajo, asimismo informa y da a conocer sobre el estado de cada fase del proyecto, también de llevar a cabo reuniones con representantes de la Institución.
Analista	Realizarán levantamiento de requerimientos que debe satisfacer a la solución a desarrollar.
Programador	Tendrán a su cargo la construcción de la solución y elaborarán la documentación respectiva.
Administrador de bases de datos	Será la persona que brindará apoyo a las tareas de diseño y creación de la base de datos, también tendrá a su cargo la administración del sistema manejador de las bases de datos que utilizará la solución.

Tabla 24: Descripción de roles y responsabilidades.

3.2.3.2 Perfiles

A continuación, se presentan las capacidades y el grado académico requerido para cada rol.

Rol	Responsabilidad	Capacidades y Competencias deseadas
Administrador de proyectos	Egresado de la carrera de Ingeniería de Sistemas Informáticos.	Facilidad para comunicar y expresar ideas. Flexibilidad mental de criterios. Habilidades para la obtención y análisis de información. Orientación al cliente. Interés por la innovación. Capacidad de síntesis. Visión estratégica
Analista	Egresado de la carrera de Ingeniería de Sistemas Informáticos.	Habilidades para la obtención y análisis de información. Capacidad de análisis. Interés por la innovación.

		Habilidades para mantener la atención.
Programador	Egresado de la carrera de Ingeniería de Sistemas Informáticos.	Habilidades para el análisis de información y con experiencia en programación Web. Manejo de diferentes lenguajes de programación Web. Interés por la innovación. Habilidades de atención.
Administrador de base de datos	Egresado de la carrera de Ingeniería de Sistemas Informáticos.	Flexibilidad mental de criterios. Habilidades para la recolección, obtención y análisis de información. Capacidad de diseñar bases de datos relacionales. Conocimiento y experiencias con Gestores de Bases de Datos.

Tabla 25: Descripción de los perfiles.

3.3 Requerimientos operativos

3.3.1 Tecnológicos

3.3.1.1 Equipo

El sistema está basado en una arquitectura cliente servidor ya que permite la centralización de la gestión de la información y la separación de responsabilidades, lo que facilita y clarifica el diseño del sistema, tomando como base lo anterior se definen los siguientes requerimientos mínimos para cada área.

Software

- **Software máquina cliente**

El software mínimo necesario para el funcionamiento del sistema o la visualización de reportes en las máquinas cliente se detalla en la tabla.

Clasificación	Software
Sistema Operativo	Cualquiera de los siguientes: <ul style="list-style-type: none"> • Windows 7 • Debian 8 • Mac OS X
Navegador	Cualquiera de los siguientes: <ul style="list-style-type: none"> • Firefox 57 • Chrome 61
Visualizador PDF	<ul style="list-style-type: none"> • Foxit Reader

Tabla 26: Descripción de los requerimientos operativos del sistema.

- **Software dispositivo móvil**

Para el funcionamiento de la app en los dispositivos móviles se detalla el software necesario.

Clasificación	Software
Sistema Operativo Móvil	Android 5.0 o superior

Tabla 27: Requerimientos operativos de software para dispositivos móviles.

- **Software Servidor**

A continuación, se detalla el software mínimo requerido para el funcionamiento del sistema en el servidor.

Clasificación	Software
Sistema Operativo	Debian 8
Servidor web	NodeJS 8.11.2 LTS / Express

Tabla 28: Descripción de requerimientos operativos de software en el servidor.

Hardware

- **Hardware maquina cliente**

En la siguiente tabla se detallan los requerimientos mínimos que debe cumplir el equipo del cliente para el sistema.

Características	Requerimientos mínimos
CPU	Pentium 4
Memoria RAM	2GB
Disco Duro (Espacio Disponible)	100 GB
Interfaz de Red	Tarjeta de red de 10/100 Mbps
Monitor	<ul style="list-style-type: none"> • VGA o HDMI • 17"
Periféricos	<ul style="list-style-type: none"> • Teclado • Mouse

Tabla 29: Requerimientos operativos de hardware del cliente web.

- **Hardware dispositivo móvil**

En la siguiente tabla se detallan los requerimientos mínimos que debe cumplir el equipo del cliente para que el sistema.

Características	Requerimientos mínimos
CPU	Procesador Snapdragon 801 2.5GHz o superior
Memoria RAM	2GB

Almacenamiento interno	8GB
Pantalla	4.7", 1080 x 1920 pixels
Cámara	16 MP
Extra	GPS

Tabla 30: Requerimientos operativos de hardware para dispositivos móviles.

- **Hardware Servidor**

A continuación, se detallan los requerimientos mínimos de Hardware con los que debe cumplir el servidor para el funcionamiento del sistema

Características	Requerimientos mínimos
CPU	Xeon E7 o superior
Memoria RAM	8GB
Disco Duro (Espacio Disponible)	500GB hábiles. Con redundancia distribuida con paridad y discos de reserva (RAID 5E).
Fuente de energía	2 Fuentes de alimentación eléctrica redundantes.
Interfaz de Red	2 Interfaces de red de 10/100 Mbps.

Tabla 31: Requerimientos operativos de hardware del servidor.

3.3.1.2 Gestor de Bases de Datos

Tomando como base la investigación realizada por el equipo¹⁹ la base de datos que mejor se acopla a las necesidades del sistema se muestra en la tabla.

Clasificación	Software
Gestor de Base de Datos	MongoDB 3.6.x

Tabla 32: Requerimiento operativo de gestor de base de datos.

3.3.1.3 Servidor Web

El servidor Web a utilizar se detalla en la siguiente tabla

Clasificación	Software
Servidor Web	Node.js + Express.js

Tabla 33: Requerimientos operativos de servidor web.

3.3.2 Recurso humano

El recurso humano necesario para el correcto funcionamiento del SICEVED se describe en la siguiente tabla:

Perfil del cargo: Soporte Técnico	
Nivel educativo:	<ul style="list-style-type: none"> • Ingeniero o estudiante de 5to año de Ciencias de la Computación o Sistemas.

¹⁹ Ver Anexo I: MongoDB

	<ul style="list-style-type: none"> • Técnico graduado en Ciencias de la Computación o Redes
Experiencia laboral	<ul style="list-style-type: none"> • Experiencia en el manejo y operación de equipo computacional y redes LAN. • Experiencia en el desarrollo de sistemas • Experiencia en administración de bases de datos • Experiencia en administración de servidores linux
Conocimientos necesarios	<ul style="list-style-type: none"> • Conocimientos sólidos de redes • Conocimiento de configuración de servidores • Conocimiento de bases de datos NoSQL • Conocimientos de servidor Node.js
Funciones	<ul style="list-style-type: none"> • Administrar la base de datos • Administrar el servidor web • Administrar la plataforma del servidor

Tabla 34: Requerimientos operativos de recurso humano.

3.4 Solución propuesta

3.4.1 Enfoque de Sistemas de Sistema Propuesto

En esta sección se presenta de forma integral el sistema propuesto, a través de una visión global que permite identificar las interrelaciones de todos los elementos que conforman el sistema.

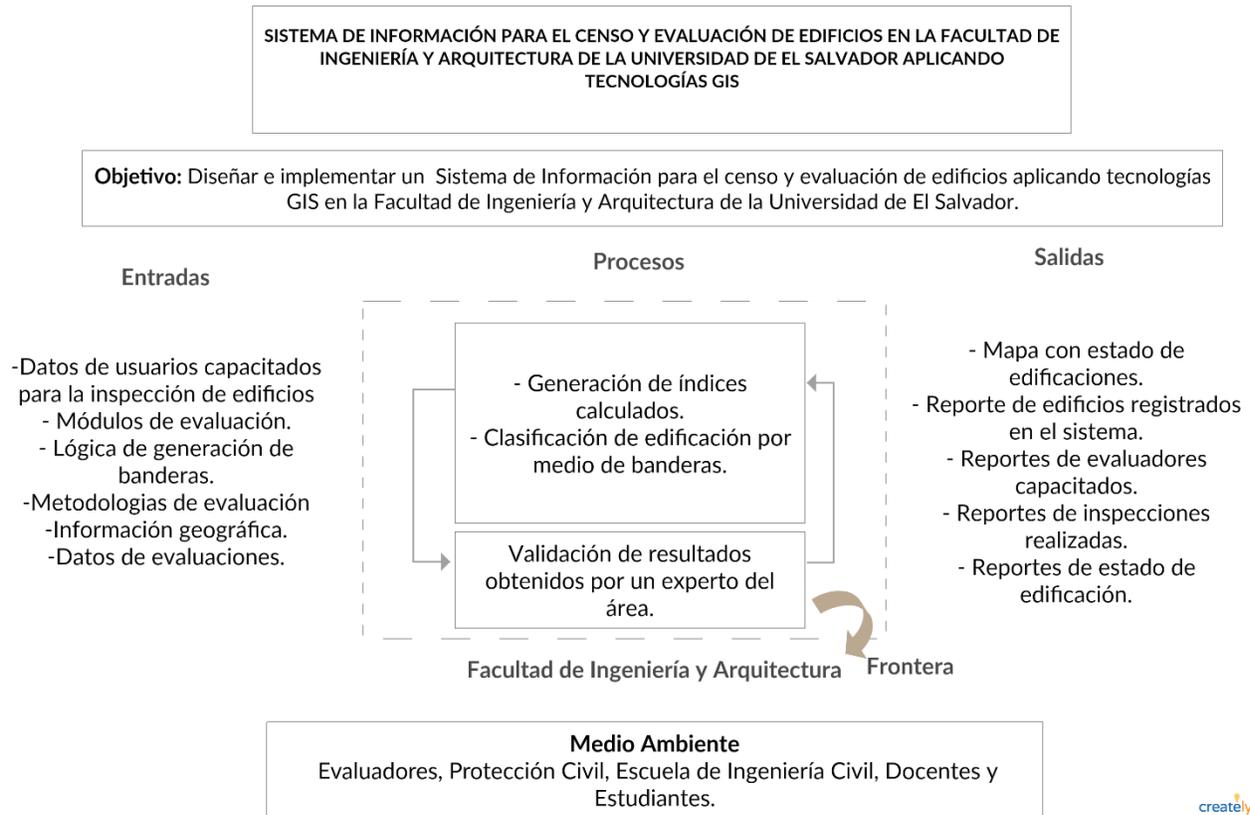


Ilustración 10. Diagrama de sistema, propuesta.

Objetivo: Diseñar e implementar un Sistema de Información para el censo y la evaluación de edificios aplicando tecnologías GIS en la Facultad de Ingeniería y Arquitectura de la Universidad de El Salvador.

Salidas:

- Mapa con estado de edificaciones: Se mostrará un mapa de las edificaciones con filtros para módulos aplicados, el cual tendrá indicadores gráficos con el resultado de las inspecciones.
- Reporte de edificios registrados en el sistema: El reporte mostrará los edificios ingresados en el sistema.
- Reportes de evaluadores capacitados: Los reportes mostrarán la información de los evaluadores capacitados para realizar evaluaciones.

- Reportes de inspecciones realizadas: El sistema podrá generar reportes de las evaluaciones realizadas, con la aplicación de módulos específicos, también podrá generar reportes de la información capturada en una inspección particular.
- Reportes de estado de edificaciones: Reporte del histórico de un edificio, mostrando el resultado de las inspecciones a lo largo del tiempo.

Entradas:

- Datos de usuarios capacitados para la inspección de edificios: Estos datos serán recolectados una vez un evaluador haya recibido capacitación en la metodología de evaluación de edificios.
- Módulos de evaluación: Los módulos de evaluación representarán los instrumentos utilizados para recolectar los datos, inicialmente se trabajará con un instrumento base que contiene ítems evaluativos que fueron consensuados por diferentes instituciones del país, sin embargo, el sistema permitirá ingresar múltiples instrumentos para poder realizar levantamientos de datos según necesidades dinámicas.
- Lógica de generación de banderas: A cada módulo de evaluación se le podrá configurar una lógica de generación de banderas en base a índices cuantitativos o cualitativos, los cuales se utilizarán como valores axiales en un directorio de resultados en formato matricial.
- Metodologías de evaluación: Ciertos elementos de la metodología de evaluación se incluirán de manera implícita en los cálculos de índices de cada módulo de evaluación.
- Información geográfica: Necesaria para dividir en elementos atómicos las edificaciones que serán inspeccionadas. Esta información podrá ser generada antes de realizar las inspecciones, contemplando de esta manera una etapa de censo de edificaciones, o en el momento de la inspección.
- Datos de evaluaciones: Esta información será recolectada por medio de los instrumentos que componen los diferentes módulos de inspección, previamente ingresados al sistema, el llenado de dichos instrumentos se llevará a cabo por medio de dispositivos móviles, para facilitar la recolección de datos en sitio.

Procesos:

- Generación de índices calculados: Previo a la clasificación de una edificación el sistema realizará los cálculos necesarios para generar índices con semántica atómica (como por ejemplo índice de seguridad estructural en paredes, índice de seguridad no estructural, etc), los cuales pueden ayudar a establecer la clasificación por medio de banderas o usarse para análisis posteriores de los datos.

- Clasificación de edificación por medio de banderas: El sistema otorgará una bandera a cada edificación de acuerdo datos recolectados con los instrumentos que conforman los módulos de evaluación.

Medio Ambiente:

- Evaluadores: personas capacitadas para poder realizar las visitas a las edificaciones.
- Protección civil: Institución interesada en el estado de los edificios, las personas que la integran aprobarán la consistencia y utilidad de los datos de entrada para su posterior procesamiento.
- Escuela de Ingeniería Civil: Entidad que se encargará de la administración del sistema y la información recolectada.
- Docentes y Estudiantes: Usuarios de las edificaciones de la Facultad de Ingeniería y Arquitectura de la Universidad de El Salvador y principales beneficiarios del control de las condiciones de riesgo en dichas instalaciones.

Control:

- Validación de resultados obtenidos por un experto del área:
 - Inspectores utilizan con facilidad el cliente móvil para el llenado de los instrumentos, permitiendo recolectar la información necesaria.
 - Los expertos del área ven la información recolectada durante las inspecciones como datos válidos.
 - Las banderas obtenidas se apegan a la realidad desde una visión objetiva.

Frontera:

- Facultad de Ingeniería y Arquitectura de la Universidad de El Salvador.

3.4.2 Modelos de Casos de Uso Propuesto

El modelo de casos de uso describe la funcionalidad propuesta del sistema. Un caso de uso es una unidad de trabajo significativo; por ejemplo, crear una solicitud y modificar una solicitud son todos casos de uso. Cada caso de uso tiene una descripción que especifica la funcionalidad que se incorporará al sistema propuesto. Los diagramas de casos de uso son, un diagrama que muestra la relación entre los actores y los casos de uso en un sistema. Una relación es una conexión entre los elementos del modelo. Los diagramas de casos de uso se utilizan para ilustrar los requerimientos del sistema al mostrar cómo reacciona una respuesta a eventos que se producen en el mismo.

Los elementos básicos para el diagrama de casos de uso son:

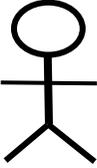
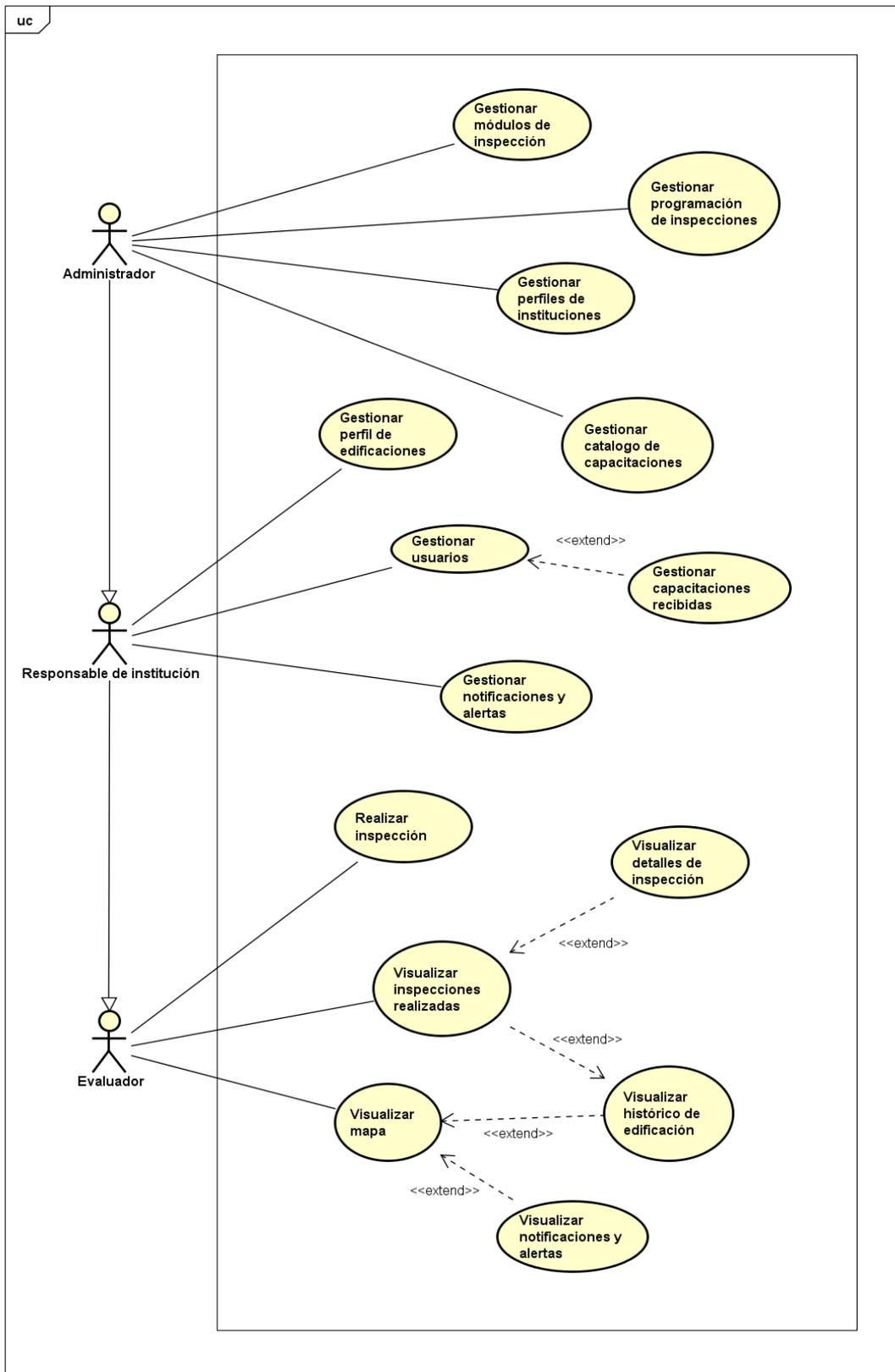
Elemento	Descripción	Símbolo
Casos de Uso	Secuencia de interacciones entre un sistema y alguien o algo que usa alguno de sus servicios.	
Actores	Rol o función que asume una persona, sistema o entidad que interactúa con el sistema que estamos construyendo	
Líneas de comunicación entre actores y casos de uso	Muestran la relación y comunicación entre los actores y los casos de uso.	

Tabla 35: Descripción de nomenclatura de casos de uso.

3.4.2.1 Diagrama de Casos de Uso Propuesto



powered by Astah

Ilustración 11. Diagrama de casos de uso.

3.4.2.2 Lista actor-objetivo

Actor	Objetivo
Administrador	Gestionar módulos de inspección. Gestionar programación de inspecciones. Gestionar perfiles de instituciones. Gestionar catálogo de capacitaciones.
Responsable de institución	Gestionar perfil de edificaciones. Gestionar usuarios. Gestionar capacitaciones recibidas. Gestionar notificaciones y alertas.
Evaluador	Realizar inspección. Visualizar inspecciones realizadas. Visualizar detalles de inspección. Visualizar mapa Visualizar histórico de edificación. Visualizar notificaciones y alertas.

Tabla 36: Descripción de los actores del sistema.

3.4.2.3 Descripción de Casos de Uso

Caso de uso	Gestionar módulos de inspección	
Actores	Administrador	
Descripción	Permite ingresar, actualizar o eliminar módulos de inspección.	
Referencias	REQ-FF01, REQ-FF03, REQ-FF04, REQ-FF05, REQ-FF06, REQ-FF07, REQ-FF08, REQ-FF12	
Precondición	El usuario debe haber iniciado sesión.	
Postcondición	Se habrá agregado, actualizado o eliminado un módulo de inspección.	
Curso normal de los eventos		
Acción de los actores	Respuesta del sistema	
1. El usuario entra a la sección de mapa.	2. El sistema muestra el mapa.	
3. El usuario entra a la sección de gestión de módulos de inspección.	4. El sistema muestra la lista de módulos existentes junto a las opciones disponibles (agregar, actualizar o eliminar módulo).	
5. El usuario elige una opción de gestión a) Agregar, ver sección agregar. b) actualización, ver sección actualización. c) eliminación, ver sección eliminación.	6. El sistema muestra la lista de módulos actualizados.	
Sección: Agregar		
Curso normal de los eventos		
Acción de los actores	Respuesta del sistema	
	1. El sistema despliega una ventana para agregar un nuevo módulo de inspección.	

2. El usuario ingresa los datos generales del nuevo módulo de inspección.	3. El sistema muestra una pantalla dividida en dos secciones: en la primera se mostrará los parámetros de configuración de entradas y la segunda mostrará una vista previa del instrumento de evaluación.
4. El usuario selecciona el componente a añadir en el instrumento.	5. El sistema muestra los parámetros de configuración de acuerdo al componente seleccionado por el usuario.
6. El usuario configura el componente.	7. El sistema actualiza la vista previa del instrumento de evaluación.
	8. El sistema muestra la sección de configuración para la lógica de banderas.
9. El usuario ingresa la cantidad de celdas para crear la matriz de referencia, para la generación de banderas.	10. El sistema muestra los campos de la matriz de referencia. Con los selectores para el color de bandera según el caso de cruce dado por los índices.
11. El usuario ingresa los índices que utilizará la matriz de referencia, así como los colores de bandera para cada caso de cruce.	
12. El usuario selecciona finalizar módulo.	13. El sistema ingresa los datos a la base de datos y muestra un mensaje de éxito.
Sección: Actualización	
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El usuario selecciona el módulo a editar.	2. El sistema despliega una ventana con campos editables incluyendo los datos actuales del módulo de inspección.
3. El usuario modifica los datos.	4. El sistema ingresa los datos a la base de datos como una versión completa y muestra un mensaje de éxito.
Sección: Eliminación	
Curso normal de los eventos	
1. El usuario selecciona el módulo a eliminar.	2. El sistema despliega mensaje de alerta ante la eliminación de datos.
3. El usuario verifica la eliminación del módulo.	4. El sistema procesa eliminación (En caso que existan inspecciones aplicadas con dicho módulo la eliminación es una inhabilitación del módulo sino se procede a una eliminación de persistencia) y muestra un mensaje de éxito.
Cursos alternativos – Sección Agregar	
<ul style="list-style-type: none"> • Línea 2. El usuario cancela la acción: El sistema regresa a las opciones de gestión. 	
Cursos alternativos – Sección Actualización	
<ul style="list-style-type: none"> • Línea 3. El usuario cancela la acción: El sistema regresa a las opciones de gestión. 	
Cursos alternativos – Sección Eliminación	

- Línea 4. El usuario cancela la eliminación del módulo: El sistema cancela la acción y regresa a la sección de gestión de módulos de inspección.

Tabla 37: Descripción de caso de uso para la gestión de módulos de inspección.

Caso de uso		Gestionar programación de inspecciones	
Actores	Administrador		
Descripción	Permite ingresar, actualizar o eliminar programaciones de inspección.		
Referencias	REQ-FI01, REQ-FI02		
Precondición	El usuario debe haber iniciado sesión.		
Postcondición	Se habrá agregado, actualizado o eliminado una programación de inspección.		
Curso normal de los eventos			
Acción de los actores		Respuesta del sistema	
1. El usuario entra a la sección de mapa.		2. El sistema muestra el mapa.	
3. El usuario entra a la sección de programación de inspecciones.		4. El sistema muestra la lista de inspecciones programadas y las opciones de gestión (agregar, actualizar o eliminar).	
5. El usuario elige una opción de gestión a) Ingreso, ver sección ingreso. b) actualización, ver sección actualización. c) eliminación, ver sección eliminación.		6. El sistema muestra la lista de inspecciones programadas actualizadas.	
Sección: Ingreso			
Curso normal de los eventos			
Acción de los actores		Respuesta del sistema	
		1. El sistema despliega una ventana para programar una nueva inspección.	
2. El usuario ingresa los detalles de la programación de inspección.		3. El sistema muestra los módulos que se le pueden aplicar a la edificación a inspeccionar.	
4. El usuario selecciona el módulo que se usará en la inspección		5. El sistema ingresa los datos a la base de datos, genera notificaciones automáticas para los evaluadores y muestra un mensaje de éxito.	
Sección: Actualización			
Curso normal de los eventos			
Acción de los actores		Respuesta del sistema	
1. El usuario selecciona la programación de inspección a editar.		2. El sistema despliega una ventana con campos editables incluyendo los datos actuales de la inspección programada.	
3. El usuario modifica los datos.		4. El sistema ingresa los datos a la base de datos, genera notificaciones automáticas para los evaluadores y muestra un mensaje de éxito.	
Sección: Eliminación			

Curso normal de los eventos	
1. El usuario selecciona la programación de inspección a eliminar.	2. El sistema despliega mensaje de alerta ante la eliminación de datos.
3. El usuario verifica la eliminación de inspección programada.	4. El sistema procesa eliminación, genera notificaciones automáticas para los evaluadores y muestra un mensaje de éxito.
Cursos alternativos – Sección Ingreso	
<ul style="list-style-type: none"> • Línea 2. El usuario cancela la acción: El sistema regresa a las opciones de gestión. 	
Cursos alternativos – Sección Actualización	
<ul style="list-style-type: none"> • Línea 3. El usuario cancela la acción: El sistema regresa a las opciones de gestión. 	
Cursos alternativos – Sección Eliminación	
<ul style="list-style-type: none"> • Línea 3. El usuario cancela la eliminación: El sistema cancela la acción y regresa a la sección de gestión. 	

Tabla 38: Descripción de caso de uso para la gestionar la programación de inspecciones.

Caso de uso	Gestionar perfiles de instituciones	
Actores	Administrador	
Descripción	Permite ingresar, actualizar o eliminar información de instituciones.	
Referencias	REQ-FG02	
Precondición	El administrador del sistema debe haber iniciado sesión.	
Postcondición	Se habrá agregado, actualizado o eliminado perfiles de instituciones.	
Curso normal de los eventos		
Acción de los actores	Respuesta del sistema	
1. El usuario entra a la sección de mapa.	2. El sistema muestra el mapa.	
3. El usuario entra a la sección de gestión de instituciones.	4. El sistema muestra las opciones disponibles (agregar, actualizar o eliminar institución).	
5. El usuario elige una opción de gestión <ul style="list-style-type: none"> a) Ingreso, ver sección ingreso. b) actualización, ver sección actualización. c) eliminación, ver sección eliminación. 	6. El sistema muestra las instituciones actualizadas.	
Sección: Ingreso		
Curso normal de los eventos		
Acción de los actores	Respuesta del sistema	
	1. El sistema despliega una ventana para agregar una nueva institución y relacionar uno o más responsables de institución.	
2. El usuario ingresa los datos de la nueva institución.	3. El sistema ingresa los datos a la base de datos y muestra un mensaje de éxito.	
Sección: Actualización		
Curso normal de los eventos		
Acción de los actores	Respuesta del sistema	
1. El usuario selecciona la institución a editar.	2. El sistema despliega una ventana con campos editables incluyendo los datos actuales de la institución.	

3. El usuario modifica los datos.	4. El sistema ingresa los datos a la base de datos y muestra un mensaje de éxito.
Sección: Eliminación	
Curso normal de los eventos	
1. El usuario selecciona la institución a eliminar.	2. El sistema verifica integridad referencial.
	3. El sistema despliega mensaje de alerta ante la eliminación de datos.
4. El usuario verifica la eliminación de institución.	5. El sistema procesa eliminación y muestra un mensaje de éxito.
Cursos alternativos – Sección Ingreso	
<ul style="list-style-type: none"> • Línea 2. El usuario cancela la acción: El sistema regresa a las opciones de gestión. 	
Cursos alternativos – Sección Actualización	
<ul style="list-style-type: none"> • Línea 3. El usuario cancela la acción: El sistema regresa a las opciones de gestión. 	
Cursos alternativos – Sección Eliminación	
<ul style="list-style-type: none"> • Línea 2. La integridad referencial es violada: El sistema despliega mensaje de error describiendo los detalles del mismo y regresa a la sección de gestión de instituciones. • Línea 4. El usuario cancela la eliminación de institución: El sistema cancela la acción y regresa a la sección de gestión de instituciones. 	

Tabla 39: Descripción de caso de uso para la gestión de instituciones.

Caso de uso	Gestionar catálogo de capacitaciones	
Actores	Administrador	
Descripción	Permite ingresar, actualizar o eliminar capacitaciones.	
Referencias	REQ-FC01, REQ-FC02	
Precondición	El usuario debe haber iniciado sesión.	
Postcondición	Se habrán agregado, actualizado o eliminado capacitaciones.	
Curso normal de los eventos		
Acción de los actores	Respuesta del sistema	
1. El usuario entra a la sección de mapa.	2. El sistema muestra el mapa.	
3. El usuario entra a la sección de catálogo de capacitaciones.	4. El sistema muestra las opciones disponibles (agregar, actualizar o eliminar capacitación).	
5. El usuario elige una opción de gestión <ul style="list-style-type: none"> a) Ingreso, ver sección ingreso. b) actualización, ver sección actualización. c) eliminación, ver sección eliminación. 	6. El sistema muestra el catálogo de capacitaciones actualizado.	
Sección: Ingreso		
Curso normal de los eventos		
Acción de los actores	Respuesta del sistema	
	1. El sistema despliega una ventana para agregar una nueva capacitación.	
2. El usuario ingresa los datos de la nueva capacitación.	3. El sistema ingresa los datos a la base de datos y muestra un mensaje de éxito.	
Sección: Actualización		

Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El usuario selecciona la capacitación a editar.	2. El sistema despliega una ventana con campos editables incluyendo los datos actuales de la capacitación.
3. El usuario modifica los datos.	4. El sistema ingresa los datos a la base de datos y muestra un mensaje de éxito.
Sección: Eliminación	
Curso normal de los eventos	
1. El usuario selecciona la capacitación a eliminar.	2. El sistema verifica integridad referencial.
	3. El sistema despliega mensaje de alerta ante la eliminación de datos.
4. El usuario verifica la eliminación de capacitación.	5. El sistema procesa eliminación y muestra un mensaje de éxito.
Cursos alternativos – Sección Ingreso	
<ul style="list-style-type: none"> • Línea 2. El usuario cancela la acción: El sistema regresa a las opciones de gestión. 	
Cursos alternativos – Sección Actualización	
<ul style="list-style-type: none"> • Línea 3. El usuario cancela la acción: El sistema regresa a las opciones de gestión. 	
Cursos alternativos – Sección Eliminación	
<ul style="list-style-type: none"> • Línea 2. La integridad referencial es violada: El sistema despliega mensaje de error describiendo los detalles del mismo y regresa a la sección de catálogo de capacitaciones. • Línea 4. El usuario cancela la eliminación de capacitación: El sistema cancela la acción y regresa a la sección de catálogo de capacitaciones. 	

Tabla 40: Descripción de caso de uso para la gestión de catálogos de capacitaciones.

Caso de uso	Gestionar perfil de edificaciones
Actores	Administrador, Responsable de institución
Descripción	Permite ingresar o actualizar perfiles de edificaciones.
Referencias	REQ-FF13, REQ-FM01
Precondición	El administrador del sistema debe haber iniciado sesión e ingresado información geográfica de zonas.
Postcondición	Se habrá agregado o actualizado información de perfiles de edificación.
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El usuario entra a la sección de mapa.	2. El sistema muestra el mapa.
3. El usuario entra a la sección de edificaciones.	4. El sistema muestra las opciones disponibles (agregar, actualizar o eliminar edificación).
5. El usuario elige una opción de gestión <ul style="list-style-type: none"> a) Ingreso, ver sección ingreso. b) actualización, ver sección actualización. c) eliminación, ver sección eliminación. 	6. El sistema muestra el catálogo de edificaciones actualizado.
Sección: Ingreso	

Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
	1. El sistema despliega una ventana para agregar una nueva edificación, incluyendo un campo para configurar su posición geográfica.
2. El usuario ingresa los datos de la nueva edificación.	3. El sistema ingresa los datos a la base de datos y muestra un mensaje de éxito.
Sección: Actualización	
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El usuario selecciona la edificación a editar.	2. El sistema despliega una ventana con campos editables incluyendo los datos actuales de la edificación.
3. El usuario modifica los datos.	4. El sistema ingresa los datos a la base de datos y muestra un mensaje de éxito.
Sección: Eliminación	
Curso normal de los eventos	
1. El usuario selecciona la edificación a eliminar.	2. El sistema despliega mensaje de alerta ante la eliminación.
3. El usuario verifica la eliminación de edificación.	4. El sistema realiza una eliminación lógica y muestra un mensaje de éxito.
Cursos alternativos – Sección Ingreso	
<ul style="list-style-type: none"> • Línea 2. El usuario cancela la acción: El sistema regresa a las opciones de gestión. 	
Cursos alternativos – Sección Actualización	
<ul style="list-style-type: none"> • Línea 3. El usuario cancela la acción: El sistema regresa a las opciones de gestión. 	
Cursos alternativos – Sección Eliminación	
<ul style="list-style-type: none"> • Línea 3. El usuario cancela la eliminación de edificación: El sistema cancela la acción y regresa a la sección de catálogo de edificaciones. 	

Tabla 41: Descripción de caso de uso para gestionar el perfil de las edificaciones.

Caso de uso	Gestionar usuarios
Actores	Administrador, Responsable de institución
Descripción	Permite ingresar, actualizar o eliminar usuarios. Los usuarios podrán tener diferentes niveles de acceso dentro del sistema, y reflejan las funciones de los diferentes roles.
Referencias	REQ-FG01
Precondición	El usuario debe haber iniciado sesión.
Postcondición	Se habrán agregado, actualizado o eliminado usuarios.
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El usuario entra a la sección de mapa.	2. El sistema muestra el mapa.
3. El usuario entra a la sección de gestión de usuario.	4. El sistema muestra las opciones disponibles (agregar, actualizar o eliminar usuario).
5. El usuario elige una opción de gestión a) Ingreso, ver sección ingreso.	6. El sistema muestra los usuarios actualizados.

b) actualización, ver sección actualización.	
c) eliminación, ver sección eliminación.	
Sección: Ingreso	
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
	1. El sistema despliega una ventana para agregar un nuevo usuario.
2. El usuario ingresa los datos del usuario, estos pueden ser cargados por medio de un csv.	3. El sistema ingresa los datos a la base de datos y muestra un mensaje de éxito, con un resumen de los usuarios procesados.
Sección: Actualización	
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El usuario selecciona el usuario a editar.	2. El sistema despliega una ventana con campos editables incluyendo los datos actuales del usuario.
3. El usuario modifica los datos.	4. El sistema ingresa los datos a la base de datos y muestra un mensaje de éxito con detalle de cambios.
Sección: Eliminación	
Curso normal de los eventos	
1. El usuario selecciona el usuario a eliminar.	2. El sistema verifica integridad referencial.
	3. El sistema despliega mensaje de alerta ante la eliminación de datos.
4. El usuario verifica la eliminación del usuario.	5. El sistema procesa eliminación y muestra un mensaje de éxito.
Cursos alternativos – Sección Ingreso	
<ul style="list-style-type: none"> • Línea 2. El usuario cancela la acción: El sistema regresa a las opciones de gestión. 	
Cursos alternativos – Sección Actualización	
<ul style="list-style-type: none"> • Línea 3. El usuario cancela la acción: El sistema regresa a las opciones de gestión. • Línea 3. El archivo csv no es tiene la estructura o los datos necesarios: El sistema solicita un nuevo archivo. 	
Cursos alternativos – Sección Eliminación	
<ul style="list-style-type: none"> • Línea 2. La integridad referencial es violada: El sistema despliega mensaje de error describiendo los detalles del mismo y regresa a la sección de gestión de usuarios. • Línea 4. El usuario cancela la eliminación de usuario: El sistema cancela la acción y regresa a la sección de gestión de usuarios. 	

Tabla 42: Descripción de caso de uso para la gestión de usuarios.

Caso de uso	Gestionar capacitaciones recibidas
Actores	Administrador, Responsable de institución
Descripción	Permite agregar o eliminar capacitaciones recibidas a un usuario específico.
Referencias	REQ-FC03

Precondición	El usuario debe haber iniciado sesión.
Postcondición	Se habrán agregado eliminado una capacitación recibida por un usuario.
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El usuario entra a la sección de mapa.	2. El sistema muestra el mapa de zonas.
3. El usuario entra a la sección de gestión de personal capacitado.	4. El sistema muestra el listado de usuarios.
5. El usuario selecciona un usuario de la lista	6. El sistema muestra la lista de capacitaciones que el usuario tiene asociados.
5. El usuario elige una opción de gestión a) Agregar, ver sección inferior. b) Eliminar, ver sección inferior.	6. El sistema muestra la lista de capacitaciones recibidas actualizada.
Sección: Agregar	
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
	1. El sistema despliega una ventana para seleccionar la capacitación recibida.
2. El usuario selecciona la capacitación recibida y llena la información complementaria.	3. El sistema ingresa los datos a la base de datos y muestra un mensaje de éxito.
Sección: Eliminar	
Curso normal de los eventos	
1. El usuario selecciona la capacitación recibida a eliminar.	2. El sistema despliega mensaje de alerta ante la eliminación de datos.
3. El usuario verifica la eliminación de la capacitación recibida.	5. El sistema procesa eliminación y muestra un mensaje de éxito.
Cursos alternativos – Sección Agregar	
<ul style="list-style-type: none"> • Línea 2. El usuario cancela la acción: El sistema regresa a las opciones de gestión. 	

Tabla 43: Descripción de caso de uso para la gestión de capacitaciones recibidas.

Caso de uso	Gestionar notificaciones y alertas
Actores	Administrador, Responsable de institución
Descripción	Permite ingresar, actualizar o eliminar entradas del panel de notificaciones y alertas.
Referencias	REQ-FG03
Precondición	El usuario debe haber iniciado sesión.
Postcondición	Se habrán agregado, actualizado o eliminado entradas del panel de notificaciones y alertas.
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El usuario entra a la sección de mapa.	2. El sistema muestra el mapa.
3. El usuario entra a la sección de gestión de notificaciones y alertas.	4. El sistema muestra la lista de entradas actuales y las opciones de gestión (agregar, actualizar o eliminar entrada).
5. El usuario elige una opción de gestión a) Ingreso, ver sección ingreso.	6. El sistema muestra las entradas actualizadas.

b) actualización, ver sección actualización. c) eliminación, ver sección eliminación.	
Sección: Ingreso	
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
	1. El sistema despliega una ventana para agregar una nueva entrada.
2. El usuario ingresa el mensaje para la nueva entrada, el tipo de entrada (aviso, emergencia o notificación), el alcance de la entrada (general o particular).	3. El sistema ingresa los datos a la base de datos y muestra un mensaje de éxito.
Sección: Actualización	
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El usuario selecciona la entrada a editar.	2. El sistema despliega una ventana con campos editables incluyendo los datos actuales de la entrada.
3. El usuario modifica los datos.	4. El sistema ingresa los datos a la base de datos y muestra un mensaje de éxito.
Sección: Eliminación	
Curso normal de los eventos	
1. El usuario selecciona la entrada a eliminar.	2. El sistema despliega mensaje de alerta ante la eliminación de datos.
3. El usuario verifica la eliminación de la entrada.	4. El sistema procesa eliminación y muestra un mensaje de éxito.
Cursos alternativos – Sección Ingreso	
<ul style="list-style-type: none"> • Línea 2. El usuario cancela la acción: El sistema regresa a las opciones de gestión. 	
Cursos alternativos – Sección Actualización	
<ul style="list-style-type: none"> • Línea 3. El usuario cancela la acción: El sistema regresa a las opciones de gestión. 	
Cursos alternativos – Sección Eliminación	
<ul style="list-style-type: none"> • Línea 3. El usuario cancela la eliminación de la entrada: El sistema cancela la acción y regresa a la sección de gestión. 	

Tabla 44: Descripción de caso de uso para la gestión de alertas y notificaciones.

Caso de uso	Realizar inspección
Actores	Administrador, Responsable de institución, Evaluador
Descripción	Permite a los evaluadores realizar la recolección de datos.
Referencias	REQ-FF02, REQ-FF10, REQ-FF11, REQ-FI01, REQ-FI02, REQ-FI03, REQ-FI04, REQ-FI06
Precondición	El usuario debe haber iniciado sesión, el perfil de la edificación a evaluar tienen que haber sido ingresados.
Postcondición	Se le asocia a la edificación una inspección realizada.
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema

1. El usuario entra a la sección de mapa.	2. El sistema muestra el mapa.
3. El usuario entra a la sección de realizar inspección. O selecciona directamente la edificación a inspeccionar, de ser así se omiten los pasos 4 y 5.	4. El sistema muestra la lista de edificaciones que puede inspeccionar el evaluador, con diferentes tipos de filtros.
5. El usuario busca y selecciona la edificación a evaluar.	6. El sistema muestra los diferentes instrumentos de los módulos que pueden ser aplicados, así como los campos para describir el motivo de la inspección y agregar los asistentes de la inspección.
7. El usuario ingresa la información solicitada e inicia la inspección.	8. El sistema muestra el instrumento de inspección.
9. El usuario llena los elementos del instrumento.	10. El sistema guarda los datos de la inspección y muestra la bandera obtenida.
Cursos alternativos	
<ul style="list-style-type: none"> • Línea 7. Si la inspección ya existía con el estado de incompleta el usuario puede continuar con el paso 9. • Línea 9. El usuario cancela la realización de inspección: El sistema guarda la inspección con el estado de incompleta y regresa a la sección de realizar inspección. 	

Tabla 45: Descripción de caso de uso para realizar una inspección.

Caso de uso	Visualizar inspecciones realizadas	
Actores	Administrador, Responsable de institución, Evaluador	
Descripción	Permite visualizar las inspecciones que hayan realizado los evaluadores.	
Referencias	REQ-FG05, REQ-FI05, REQ-FM02	
Precondición	El usuario debe haber iniciado sesión.	
Postcondición		
Curso normal de los eventos		
Acción de los actores	Respuesta del sistema	
1. El usuario entra a la sección de mapa.	2. El sistema muestra el mapa.	
3. El usuario entra a la sección de inspecciones realizadas.	4. El sistema muestra la lista de inspecciones realizadas, con diferentes filtros y una barra de búsqueda.	
Cursos alternativos		

Tabla 46: Descripción de caso de uso para la visualización de inspecciones realizadas.

Caso de uso	Visualizar detalles de inspección	
Actores	Administrador, Responsable de institución, Evaluador	
Descripción	Permite ver los detalles de una inspección realizada.	
Referencias	REQ-FI05	
Precondición	El usuario debe haber iniciado sesión e iniciado el caso de uso de visualizar inspecciones o visualizar histórico de edificación.	
Postcondición		
Curso normal de los eventos		
Acción de los actores	Respuesta del sistema	

1. El usuario selecciona una inspección de la lista de inspecciones realizadas.	2. El sistema muestra los detalles de la inspección seleccionada: fecha de realización, instrumento lleno y bandera obtenida.
Cursos alternativos	

Tabla 47: Descripción de caso de uso para visualizar detalles de la inspección.

Caso de uso	Visualizar mapa	
Actores	Administrador, Responsable de institución, Evaluador	
Descripción	Permite visualizar un mapa con las edificaciones a las que se les ha aplicado módulos de evaluación.	
Referencias	REQ-FF9, REQ-FM01, REQ-FM02, REQ-FM03, REQ-FM04, REQ-FM05, REQ-FM06, REQ-FM07, REQ-FM08, REQ-FM09	
Precondición		
Postcondición		
Curso normal de los eventos		
Acción de los actores	Respuesta del sistema	
1. El usuario entra al sistema.	2. El sistema muestra el mapa de zonas.	
Cursos alternativos		

Tabla 48: Descripción de caso de uso para visualización del mapa.

Caso de uso	Visualizar histórico de edificación	
Actores	Administrador, Responsable de institución, Evaluador	
Descripción	Permite ver un registro histórico de una edificación específica.	
Referencias	REQ-FF9	
Precondición	El usuario debe haber iniciado el caso de uso visualizar estado de edificaciones.	
Postcondición		
Curso normal de los eventos		
Acción de los actores	Respuesta del sistema	
1. El usuario selecciona una edificación desde el mapa.	2. El sistema muestra una lista con las inspecciones aplicadas a lo largo del tiempo.	
3. El usuario selecciona una entrada de la lista de inspecciones aplicadas.	4. El sistema muestra los detalles de la instancia temporal del edificio: resultado de los diferentes módulos aplicados al edificio.	
Cursos alternativos		
<ul style="list-style-type: none"> • Línea 3. El usuario sale de la sección: El sistema regresa a la sección del estado de edificaciones. 		

Tabla 49: Descripción de caso de uso para visualizar el histórico de una edificación.

Caso de uso	Visualizar notificaciones y alertas
Actores	Administrador, Responsable de institución, Evaluador
Descripción	Permite visualizar las entradas del panel de notificaciones y alertas.
Referencias	REQ-FG03, REQ-FC02, REQ-FI02
Precondición	

Postcondición	
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El usuario entra a la sección de mapa.	2. El sistema muestra el mapa.
	3. El sistema muestra una alerta visual y cuando se recibe una entrada en el panel de notificaciones.
4. El usuario abre el panel de notificaciones y alertas.	5. El panel muestra una lista con las notificaciones recibidas.
6. El usuario selecciona una notificación para visualizar el contenido completo.	7. El sistema actualiza el estado de la notificación marcándola como leída.
Cursos alternativos	
<ul style="list-style-type: none"> • Línea 6. En portal web: El usuario clikea en cualquier área fuera del panel de notificaciones y alertas: El sistema cierra el panel mostrando sólo el mapa. 	

Tabla 50: Descripción de caso de uso para visualizar notificaciones y alertas.

3.4.3 Modelo de Clases

El diagrama de clases presentará las clases del sistema con sus relaciones estructurales y de herencia. El modelo de casos de uso aportará la información para establecer las clases, atributos y operaciones:

- **Clase:** abstracción que define un tipo de objeto real especificando un conjunto de propiedades o atributos y de comportamiento o funcionalidad.
- **Propiedades:** también llamados atributos o características, son valores que corresponden a un objeto, como color, material, cantidad, ubicación. Generalmente se conoce como la información detallada del objeto. Suponiendo que el objeto es un formulario, sus propiedades serían: tipo de formulario, nombre, etc.
- **Operaciones:** o métodos, son aquellas actividades o verbos que se pueden realizar con este objeto, como por ejemplo abrir, cerrar, buscar, cancelar, acreditar, cargar.

3.4.3.1 Diagrama del Modelo de Clases

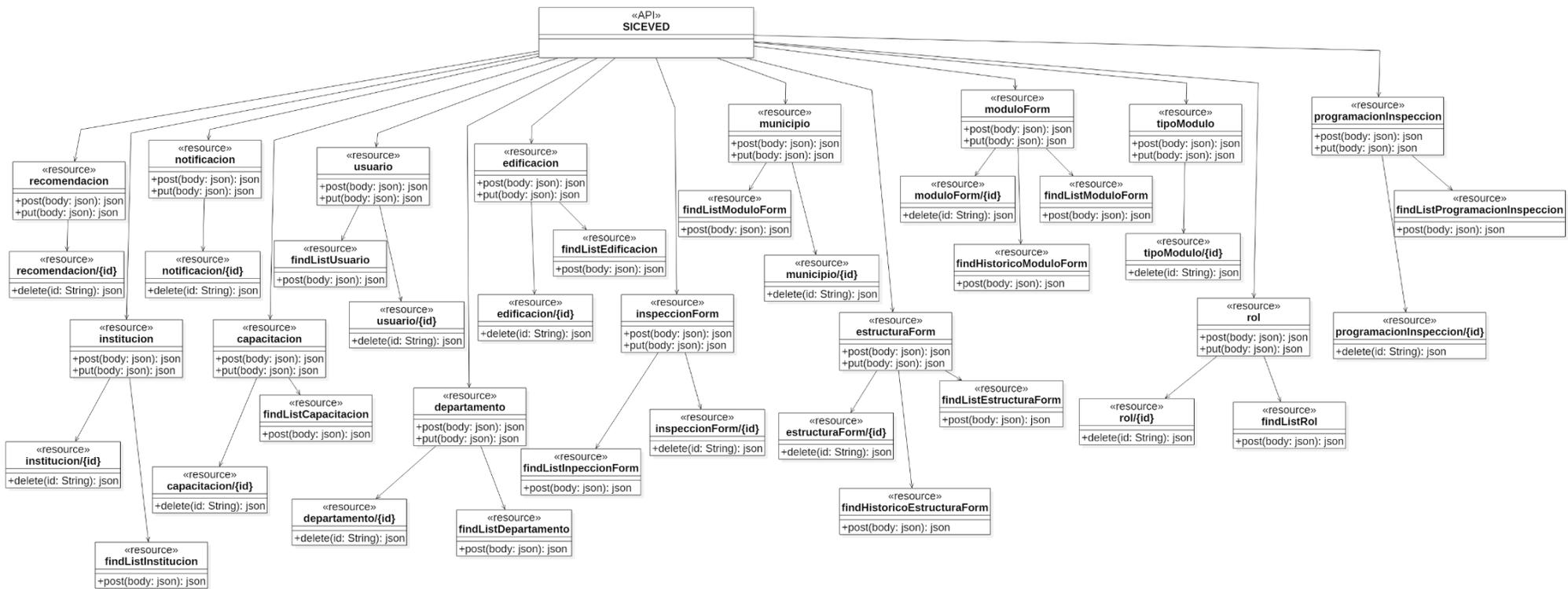


Ilustración 12. Modelo de clases.

CAPÍTULO IV: Diseño

4.1 Introducción

A continuación, se presenta el diseño de datos del sistema según los requerimientos definidos en la etapa de análisis. Iniciando por un análisis amplio de las entidades involucradas con un ER, pasando luego a un modelo físico orientado a documentos.

Se establece la arquitectura de la solución, definiendo primero la arquitectura física, donde se establece tanto el entorno de desarrollo como el de producción. Luego, se define la arquitectura lógica, la cual se expresa con un diagrama de componentes que detalla la interrelación del stack de desarrollo en conjunto con los módulos o librerías utilizadas en cada componente.

Finalmente, se definen los diagramas de navegación de la aplicación web y móvil, y las distintas interfaces aproximadas de cada una de las aplicaciones. Así mismo, se detallan ejemplos de formatos de los reportes que pueden generarse.

4.2 Diseño de Datos

Consiste en identificar y organizar los datos apropiadamente de manera que el acceso a estos sea de manera rápida y eficaz.

En un modelo relacional esto se hace mediante la identificación de las tablas necesarias con sus atributos y las posibles llaves primarias y las relaciones.

En una base de datos NoSQL es de utilidad hacer una comparación entre las relaciones de las tablas de una base de datos relacional y las relaciones entre los distintos documentos, para esto hemos utilizado como punto de partida un diagrama ER el cual nos permite conocer la estructura básica de los documentos necesarios en nuestra base y la manera en que estos se relacionan.

Una vez identificadas las relaciones en el ER se procede al diseño del modelo físico de tal manera que una relación 1 a 1 en una base relacional; en una NoSQL sería un documento embebido o la implementación de una llave foránea como es el caso de la relación entre usuario y dirección²⁰.

²⁰ <http://learnmongodbthehardway.com/> MONGODB SCHEMA DESIGN
<http://learnmongodbthehardway.com/schema/schemabasics/>

```
{
  nombre: "Juan Perez",
  telefonos: ["76547544", "24564455"],
  correos: ["juanperez@gmail.com"],
  direccion: {
    departamento: "San Salvador",
    municipio: "San Salvador",
    descripcion: "Barrio Las Vega, Pje 4, Casa #5"
  }
}
```

Ejemplo: Documento 'direccion' embebido en documento 'usuario'

Es importante destacar que cuando se realizan "relaciones" embebiendo un documento dentro de otro, la relación definida en el diagrama ER ya no es necesaria y por tanto en nuestro diagrama físico desaparece dicha relación.

4.2.1 Diagrama Entidad Relación

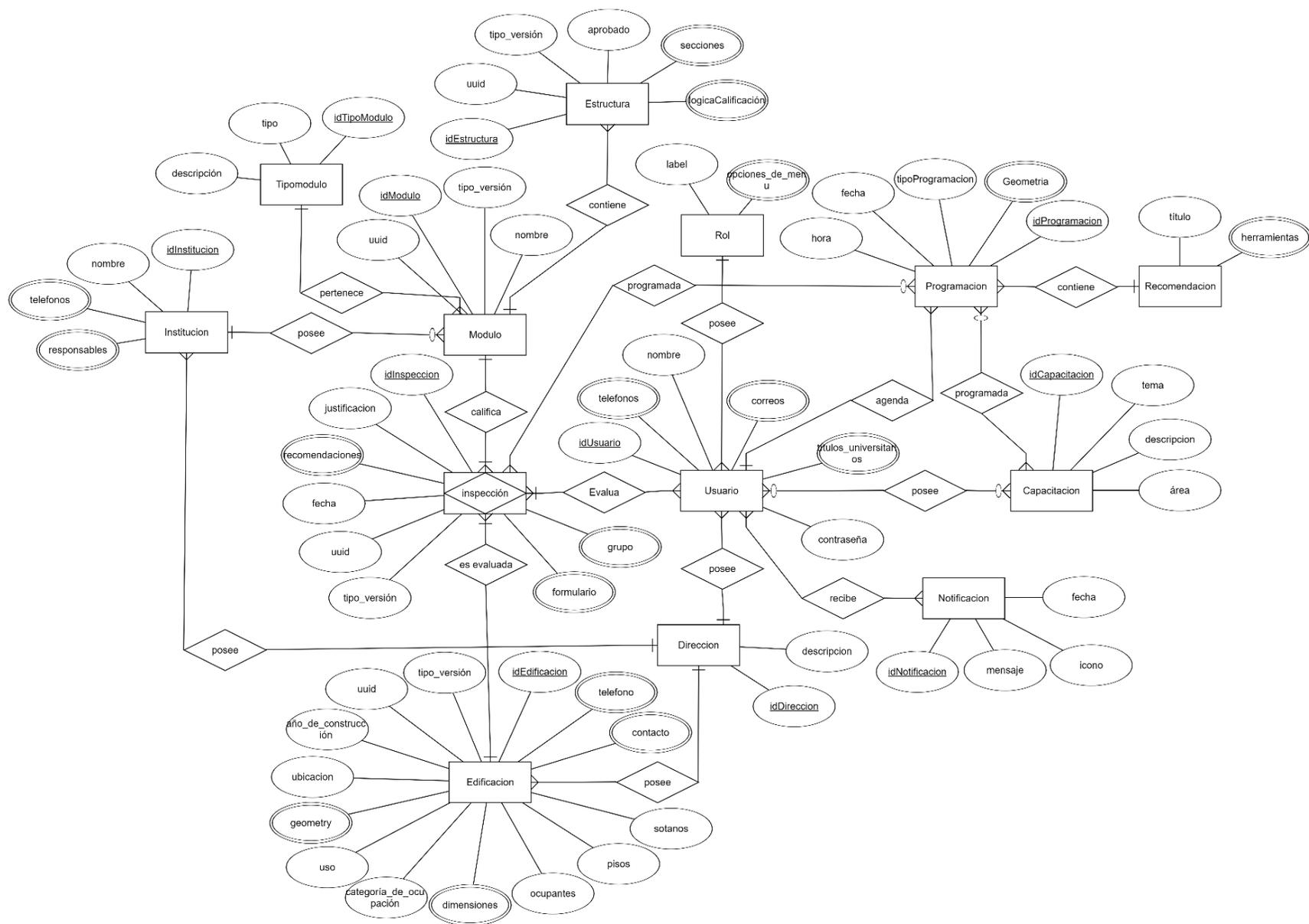


Ilustración 13. Diagrama entidad relación.

4.2.2 Modelo Físico de la Base de Datos

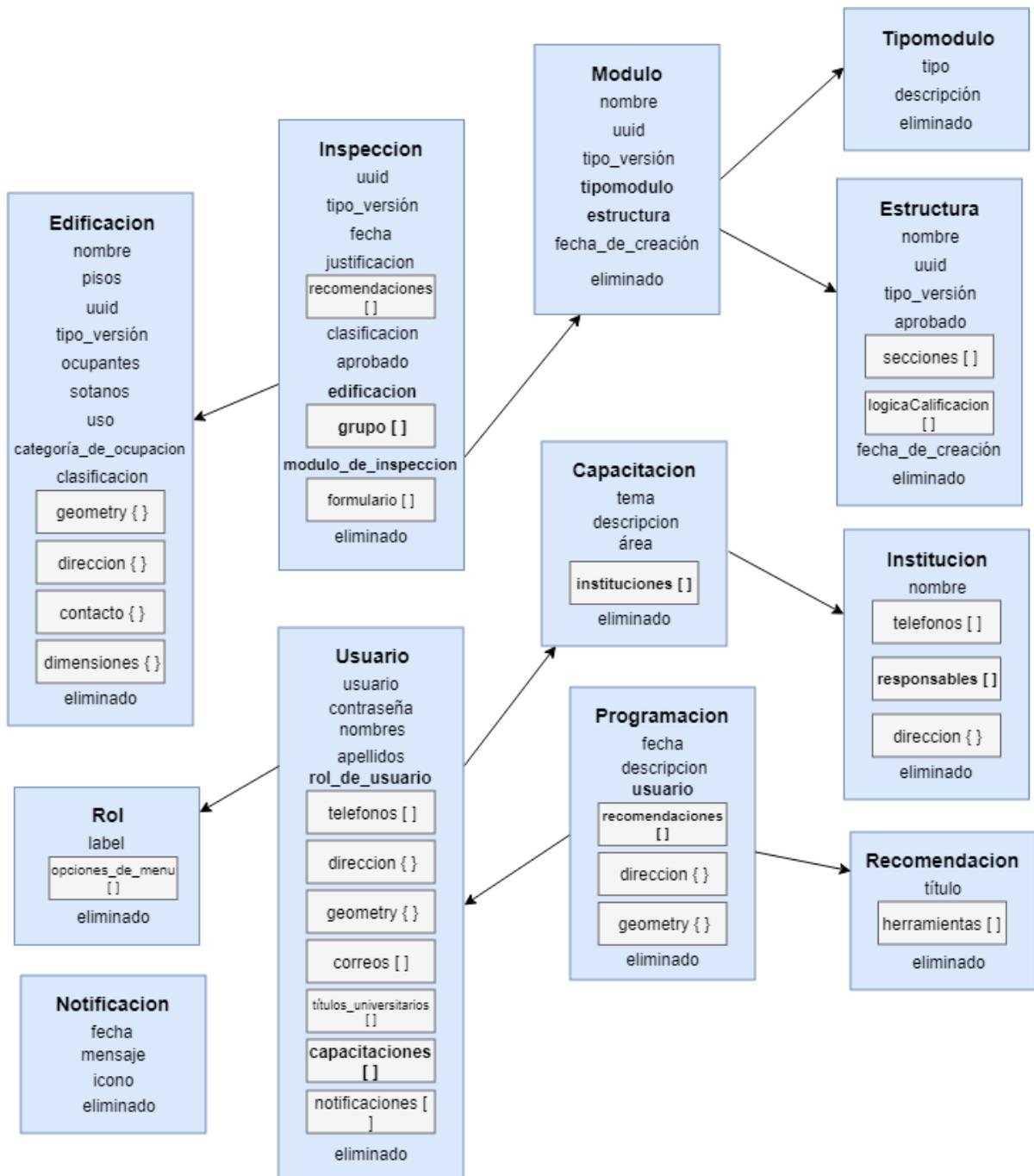


Ilustración 14. Modelo físico de base de datos.

4.2.3 Diccionario de datos

Institución:

Colección de los documentos de instituciones que forman parte del sistema.

- **nombre:** Tipo 'String'. Nombre de la institución.
- **telefonos:** Tipo 'Array'. Arreglo de 'String' de los distintos números de teléfono de la institución.
- **responsables:** Tipo 'Array'. Arreglo de 'ObjectId' de los responsables de la institución.
- **direccion:** Tipo 'Object'. Describe la dirección de la institución. Posee la siguiente estructura:

```
{
  departamento: "San Salvador",
  municipio: "San Marcos",
  descripcion: "Calle Antigua a Zacatecoluca"
}
```
- **eliminado:** Tipo 'Number'. Denota si el documento ha sido eliminado. El valor 1 denota que sí, y el 0 que no. Por defecto es 0.

Modulo:

Colección de documentos de los distintos formularios dentro del sistema.

- **nombre:** Tipo 'String'. Nombre del módulo.
- **uuid:** Tipo 'String'. Identificador único para una línea histórica de la entidad.
- **tipo_versión:** Tipo 'String'. Representa la versión del documento en la línea histórica. Los posibles valores son 'H' (Histórica) y 'A' (Actual).
- **tipomodulo:** Tipo 'ObjectId'. Hace referencia al documento que detalla la categoría del módulo.
- **estructura:** Tipo 'ObjectId'. Referencia al documento de la estructura más actual que se utilizaría en una inspección donde se haya elegido este módulo.
- **fecha_de_creación:** Tipo 'Date'. Fecha y hora en la que se creó la estructura de formulario.
- **eliminado:** Tipo 'Number'. Denota si el documento ha sido eliminado. El valor 1 denota que sí, y el 0 que no. Por defecto es 0.

Tipomodulo:

Colección de documentos de los distintos tipos de módulo del sistema.

- **tipo:** Tipo 'String'. Nombre del tipo de módulo.
- **descripción:** Tipo 'String'. Descripción del tipo de módulo.
- **eliminado:** Tipo 'Number'. Denota si el documento ha sido eliminado. El valor 1 denota que sí, y el 0 que no. Por defecto es 0.

Estructura:

Colección de documentos con las distintas estructuras de los módulos a lo largo del tiempo de acuerdo a los cambios realizados en los campos que componen el formulario, o la lógica de evaluación de los campos. Dichos cambios reflejados en la propiedad incremental 'tipo_versión', que denota la vigencia de la estructura, en conjunto con la aprobación de dicha versión ('aprobado' = true).

- **nombre:** Tipo 'String'. Nombre de la estructura.
- **uuid:** Tipo 'String'. Identificador único para una línea histórica de la entidad.

- **tipo_versión:** Tipo 'String'. Representa la versión del documento en la línea histórica. Los posibles valores son 'H' (Histórica) y 'A' (Actual).
- **aprobado:** Tipo 'Boolean'. Denota la aprobación de una nueva revisión en el documento.
- **secciones:** Tipo 'Array'. Arreglo de 'Object' que describe las secciones y componentes que comprenden la estructura del formulario.
- **logicaCalificacion:** Tipo 'Array'. Arreglo de 'Object' que enlaza la lógica de evaluación y calificación entre las secciones del formulario.
- **fecha_de_creación:** Tipo 'Date'. Fecha y hora en la que se creó la estructura de formulario.
- **eliminado:** Tipo 'Number'. Denota si el documento ha sido eliminado. El valor 1 denota que sí, y el 0 que no. Por defecto es 0.

Inspeccion:

Colección de documentos de las inspecciones realizadas a las edificaciones.

- **uuid:** Tipo 'String'. Identificador único para una línea histórica de la entidad.
- **tipo_versión:** Tipo 'String'. Representa la versión del documento en la línea histórica. Los posibles valores son 'H' (Histórica) y 'A' (Actual).
- **fecha:** Tipo 'Date'. Fecha de la realización de la inspección.
- **justificación:** Tipo 'String'. Justificación del resultado final de la evaluación. Esto es para dar más contexto al color de una bandera y que en el portal público sea más claro para la población.
- **recomendaciones:** Tipo 'Array'. Arreglo de 'String' de cada uno de las recomendaciones al dueño de la edificación en caso de obtener una bandera no favorable.
- **clasificacion:** Tipo 'String'. Indica el color de la bandera obtenida.
- **aprobado:** Tipo 'Boolean'. Denota que la inspección ha sido aprobada.
- **edificacion:** Tipo 'ObjectId'. Referencia al documento de la edificación a la cual se le ha realizado la inspección.
- **grupo:** Tipo 'Array'. Arreglo de 'ObjectId' con los usuarios que llevaron a cabo la inspección.


```
{
  supervisor: ObjectId("5a8252faa501b819b45253de"),
  evaluadores: [
    ObjectId("684282faa501b819b45253de"),
    ObjectId("53d252faa501b819b452r4w7"),
    ObjectId("t7a8252faa501b819b4525de")
  ]
}
```
- **modulo_de_inspeccion:** Tipo 'ObjectId'. Referencia al documento del módulo utilizado para llevar a cabo la inspección.
- **formulario:** Tipo 'Array'. Arreglo de 'Object' de las secciones y componentes de la estructura del formulario con la información recolectada durante la inspección.
- **eliminado:** Tipo 'Number'. Denota si el documento ha sido eliminado. El valor 1 denota que sí, y el 0 que no. Por defecto es 0.

Edificación:

Colección de documentos de los edificios cargados en el sistema.

- **nombre:** Tipo 'String'. Nombre de la edificación.
- **pisos:** Tipo '32-bit integer'. Cantidad de pisos de la edificación.
- **uuid:** Tipo 'String'. Identificador único para una línea histórica de la entidad.

- **tipo_versión:** Tipo 'String'. Representa la versión del documento en la línea histórica. Los posibles valores son 'H' (Histórica) y 'A' (Actual).
- **ocupantes:** Tipo '32-bit integer'. Cantidad de ocupantes de la edificación.
- **sotanos:** Tipo '32-bit integer'. Cantidad de sotanos.
- **uso:** Tipo 'String'. Indica el tipo de uso de la edificación.
- **categoría_de_ocupacion:** Tipo 'String'. Indica la categoría de ocupación de la edificación según la NTDS.
- **clasificación:** Tipo 'String'. El nivel de riesgo de la edificación de acuerdo a la última inspección realizada.
- **geometry:** Tipo 'Object'. Representa la ubicación geográfica de la edificación. Se define con un objeto GeoJSON.


```
{
  type: "Point",
  coordinates: [-89.20294940471649, 13.718285117623909]
}
```
- **dirección:** Tipo 'Object'. Describe la dirección de la edificación. Posee la misma estructura descrita en colección de 'Institución'.
- **año_de_construcción:** Tipo 'Date'. Año de construcción del edificio.
- **contacto:** Tipo 'Object'. Describe información de contacto del responsable de la edificación.


```
{
  nombre: "Raúl Pleitez",
  telefonos: ["72939658", "23649172"],
  correos: ["racsoraul@gmail.com"],
  direccion: {
    departamento: "La Libertad",
    municipio: "Ciudad Arce",
    descripcion: "Col. Las Vegas, Casa #5"
  }
}
```
- **dimensiones:** Tipo 'Object'. Describe las dimensiones de la edificación en metros.


```
{
  frente: 300,
  fondo: 200,
  alturaPiso: 10
}
```
- **eliminado:** Tipo 'Number'. Denota si el documento ha sido eliminado. El valor 1 denota que sí, y el 0 que no. Por defecto es 0.

Programación:

Colección de documentos de las programaciones de jornadas de inspección o de capacitación realizadas.

- **fecha:** Tipo 'Date'. Fecha de realización de la jornada.
- **descripción:** Tipo 'String'. Detalla de lo que tratará la actividad programada.
- **usuario:** Tipo 'ObjectId'. Referencia al documento del usuario que programó la jornada.
- **recomendaciones:** Tipo 'ObjectId'. Hace referencia al documento que describe un conjunto de recomendaciones estándar o habituales.
- **dirección:** Tipo 'Object'. Describe la dirección de la edificación. Posee la misma estructura descrita en colección de 'Institución'.

- **geometry:** Tipo 'Object'. Define la data geoespacial de la zona donde se llevará a cabo la jornada. Se define con un objeto GeoJSON de tipo 'Point' o 'Polygon'.

```
{
  "type": "Polygon",
  "coordinates": [
    [
      [
        -89.20109331607819,
        13.7203514246191
      ],
      [
        -89.20111209154129,
        13.720205506941545
      ], ...
    ]
  ]
}
```

- **eliminado:** Tipo 'Number'. Denota si el documento ha sido eliminado. El valor 1 denota que si, y el 0 que no. Por defecto es 0.

Capacitación:

Colección de documentos de las distintas capacitaciones impartidas.

- **tema:** Tipo 'String'. Tópico del que trata la capacitación.
- **descripción:** Tipo 'String'. Detalle del tema de la capacitación.
- **área:** Tipo 'String'. Área técnica a la que corresponde la capacitación.
- **instituciones:** Tipo 'Array'. Arreglo de 'ObjectId' que referencian al documento o documentos de las instituciones.
- **eliminado:** Tipo 'Number'. Denota si el documento ha sido eliminado. El valor 1 denota que si, y el 0 que no. Por defecto es 0.

Usuario:

Colección de documentos de los usuarios registrados en el sistema.

- **usuario:** Tipo 'String'. Nombre del usuario.
- **contraseña:** Tipo 'String'. Contraseña encriptada.
- **nombres:** Tipo 'String'. Nombres del usuario.
- **apellidos:** Tipo 'String'. Apellidos del usuario.
- **rol_de_usuario:** Tipo 'ObjectId'. Referencia al documento que describe el rol.
- **telefonos:** Tipo 'Array'. Arreglo de 'Object' de los distintos números de teléfono del usuario.
- **direccion:** Tipo 'Object'. Describe la dirección del usuario.
- **geometry:** Tipo 'Object'. Define la data geoespacial de punto donde el usuario es más usual que radique en caso de una emergencia. Se define con un objeto GeoJSON de tipo 'Point'.
- **correos:** Tipo "Array". Arreglo de 'Object' con los distintos correos del usuario.
- **títulos_universitarios:** Tipo "Array". Arreglo de 'Object' con los distintos títulos del usuario.
- **capacitaciones:** Tipo "Array". Arreglo de 'Object' con las distintas capacitaciones del usuario.
- **notificaciones:** Tipo "Array". Arreglo de 'Object' con las distintas notificaciones recibidas por parte del usuario.

- **eliminado:** Tipo 'Number'. Denota si el documento ha sido eliminado. El valor 1 denota que si, y el 0 que no. Por defecto es 0.

Rol:

Colección de documentos de los distintos roles del sistema.

- **label:** Tipo 'String'. Nombre del rol.
- **opciones_de_menu:** Tipo 'Array'. Arreglo de objetos que describe cada opción del menú que corresponde al rol.
- **eliminado:** Tipo 'Number'. Denota si el documento ha sido eliminado. El valor 1 denota que si, y el 0 que no. Por defecto es 0.

Notificación:

Colección de documentos de las distintas notificaciones del sistema.

- **fecha:** Tipo 'Date'. Fecha de emisión de la notificación.
- **mensaje:** Tipo 'String'. Cuerpo de la notificación.
- **icono:** Tipo 'String'. Nombre del icono que se desea mostrar; siguiendo la lista de nombres de iconos de Material Design.
- **eliminado:** Tipo 'Number'. Denota si el documento ha sido eliminado. El valor 1 denota que si, y el 0 que no. Por defecto es 0.

Recomendacion:

Colección de documentos de las distintas recomendaciones del sistema.

- **título:** Tipo 'String'. Título significativo para el conjunto de recomendaciones.
- **herramientas:** Tipo 'Array'. Arreglo de string que detallan cada una de las
- **eliminado:** Tipo 'Number'. Denota si el documento ha sido eliminado. El valor 1 denota que si, y el 0 que no. Por defecto es 0.

4.3 Diseño Arquitectónico

Arquitectura física

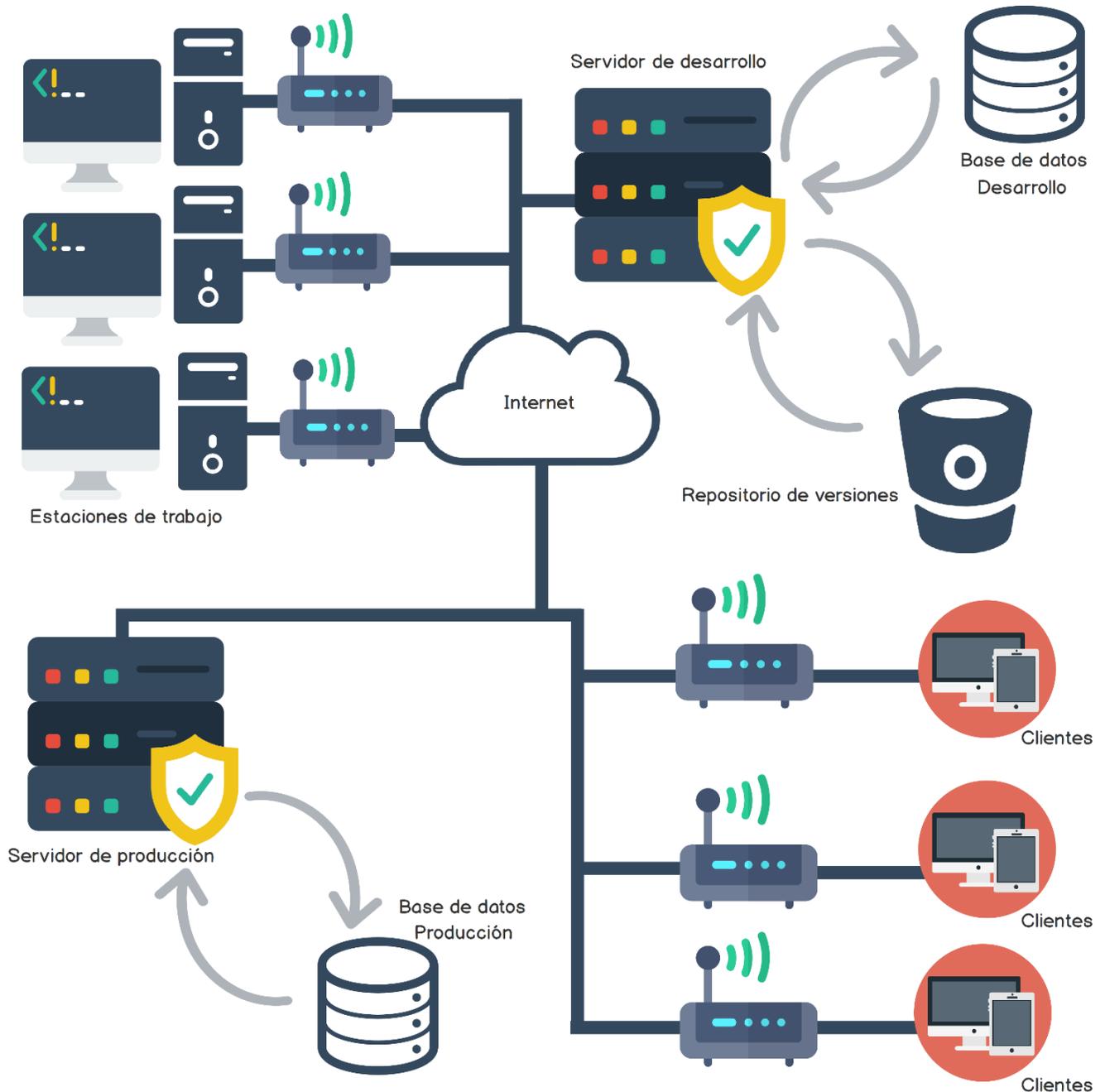


Ilustración 15. Arquitectura física.

- **Repositorio de versiones:** herramienta que facilita la administración de cambios en el desarrollo de un proyecto informático.
- **Servidor de producción:** servidor el cual se pone en marcha el sistema desarrollado libre de errores y terminado.
- **Servidor de desarrollo:** servidor utilizado en la fase de desarrollo en el cual se realizan las pruebas al sistema que se está construyendo.
- **Estación de trabajo:** computadora del desarrollador del sistema.
- **Cliente:** usuarios que serán capaces de acceder al sistema vía navegador web y aplicación móvil.

Arquitectura lógica

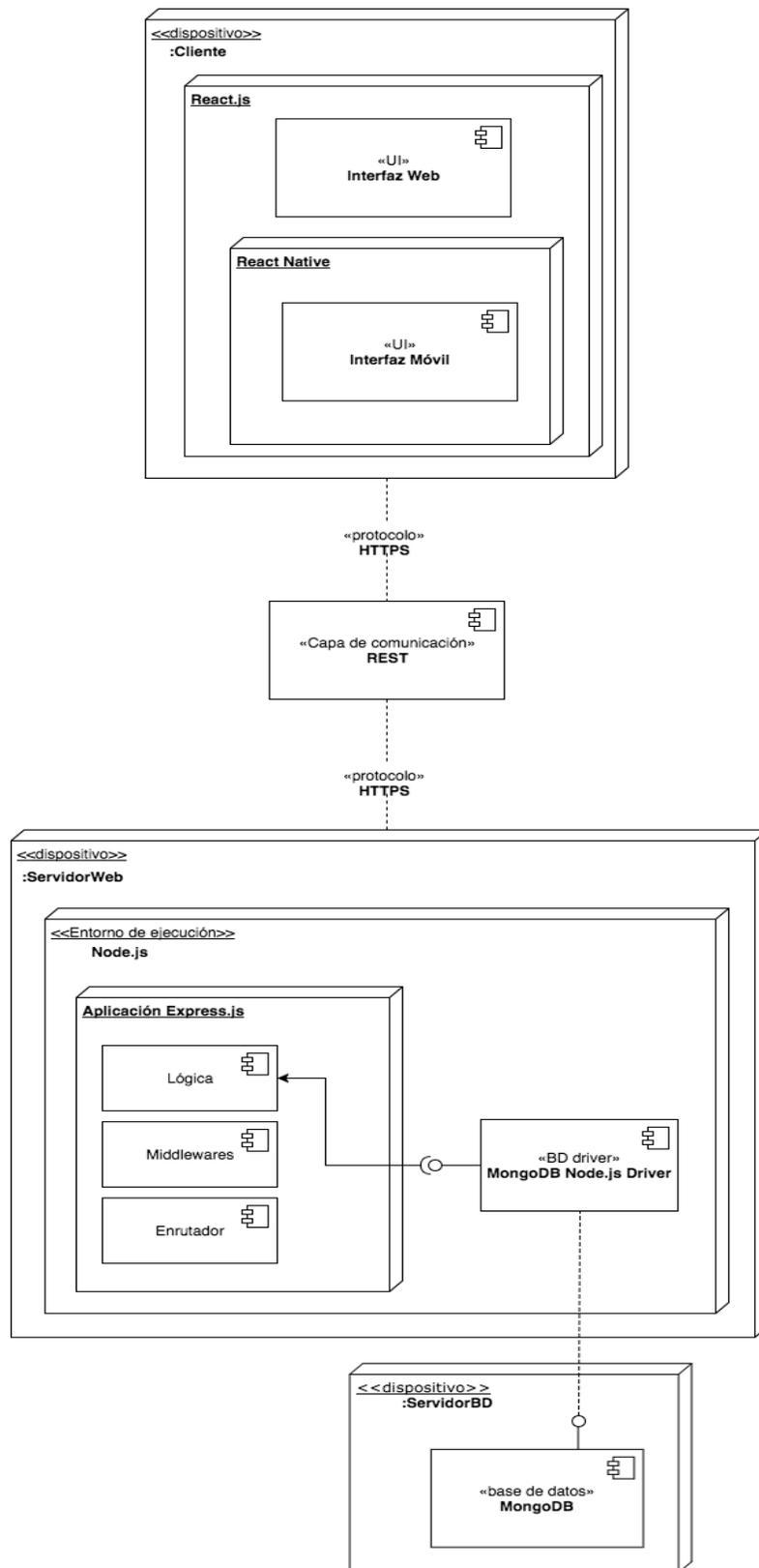


Ilustración 16. Arquitectura lógica.

4.4 Diseño de Interfaces

4.4.1 Diagrama de navegación vistas web

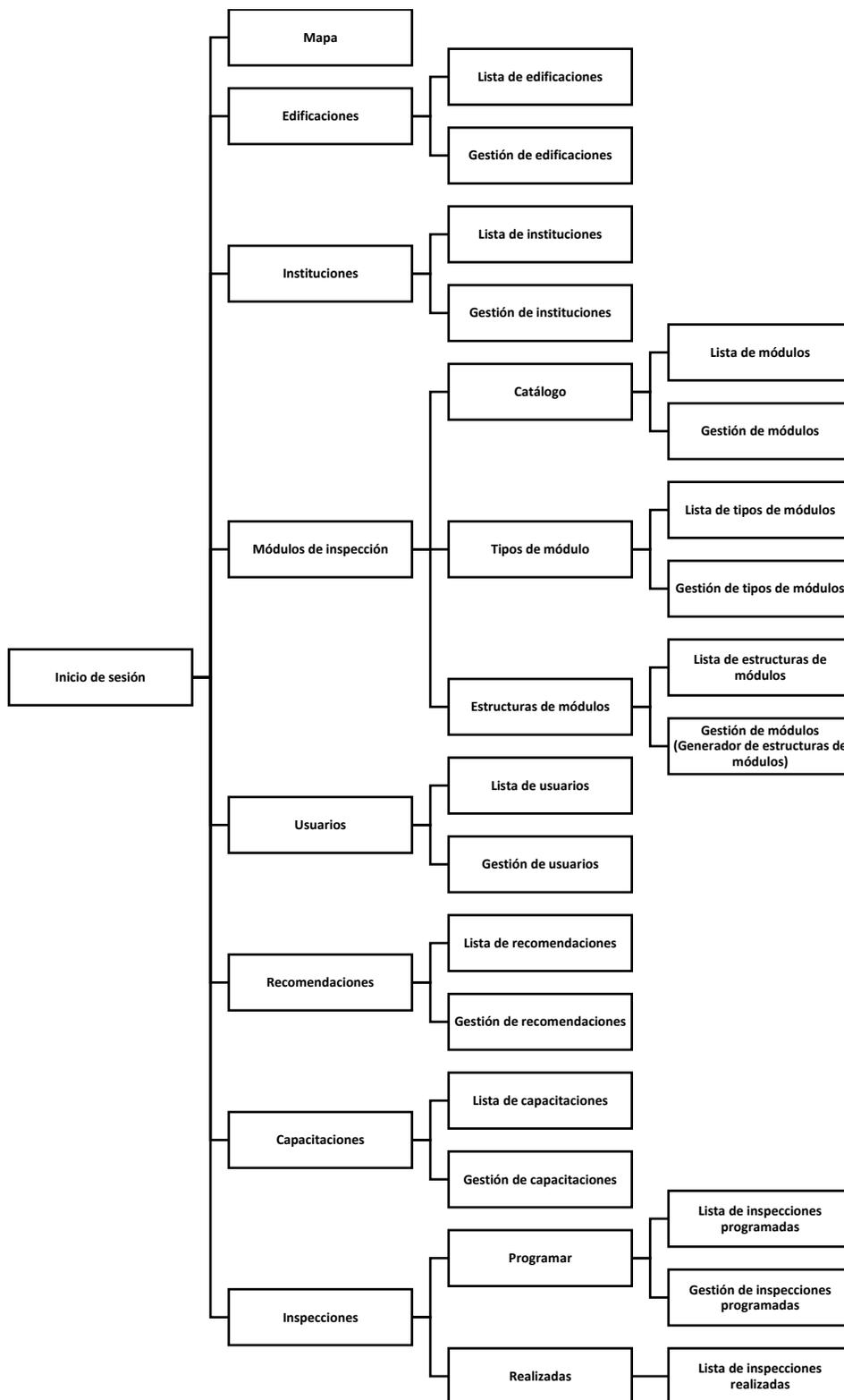


Ilustración 17. Diagrama de navegación web.

4.4.2 Diagrama de navegación vistas móviles

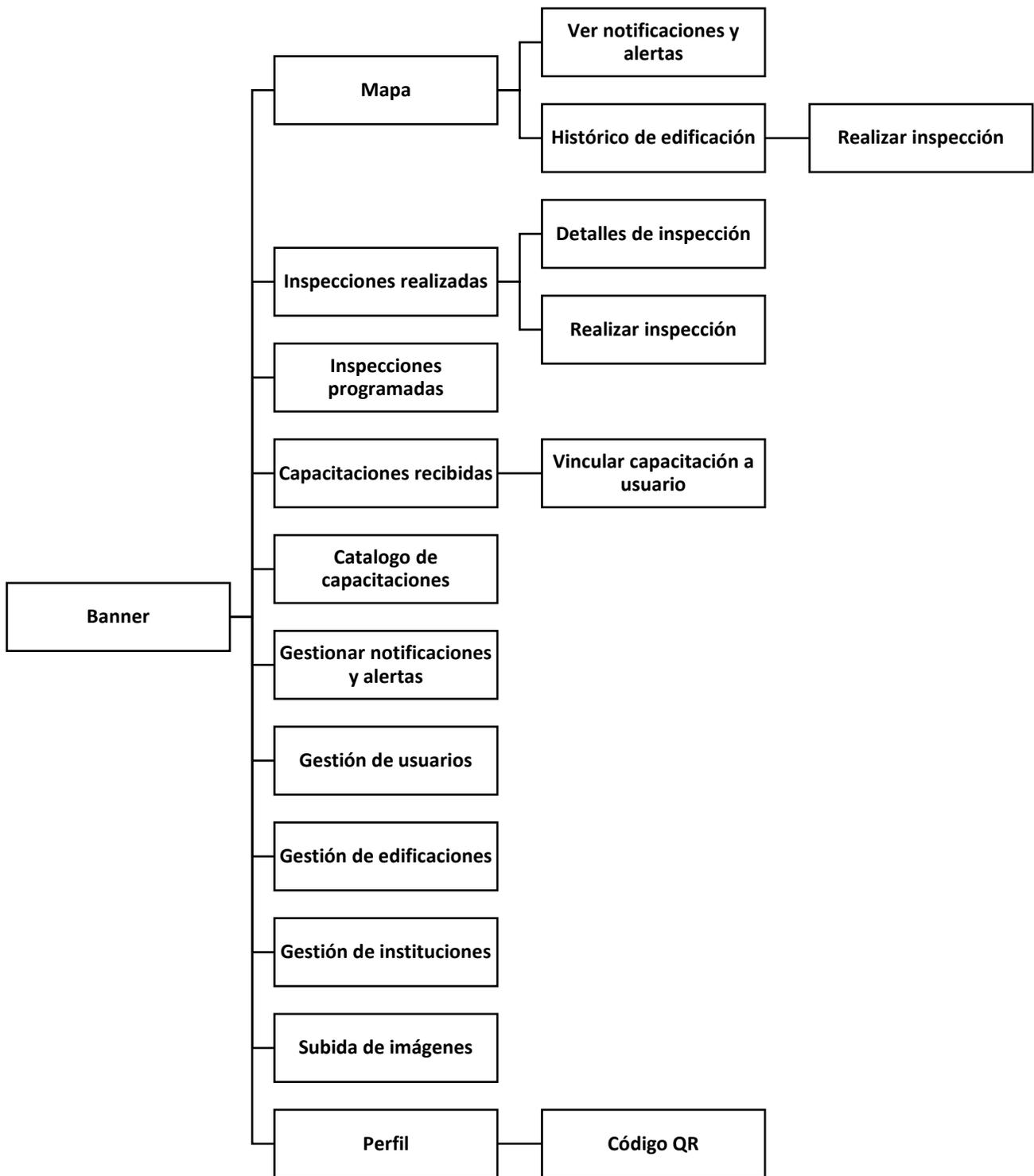


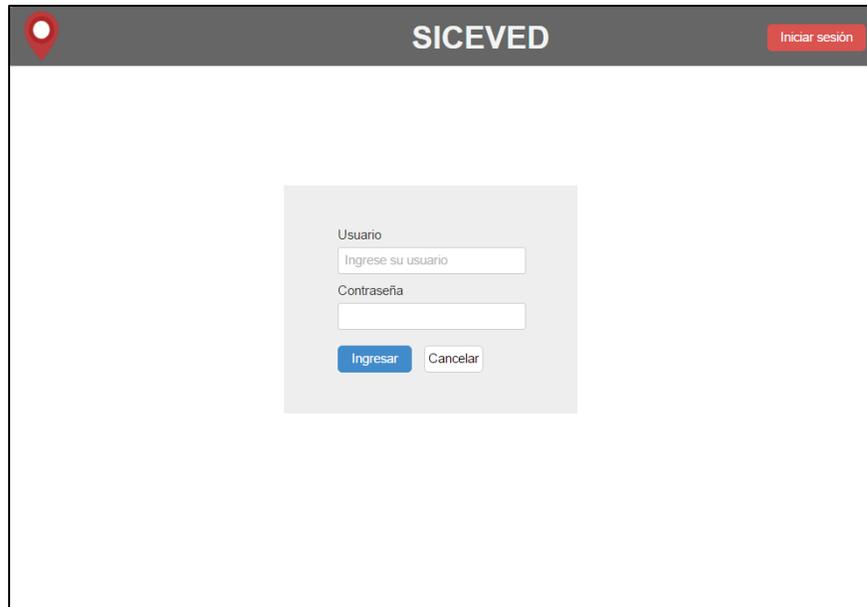
Ilustración 18. Diagrama de navegación móvil.

El diagrama de navegación de vistas móviles contempla todas las vistas existentes. Para ver el detalle de los niveles de acceso por roles, ver el Manual de Usuario Móvil, el cual se encuentra en el disco contenedor de los recursos en formato digital del trabajo de grado.

4.4.3 Vistas

4.4.3.1 Vistas Web

Inicio de sesión



The screenshot shows the SICEVED login interface. At the top, there is a header with a location pin icon on the left, the text "SICEVED" in the center, and a red "Iniciar sesión" button on the right. Below the header is a large white area containing a login form. The form has two input fields: "Usuario" with the placeholder text "Ingrese su usuario" and "Contraseña". Below these fields are two buttons: a blue "Ingresar" button and a white "Cancelar" button.

Ilustración 19. Inicio de sesión, vista web.

Mapa con zonas geográficas

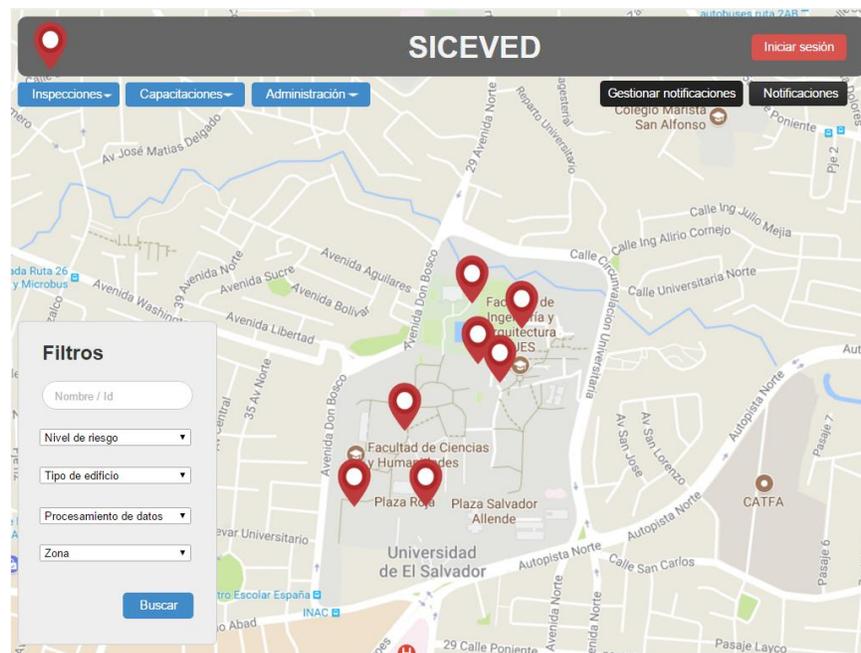


Ilustración 20. Mapa, vista web.

Gestión de edificios

SICEVED Iniciar sesión

Inspecciones Capacitaciones Administración Gestionar notificaciones Notificaciones

Ingrese parametros de busqueda Buscar

Código	Nombre	Última actualización	Opciones
E01	Edificación X	07/10/2017	Editar perfil
E02	Edificación X	07/10/2017	Editar perfil
E03	Edificación X	07/10/2017	Editar perfil
E04	Edificación X	07/10/2017	Editar perfil
E05	Edificación X	07/10/2017	Editar perfil
E06	Edificación X	07/10/2017	Editar perfil
E07	Edificación X	07/10/2017	Editar perfil
E08	Edificación X	07/10/2017	Editar perfil
E09	Edificación X	07/10/2017	Editar perfil
E10	Edificación X	07/10/2017	Editar perfil
E11	Edificación X	07/10/2017	Editar perfil
E12	Edificación X	07/10/2017	Editar perfil
E13	Edificación X	07/10/2017	Editar perfil

1 2 3 4 5 »

Salir

Ilustración 21. Edificios, vista web.

Gestión de usuarios

SICEVED Iniciar sesión

Inspecciones Capacitaciones Administración Gestionar notificaciones Notificaciones

Ingrese parametros de busqueda Buscar

- Nombre de usuario
Tipo de usuario
Capacitacion 01, Capacitación 02, Capacitación 03, Capacit.....
Último inicio de sesión 15m ☆
- Nombre de usuario
Tipo de usuario
Capacitacion 01, Capacitación 02, Capacitación 03, Capacit.....
Último inicio de sesión 15m ☆
- Nombre de usuario
Tipo de usuario
Capacitacion 01, Capacitación 02, Capacitación 03, Capacit.....
Último inicio de sesión 15m ☆
- Nombre de usuario
Tipo de usuario
Capacitacion 01, Capacitación 02, Capacitación 03, Capacit.....
Último inicio de sesión 15m ☆

1 2 3 4 5 »

Agregar Eliminar Actualizar Salir

Ilustración 22. Usuarios, vista web.

Gestión de instituciones

Código	Nombre	Rubro	Opciones
Inst00	ASIA	Rubro00	Vincular responsable
Inst00	CASALCO	Rubro00	Vincular responsable
Inst00	Institucion00	Rubro00	Vincular responsable
Inst00	Institucion00	Rubro00	Vincular responsable
Inst00	Institucion00	Rubro00	Vincular responsable
Inst00	Institucion00	Rubro00	Vincular responsable
Inst00	Institucion00	Rubro00	Vincular responsable
Inst00	Institucion00	Rubro00	Vincular responsable
Inst00	Institucion00	Rubro00	Vincular responsable
Inst00	Institucion00	Rubro00	Vincular responsable
Inst00	Institucion00	Rubro00	Vincular responsable
Inst00	Institucion00	Rubro00	Vincular responsable
Inst00	Institucion00	Rubro00	Vincular responsable

Ilustración 23. Instituciones, vista web.

Gestión de módulos de inspección

Codigo	Modulo	Institución que lo imparte	Area	Número de veces
Codigo00	Modulo00	ASIA	Area00	5
Inst00	Modulo00	CASALCO	Area00	5
Inst00	Modulo00	Institucion00	Area00	5
Inst00	Modulo00	Institucion00	Area00	5
Inst00	Modulo00	Institucion00	Area00	5
Inst00	Modulo00	Institucion00	Area00	5
Inst00	Modulo00	Institucion00	Area00	5
Inst00	Modulo00	Institucion00	Area00	5
Inst00	Modulo00	Institucion00	Area00	5
Inst00	Modulo00	Institucion00	Area00	5
Inst00	Modulo00	Institucion00	Area00	5
Inst00	Modulo00	Institucion00	Area00	5
Inst00	Modulo00	Institucion00	Area00	5

Ilustración 24. Inspecciones, vista web.

Maquetar módulo



Ilustración 25. Maquetar módulo, vista web.

Catálogo de capacitaciones

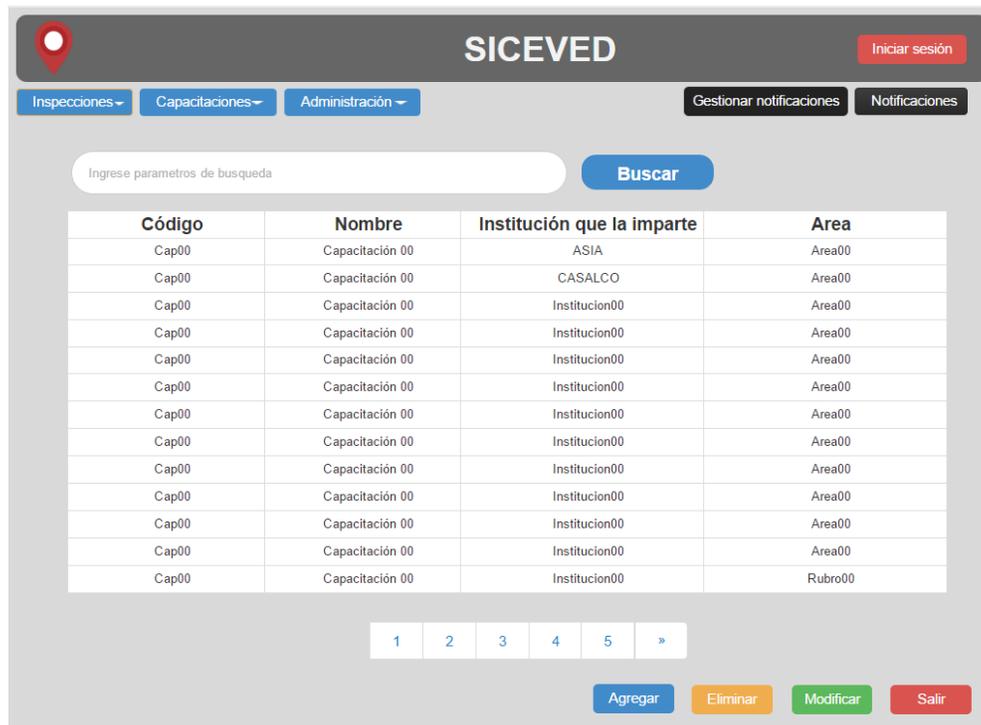


Ilustración 26. Catalogo de capacitaciones, vista web.

Capacitaciones programadas

SICEVED Iniciar sesión

Inspecciones - Capacitaciones - Administración - Gestionar notificaciones Notificaciones

Ingrese parametros de busqueda Buscar

Fecha y hora	Codigo	Nombre	Institución	Area
07/10/2017 13:00	Cap00	Capacitación 00	CASALCO	Ing. Estructural
07/10/2017 13:00	Cap00	Capacitación 00	ASIA	Ing. Estructural
07/10/2017 13:00	Cap00	Capacitación 00	ASIA	Ing. Estructural
07/10/2017 13:00	Cap00	Capacitación 00	CASALCO	Ing. Estructural
07/10/2017 13:00	Cap00	Capacitación 00	CASALCO	Ing. Estructural
07/10/2017 13:00	Cap00	Capacitación 00	ASIA	Ing. Estructural
07/10/2017 13:00	Cap00	Capacitación 00	CASALCO	Ing. Estructural
07/10/2017 13:00	Cap00	Capacitación 00	ASIA	Ing. Estructural
07/10/2017 13:00	Cap00	Capacitación 00	ASIA	Ing. Estructural
07/10/2017 13:00	Cap00	Capacitación 00	ASIA	Ing. Estructural
07/10/2017 13:00	Cap00	Capacitación 00	ASIA	Ing. Estructural
07/10/2017 13:00	Cap00	Capacitación 00	ASIA	Ing. Estructural
07/10/2017 13:00	Cap00	Capacitación 00	CASALCO	Ing. Estructural

1 2 3 4 5 »

Agregar Eliminar Modificar Salir

Ilustración 27. Capacitaciones programadas, vista web.

Inspecciones programadas

SICEVED Iniciar sesión

Inspecciones - Capacitaciones - Administración - Gestionar notificaciones Notificaciones

Ingrese parametros de busqueda Buscar

Fecha	Edificación	Evaluadores	Módulo	Tipo
07/10/2017	Edificio00	Evaluador X, Evaluador Y	CASALCO	Presismo
07/10/2017	Edificio01	Evaluador Z	ASIA	Presismo
07/10/2017	Edificio02	Evaluador X, Evaluador Y	ASIA	Postsismo
07/10/2017	Edificio03	Evaluador Z	CASALCO	Presismo
07/10/2017	Edificio04	Evaluador X, Evaluador Y	CASALCO	Postsismo
07/10/2017	Edificio05	Evaluador Z	ASIA	Postsismo
07/10/2017	Edificio06	Evaluador X, Evaluador Y, Evaluador Z	CASALCO	Presismo
07/10/2017	Edificio07	Evaluador X, Evaluador Y	ASIA	Postsismo
07/10/2017	Edificio08	Evaluador X, Evaluador Y, Evaluador Z	ASIA	Presismo
07/10/2017	Edificio09	Evaluador Z	ASIA	Presismo
07/10/2017	Edificio10	Evaluador X, Evaluador Y, Evaluador Z	ASIA	Postsismo
07/10/2017	Edificio11	Evaluador X, Evaluador Y	ASIA	Postsismo
07/10/2017	Edificio12	Evaluador X, Evaluador Y	CASALCO	Presismo

1 2 3 4 5 »

Agregar Eliminar Actualizar Salir

Ilustración 28. Inspecciones programadas, vista web.

Inspecciones realizadas

Fecha	Edificio	Clasificación	Evaluadores	Módulo	Tipo
07/10/2017	Edificio00	Rojo	Evaluador X, Evaluador Y	CASALCO	Presismo
07/10/2017	Edificio01	Amarillo	Evaluador Z	ASIA	Presismo
07/10/2017	Edificio02	Amarillo	Evaluador X, Evaluador Y	ASIA	Postsismo
07/10/2017	Edificio03	Rojo	Evaluador Z	CASALCO	Presismo
07/10/2017	Edificio04	Verde	Evaluador X, Evaluador Y	CASALCO	Postsismo
07/10/2017	Edificio05	Verde	Evaluador Z	ASIA	Postsismo
07/10/2017	Edificio06	Verde	Evaluador X, Evaluador Y, Evaluador Z	CASALCO	Presismo
07/10/2017	Edificio07	Rojo	Evaluador X, Evaluador Y	ASIA	Postsismo
07/10/2017	Edificio08	Verde	Evaluador X, Evaluador Y, Evaluador Z	ASIA	Presismo
07/10/2017	Edificio09	Amarillo	Evaluador Z	ASIA	Presismo
07/10/2017	Edificio10	Rojo	Evaluador X, Evaluador Y, Evaluador Z	ASIA	Postsismo
07/10/2017	Edificio11	Verde	Evaluador X, Evaluador Y	ASIA	Postsismo
07/10/2017	Edificio12	Rojo	Evaluador X, Evaluador Y	CASALCO	Presismo

Ilustración 29. Inspecciones realizadas, vista web.

Detalles de inspección

Estructuras de concreto

Inspección interna

- Los muros en ambas direcciones no están correctamente alineados con los ejes ortogonales principales 0 2
- Los muros en una dirección no están correctamente alineados con el eje ortogonal principal 0 4
- Los muros están correctamente alineados con los ejes ortogonales principales 0 1
- Los muros en ambas direcciones no están correctamente alineados con los ejes ortogonales principales 0 2
- Los muros en una dirección no están correctamente alineados con el eje ortogonal principal 0 4
- Los muros están correctamente alineados con los ejes ortogonales principales 0 1

1 2 3 4 5 »

Ver resultado Actualizar Aprobar Salir

Ilustración 30. Detalles de inspección.

Gestionar notificaciones y alertas

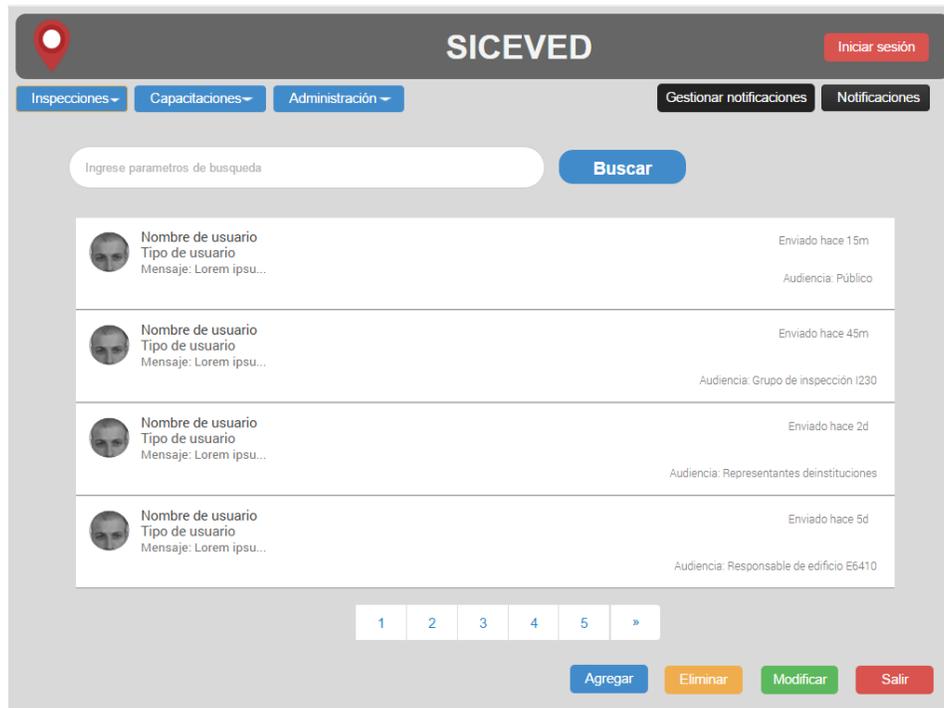


Ilustración 31. Gestión de notificaciones, vista web.

Panel de notificaciones y alertas

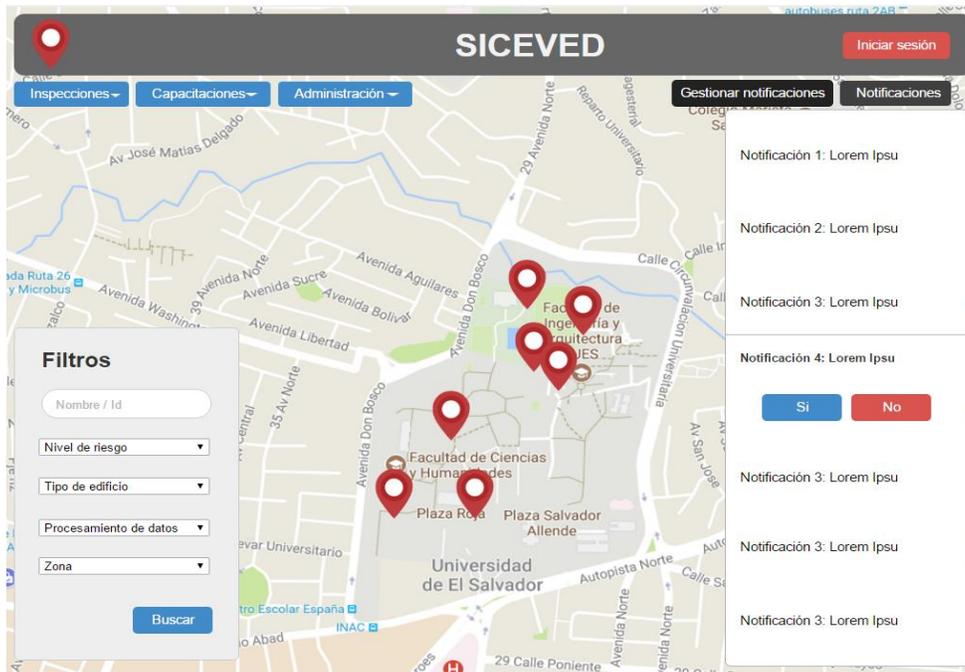


Ilustración 32. Panel de notificaciones, vista web.

Visualizar estado de edificaciones

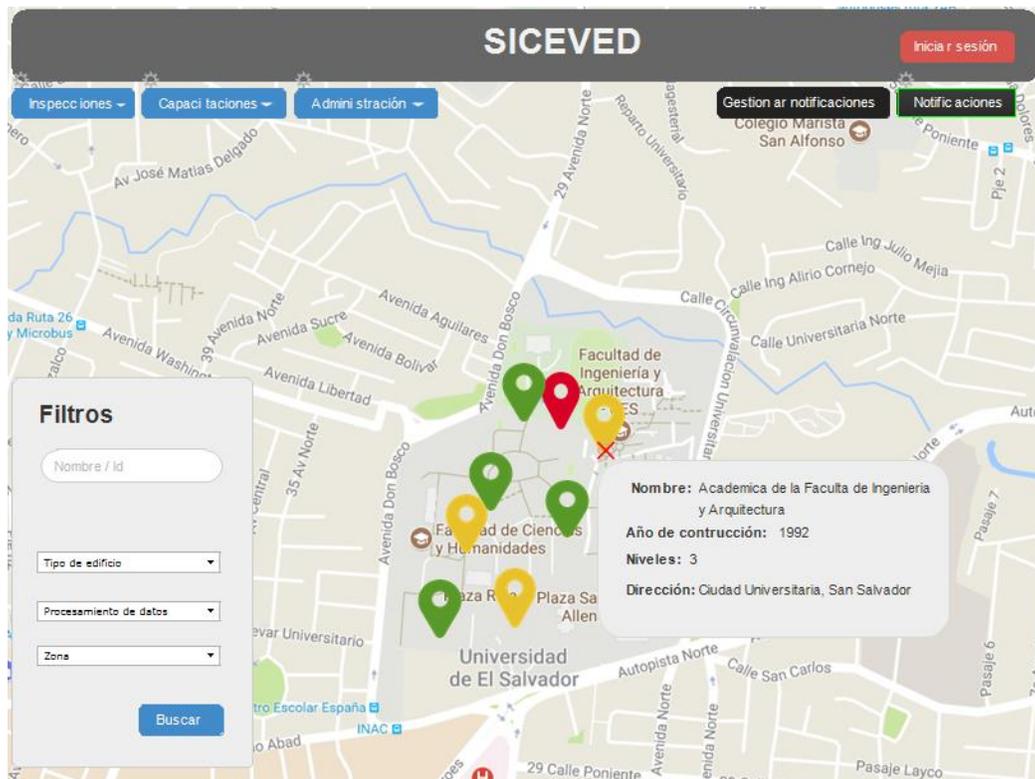


Ilustración 33. Estado de edificación, vista web.

Gestión de usuarios



Ilustración 36. Gestión de usuarios, vista móvil.

Nuevo usuario

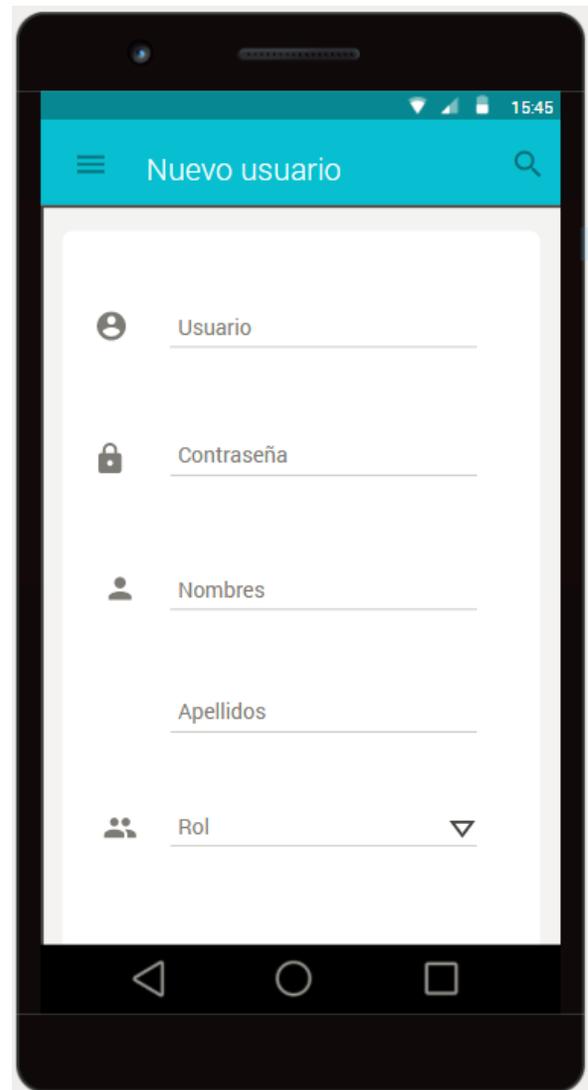


Ilustración 37. Nuevo usuario, vista móvil.

Nueva inspección

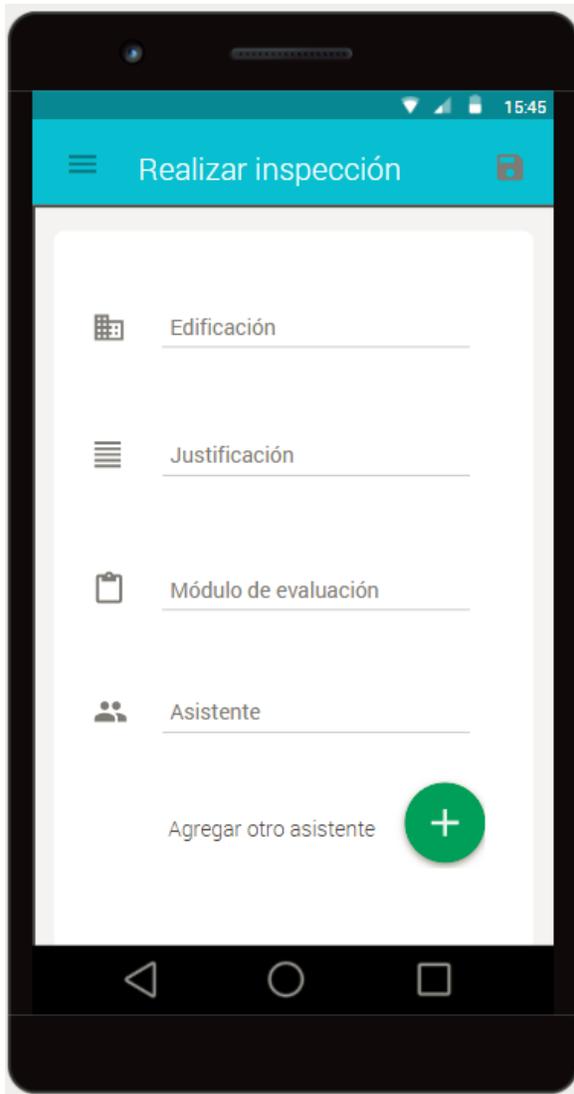


Ilustración 38. Nueva inspección, vista móvil.

Código QR



Ilustración 39. Código QR, vista móvil.

Realizar inspección

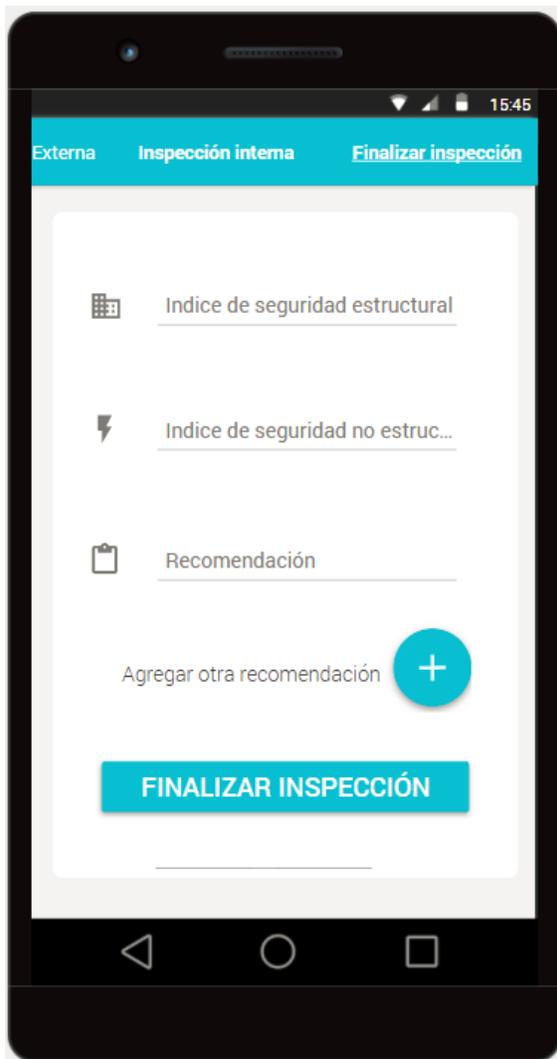


Ilustración 40. Realizar inspección, vista móvil.

Editor de polígonos

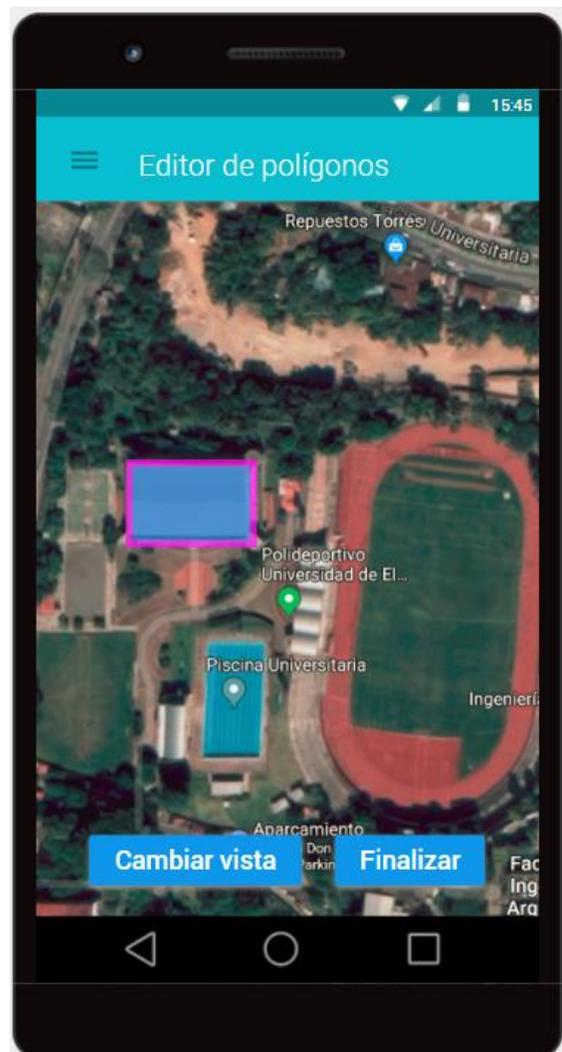


Ilustración 41. Editor de polígonos, vista móvil.

Entrada de imágenes

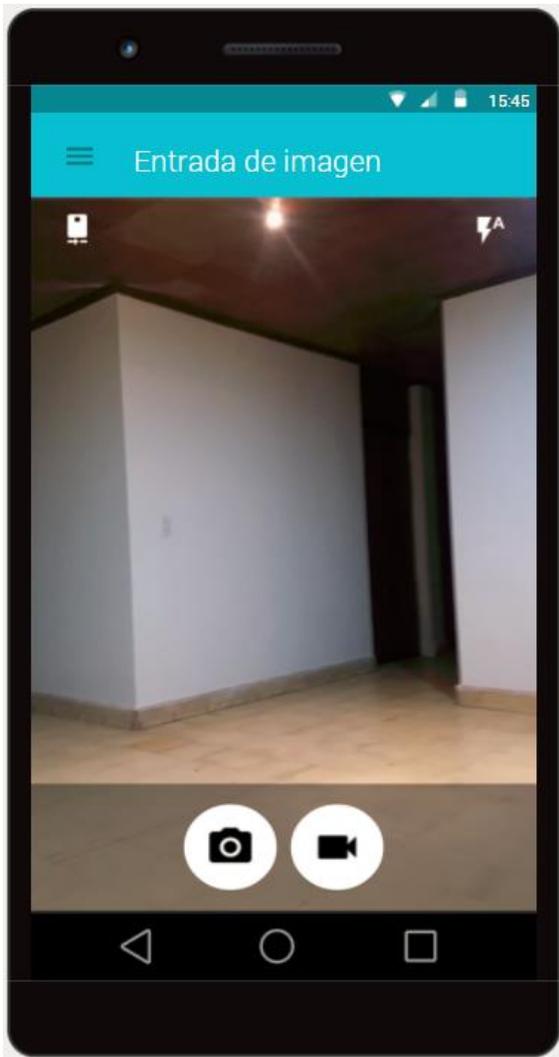


Ilustración 42. Entrada de imágenes, vista móvil.

Histórico de edificación



Ilustración 43. Histórico, vista móvil.

4.4.3.3 Reportes

Inspecciones Realizadas

Fecha	Edificación	Clasificación	Evaluadores	Módulo	Tipo
07/10/2017	Edificio00	Rojo	Evaluador X, Evaluador Y	CASALCO	Presismo
07/10/2017	Edificio01	Amarillo	Evaluador Z	ASIA	Presismo
07/10/2017	Edificio02	Amarillo	Evaluador X, Evaluador Y	ASIA	Postsismo
07/10/2017	Edificio03	Rojo	Evaluador Z	CASALCO	Presismo
07/10/2017	Edificio04	Verde	Evaluador X, Evaluador Y	CASALCO	Postsismo
07/10/2017	Edificio05	Verde	Evaluador Z	ASIA	Postsismo
07/10/2017	Edificio06	Verde	Evaluador X, Evaluador Y, Evaluador Z	CASALCO	Presismo
07/10/2017	Edificio07	Rojo	Evaluador X, Evaluador Y	ASIA	Postsismo
07/10/2017	Edificio08	Verde	Evaluador X, Evaluador Y, Evaluador Z	ASIA	Presismo
07/10/2017	Edificio09	Amarillo	Evaluador Z	ASIA	Presismo
07/10/2017	Edificio10	Rojo	Evaluador X, Evaluador Y, Evaluador Z	ASIA	Postsismo
07/10/2017	Edificio11	Verde	Evaluador X, Evaluador Y	ASIA	Postsismo
07/10/2017	Edificio12	Rojo	Evaluador X, Evaluador Y	CASALCO	Presismo

Ilustración 44. Reporte de inspecciones.

Listado de Capacitaciones

Tema	Descripción	Área	Impartido
Tema01	Descripción01	Área01	4
Tema02	Descripción02	Área02	2
Tema03	Descripción03	Área02	1
Tema04	Descripción04	Área01	4
Tema05	Descripción05	Área01	3

Ilustración 45. Listado de capacitaciones.

Listado de Módulos

LOGO		SICEVED		04/11/2017	
		Listado de Módulos		Página 1 de 1	
Generado por: Juan Pérez					
Creado	Nombre	Institución	Tipo		
01/09/2017	Módulo01	CASALCO	Tipo01		
13/09/2017	Módulo02	ASIA	Tipo02		
01/09/2017	Módulo03	ASIA	Tipo01		
13/09/2017	Módulo04	CASALCO	Tipo01		
13/09/2017	Módulo05	ASIA	Tipo02		

1

Ilustración 46. Listado de módulos.

Perfil de Edificaciones

LOGO		SICEVED		04/11/2017				
		Perfil de Edificios		Página 1 de 10				
Generado por: Juan Perez								
Nombre	Zona	Ubicación	Contacto	Dirección	Pisos	Ocupantes	Uso	Clasificación
Edificio01	Zona02	23.9729, -94.19626	Kiyada Montoya	P.O. Box 378, 4604 Elementum Ave	1	3	Vivienda	verde
Edificio02	Zona05	-80.51631, 75.42629	Lois Kaufman	6380 Erat Av.	1	50	Oficina	rojo
Edificio03	Zona02	65.37926, -137.08318	Remedios Barron	4976 Donec Ave	2	25	Oficina	verde
Edificio04	Zona03	-33.84071, -107.43558	Evelyn Griffin	8901 Vel Avenue	4	200	Hospital	rojo
Edificio05	Zona01	8.02283, 13.92043	Robin Serrano	P.O. Box 655, 5050 Laoreet, Av.	3	300	Escuela	amarillo
Edificio06	Zona02	-72.71049, -105.88312	Cathleen Mann	339-7821 Mauris Rd.	1	75	Oficina	verde
Edificio07	Zona10	29.19024, 25.40018	Victoria Shepard	P.O. Box 110, 6576 Accumsan Rd.	2	400	Hospital	rojo
Edificio08	Zona12	-80.00806, 171.48398	Kiona Aguilar	6146 Ante St.	4	3	Vivienda	verde
Edificio09	Zona02	65.37926, -137.08318	Anastasia Cannon	Ap #784-7078 Et Av.	5	5	Vivienda	verde

1

Ilustración 47. Listado de edificaciones.

Usuarios



LOGO

SICEVED

Usuarios Registrados

04/11/2017

Página 1 de 1

Generado por: Juan Perez

Nombre	Dirección	Teléfono	Institución	Tipo
Jesse Roberson	729-9658 Lobortis. Rd.	7422-7700	ASIA	Responsable de institución
Rhonda U. Freeman	P.O. Box 816, 3802 Ac, Avenue	9567-235	ASIA	Coordinador
Denise Ochoa	P.O. Box 655, 5050 Laoreet, Av	6158-6983	ASIA	Supervisor
Miriam Melendez	7411 Mauris. St.	2428-7281	ASIA	Evaluador
Judith V. Weeks	Ap #784-7078 Et Av.	2495-2029	ASIA	Auxiliar
Victoria T. Myers	Ap #704-8179 Maecenas Avenue	6883-4116	CASALCO	Responsable de institución
Isaac Y. James	Ap #467-2531 Tellus Av.	7591-5756	CASALCO	Coordinador
Claire Joseph	P.O. Box 378, 4604 Elementum Ave	7148-5909	CASALCO	Evaluador
Sophia U. Savage	639-218 Odio Rd.	7904-2850	CASALCO	Evaluador

1

Ilustración 48. Usuarios registrados.

CAPÍTULO V: Implementación

5.1 Tecnologías a utilizar

El listado completo de las tecnologías con sus respectivas versiones a utilizar se encuentra en la sección de los requerimientos de desarrollo en la Sección 3.2.1 Requerimientos tecnológicos.

5.2 Definición de estándares

Los estándares sirven para normar y definir los requisitos que deben de cumplir todos los procesos que se llevan a cabo para el desarrollo de un proyecto, en nuestro caso los estándares utilizados nos brindan la norma a seguir tanto en el desarrollo del sistema como en la presentación de la información que será brindada a los usuarios, teniendo en mente la facilidad de uso y aprendizaje del usuario final del sistema.

Los estándares de desarrollo que se utilizaran nos permiten mantener un orden a pesar de que no es una sola la persona encargada del desarrollo del sistema.

5.2.1 Programación

Se utilizan nombres mnemónicos para las clases, atributos, constantes, variables y objetos.

Se utilizan solamente sustantivos y verbos en caso de que sus nombres consten de más de una palabra (no se utilizan adverbios ni preposiciones).

Se utilizan comentarios para describir los métodos, funciones y procedimientos. Cada archivo se comenta al inicio para describir su propósito.

Los estándares de codificación tienen los objetivos siguientes:

- Crear una apariencia coherente en el código, para que los lectores puedan centrarse en el contenido, no en el diseño.
- Permiten a los lectores comprender el código más rápidamente al hacer suposiciones basadas en la experiencia anterior.
- Facilitan la copia, el cambio y el mantenimiento del código.

Clases

Si constan de una palabra, la primera letra en mayúscula y las demás en minúscula.

Si el nombre de una clase consta de dos palabras o más, se unen iniciando cada una con la letra mayúscula y las demás en minúscula.

No incluir ningún espacio entre el nombre del método y el paréntesis inicial del listado de parámetros.

Sintaxis: NombreClase

Ejemplo:
Solicitud
Banco

Atributos

Si el nombre del atributo consta de una sola palabra, se escribe en letra minúscula; por otro lado, si el nombre del atributo contiene más de una palabra cada palabra es unida a la anterior y comienza con la letra mayúscula, a excepción de la primera palabra que comienza con minúscula.

Sintaxis: nombreAtributo

Ejemplo:
dirección
fechaNacimiento

Funciones

El nombre de una operación se escribe en letra minúscula si consta de una sola palabra. Si el nombre consta de más de una palabra, se une iniciando cada una con la letra mayúscula exceptuando la primera.

Sintaxis: nombreFuncion()

Ejemplo:
imprimir()
mostrarFecha(argumentos)

Objetos

Se escriben con letras minúsculas en caso de constar de una sola palabra. Si consta de varias palabras se unen iniciando cada una con la letra mayúscula exceptuando la primera.

Sintaxis: nombreObjeto

Ejemplo:
empleado
solicitudPrestamo

Variables

Los nombres que se usen deben ser significativos. Se escriben con letras minúsculas en caso de constar de una sola palabra. Si el nombre de la variable consta de más de una palabra se unen iniciando cada una con letra mayúscula exceptuando la primera. Toda variable es precedida por un argumento que identifica su alcance.

Sintaxis:

nombreVariable

Ejemplo:

fechaInicial
sesion

Constantes

Se escribe su nombre con letras mayúsculas. En caso de constar de varias palabras, se unen mediante un guión bajo.

Sintaxis:

NOMBRE_CONSTANTE

Ejemplo:

INTERES
NOMBRE_USUARIO

JavaScript

A continuación se listan los estándares a tomar en cuenta con la utilización de JS.

Tipos de datos: Los tipos de datos son los siguientes:

- **Primitivos:** En los tipos primitivos (string, number, boolean, null, undefined) se trabaja directamente con su valor.

Ejemplo:

```
const foo = 1;  
let bar = foo;  
bar = 9;
```

- **Complejos:** Al utilizar tipos complejos (object, array, function) se trabaja por referencia.

Ejemplo:

```
const foo = [1, 2];  
const bar = foo;  
bar[0] = 9;
```

Referencias:

- Usar **const** para las referencias en lugar de **var**

Ejemplo:

```
const a = 1;
const b = 2;
```

- Si se reasignan referencias se utilizará **let** en lugar de **var**

Ejemplo:

```
let count = 1;
if (true) {
  count += 1;
}
```

Objetos:

- Los objetos se inicializan de forma literal, sin la utilización de **new**

Ejemplo:

```
const item = {};
```

- Usar el método abreviado para los objetos

Ejemplo:

```
const atom = {
  value: 1,
  addValue(value) {
    return atom.value + value;
  },
};
```

En lugar de:

```
const atom = {
  value: 1,
  addValue: function (value) {
    return atom.value + value;
  },
};
```

Arreglos:

- Usar la sintaxis literal para la creación de arreglos

Ejemplo:

```
const items = [];
```

- Usar **Array#push** en lugar de la asignación directa para agregar elementos

Ejemplo:

```
const someStack = [];
someStack.push('abracadabra');
const items = [];
```

- Usar “...” para copiar arreglos

Ejemplo:

```
const itemsCopy = [...items];
```

- Usar salto de línea después de abrir y antes de cerrar las llaves cuando el arreglo tiene más de una línea

Ejemplo:

```
const arr = [[0, 1], [2, 3], [4, 5]];
const objectInArray = [
  {
    id: 1,
  },
  {
    id: 2,
  },
];
```

Cadenas de caracteres:

- Usar comillas simples para cadenas de caracteres

Ejemplo:

```
const name = 'Capt. Janeway';
```

- Si la cadena de caracteres es mayor a 100 no usar concatenación (+) para separar las múltiples líneas.
- Al construir cadenas con la utilización de objetos no usar concatenación (+) para su construcción

Ejemplo:

```
function sayHi(name) {
  return `How are you, ${name}?`;
}
```

- No usar **eval ()** en cadenas de caracteres para evitar vulnerabilidades.

Funciones:

- Usar expresiones de funciones en lugar de declaraciones de funciones

Ejemplo:

```
const foo = function bar() {
```

- La invocación de funciones debe ir entre paréntesis

Ejemplo:

```
(function () {
  console.log('Welcome to the Internet. Please follow me. ');
})();
```

- No re asignar parámetros dentro de la función

Ejemplo:

Incorrecto

```
function f1(a) {  
  a = 1;  
  // ...  
}
```

Correcto

```
function f3(a) {  
  const b = a || 1;  
  // ...  
}
```

Clases y Constructores

- Siempre usar class en lugar de prototype

Ejemplo:

```
class Queue {  
  constructor(contents = []) {  
    this.queue = [...contents];  
  }  
  pop() {  
    const value = this.queue[0];  
    this.queue.splice(0, 1);  
    return value;  
  }  
}
```

- Usar extends para la herencia

Ejemplo:

```
class PeekableQueue extends Queue {  
  peek() {  
    return this.queue[0];  
  }  
}
```

React

A continuación, se mencionan los estándares a utilizar con la librería React de JS que se utilizara para el frontend.

Incluir sólo un componente React por archivo.

Utilizar la sintaxis de JSX.

No usar `React.createElement` a menos que se está inicializando la aplicación desde un archivo que no es JSX.

No utilizar mixins ya que introducen dependencias implícitas, provocan choques de nombres y causan complejidad.

Nombre: Se siguen las siguientes pautas:

Extensiones: Utilizar la .jsx extensión para los componentes de React.

Nombre de archivo: Utilizar PascalCase para nombres de archivo.

Ejemplo:

ReservationCard.jsx.

Nombres de referencia: Utilizar PascalCase para los componentes React y camelCase para sus instancias.

Ejemplo:

```
const reservationItem = <ReservationCard />;
```

Nombres de componentes: Utilizar el nombre de archivo como nombre de componente.

Ejemplo:

ReservationCard.jsx debe tener un nombre de referencia de ReservationCard

Nombres de componentes de orden superior: Utilice un compuesto del nombre del componente de orden superior y el nombre del componente transferido como el displayName del componente generado.

Nombres de propiedades: No usar nombres prop de componentes de DOM para diferentes propósitos.

Ejemplo:

Utilizar `<MyComponent variant="fancy" />`
En lugar de `<MyComponent style="fancy" />`

Declaraciones. No usar displayName para nombrar componentes. En su lugar, declarar el componente por referencia.

Ejemplo:

```
export default class ReservationCard extends React.Component {  
}
```

Sangrado: Utilizar los siguientes estilos de alineación para la sintaxis de JSX

- Cuando es en una sola línea

Ejemplo:

```
<Foo bar="bar" />
```

- Más de una línea

Ejemplo:

```
<Foo  
  superLongParam="bar"
```

```
    anotherSuperLongParam="baz"  
  />
```

Comillas: Usar comillas dobles (" ") para los atributos de JSX, pero comillas simples (' ') para todos los otros JS.

Ejemplo:

```
<Foo bar="bar" />  
  
<Foo style={{ left: '20px' }} />
```

Espaciado: Usar únicamente un espacio entre la etiqueta de cierre y entre las llaves

Ejemplo:

```
< Foo />  
< Foo bar = { baz } />
```

Propiedades

- **Imágenes:** Siempre incluir la propiedad **alt** en las etiquetas de imagen ``, si la imagen es de presentación **alt** puede ir vacío o se utiliza **role="presentation"**

Ejemplo:

```
< img src = " hello.jpg " alt = " Yo diciendo hola " />  
  
< img src = " hello.jpg " alt = " " />  
  
< img src = " hello.jpg " role = " presentation " />
```

- **Paréntesis:** Las etiquetas JSX siempre ponerlas entre paréntesis cuando abarcan más de una línea

Ejemplo:

```
render () {  
  return (  
    < MyComponent className = " cuerpo largo " foo = " bar " >  
      < MyChild />  
    < / MyComponent >  
  );  
}
```

- **Etiquetas:** Siempre cerrar las etiquetas que no tienen hijos, si la etiqueta tiene varias propiedades de línea cerrarla en una línea nueva

Ejemplo:

```
< Foo className = "stuff" />  
< Foo  
  bar = "bar"  
  baz = "baz"  
>
```

Métodos

- No utilice prefijo de subrayado para los métodos internos de un componente React.

Ejemplo:

```
class extends React.Component {
  onClickSubmit () {
    // hacer cosas
  }
  // otras cosas
}
```

- Siempre devolver un valor en los render

Ejemplo:

```
render () {
  return (< div />);
}
```

- Vincular los controladores de eventos para el método render en el constructor.

Ejemplo:

```
class extends React.Component {
  constructor(props) {
    super(props);
    this.onClickDiv = this.onClickDiv.bind(this);
  }
  onClickDiv() {
    // do stuff
  }
  render() {
    return <div onClick={this.onClickDiv} />;
  }
}
```

5.2.2 Base de datos

La base de datos a utilizar para el desarrollo del sistema es MongoDB, que es una base de datos no relacional, que almacena documentos JSON (JavaScript Object Notation).

JSON se puede definir como un estándar para definir objetos en forma de texto plano (ECMA-404)

Los datos se representan internamente en un formato llamado BSON, binary json. BSON es la forma serializada de manejar documentos JSON y permite la representación de tipos de datos que no forman parte de JSON, como por ejemplo los Date.

Para trabajar con MongoDB se deben tener en cuenta las siguientes restricciones propias de la base de datos

Nombre de la BD

Al no ser Case sensitive el nombre de la base de datos debe tomar en cuenta lo siguiente:

- **Windows:** El nombre de la base de datos no puede contener los siguientes caracteres:
Λ. "\$*<>|?
- **Linux:** El nombre de la base de datos no debe contener los siguientes caracteres:
Λ. "\$
- La longitud del nombre no debe exceder los 64 caracteres
- El nombre de la base no puede ser nulo.

Nombres de Colecciones

Los nombres de las colecciones deben comenzar con “_” o con una letra y no pueden contener los siguientes:

- Llevar el signo \$.
- Ser una cadena vacía (por ejemplo "").
- Contener el carácter nulo.
- Comience con el **system.** prefijo (Reservado para uso interno.)

Nombres de Campos

Los nombres de campos no pueden contener puntos, null o comenzar con el signo \$.

5.2.3 Reportes

Las características de los reportes generados por el sistema son las siguientes:

Encabezado: Aquí va ubicado el logo del sistema, seguido del nombre del Sistema, la fecha de emisión del reporte en el formato dd/mm/aaaa, el nombre del reporte y el usuario que lo generó.

El tipo de letra para el encabezado: Arial 13 en negritas y mayúsculas

Tipo de letra para la fecha: Arial 11

Detalle del reporte: Aquí se muestra el detalle del reporte generado

Tipo de letra para el detalle: Arial 12 justificado

Pie de página: Aquí ira el número de páginas, con el siguiente formato: pág. ##/##

Tipo de letra: Arial 10

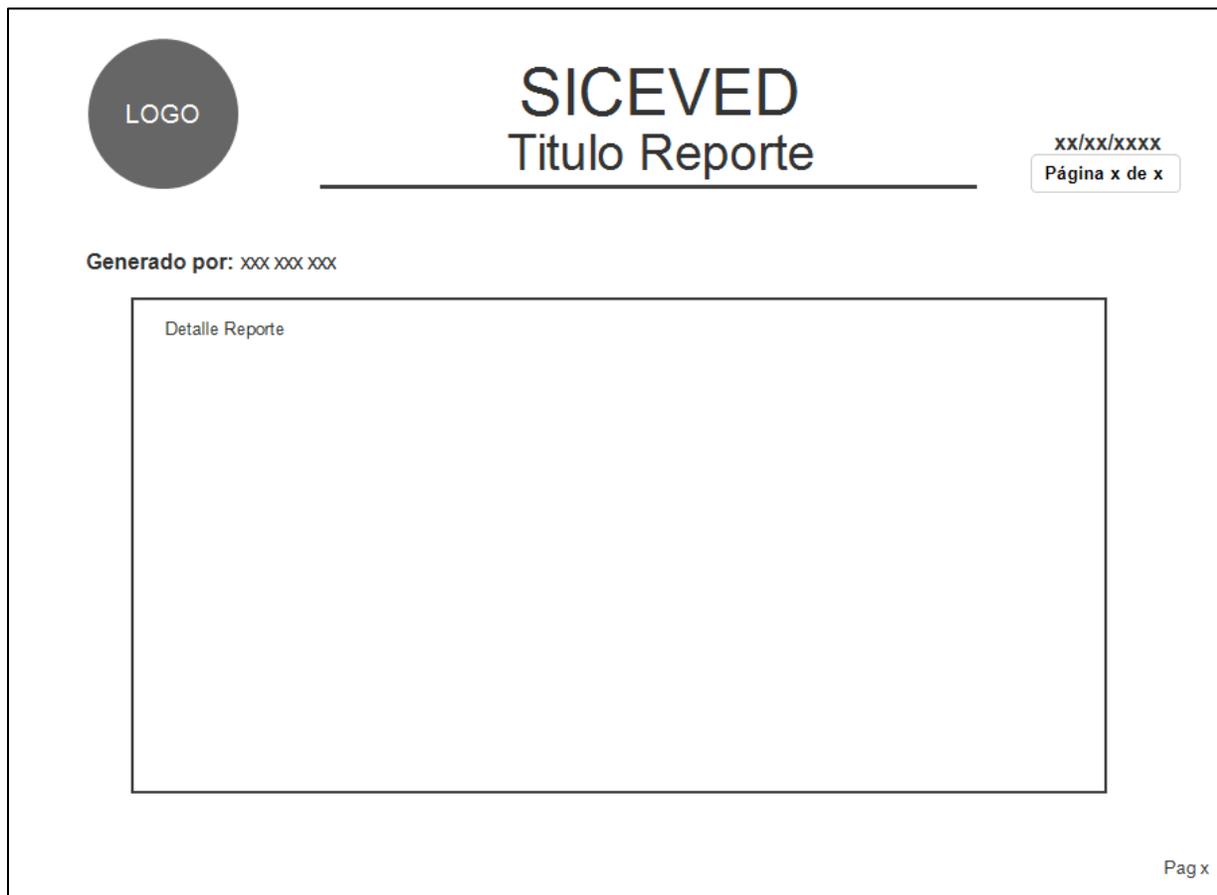


Ilustración 49. Modelo reporte.

5.2.4 Pantallas de Entradas

La calidad de las entradas de datos determina la calidad de las salidas del software. Los estándares de pantallas de captura de datos permiten crear una interfaz gráfica entre el usuario y el software. Es por esto que es de vital importancia que las pantallas de entrada cumplan con los siguientes objetivos:

- Efectividad
- Precisión
- Facilidad de uso
- Consistencia
- Simplicidad
- Atractivo

Lineamientos para la captura de datos de entrada:

- La pantalla de entrada de datos debe satisfacer el objetivo para el que ha sido creado
- Debe asegurarse el llenado preciso de la pantalla de entrada.
- Las pantallas de captura de datos deben facilitar el llenado de los datos.
- Efectuar validaciones para cada elemento de entrada de datos.

5.2.5 Pantallas de Salidas

La salida es la información que se entrega a los usuarios por tanto ésta debe ser útil y esencial para asegurar el uso y aceptación del software.

Lineamientos para la salida de datos:

- Presentar la información de forma ordenada.
- La pantalla de salida de datos debe satisfacer el objetivo para el que fue creado.
- Debe mantenerse la consistencia en las pantallas de salida.

5.3 Documentación durante el desarrollo

Se han definido dos tipos de documentación con la cual se trabajará:

Documentación interna: Dentro del código de programación

Documentación Externa: Referente a los manuales del Sistema

5.3.1 Documentación interna

La documentación interna en el código comienza con la elección de los nombres de identificadores (variables, constantes, objetos, etc.) y su declaración explícitamente, la asignación de nombres mnemotécnicos en el caso de las variables locales o globales, para los nombres de tablas, campos y objetos.

En cada módulo desarrollado se incluirá un encabezado con los siguientes datos:

- nombre del módulo
- tipo
- fecha de creación

Además de la documentación interna por medio de comentarios descriptivos que expliquen las funciones de los procesos.

Para la documentación interna se tomarán en cuenta los siguientes puntos:

- Usar `/** */` para comentarios de más de una línea

Ejemplo:

```
/**
 * make() returns a new element
 * based on the passed-in tag name
 */
```

- Usar `//` para comentarios de una sola línea

Ejemplo:

```
// set the default type to 'no type'  
const type = this.type || 'no type';
```

- Comenzar todos los comentarios con un espacio en blanco para facilitar su lectura

Ejemplo:

```
// este es un comentario
```

5.3.2 Documentación externa

La documentación externa del sistema está constituida por:

- **Manual de Usuario:** Este manual contiene toda la información para lograr un mejor aprendizaje y entendimiento de la funcionalidad del Sistema de Información Gerencial.
- **Manual Técnico:** Este manual contiene la información sobre los recursos utilizados en el proyecto, además de una descripción sobre las características físicas y técnicas de los elementos más importantes del Sistema, cabe mencionar que este manual está dirigido a personas con conocimientos especializados.
- **Manual de instalación y configuración:** Este manual contiene información para instalar y configurar los servidores necesarios para el correcto funcionamiento del sistema informático.

Se realizarán 2 manuales de usuario, uno para el portal web y otro para la aplicación móvil, 3 manuales técnicos, todos a nivel de desarrollador, uno para el portal web, otro para la aplicación móvil y otro para el backend del sistema. Y por último un manual de instalación y configuración, el cual será para poner en funcionamiento los servidores que dan soporte al sistema.

La documentación externa sigue los siguientes estándares:

Tipo papel: Bond tamaño carta

Márgenes: Sup. 2.25cm, Inf. 2.5 cm, Izq. 2.5 cm, Der. 2.5cm

Encabezado y pie de página: 1.27cm

Numeración de página: Centro inferior de la página

Interlineado: 1.5 líneas

Tipo de letra: Calibri

Títulos: Calibri 15, con color de resaltado.

Normal: Calibri 12 justificado.

Estructura básica de los manuales:

Índice: Relación de los capítulos y páginas correspondientes que forman parte del documento

Introducción: Se debe presentar una breve descripción de las áreas comprendidas en el manual.

Objetivo general: Se debe de describir el objetivo general del manual.

Objetivos específicos: Se deben describir brevemente los objetivos específicos que se cumplieron con el desarrollo del manual.

Contenido: El contenido del manual

Responsables: Personal responsable de la elaboración de dicho manual

CAPÍTULO VI: Plan de implementación

6.1 Introducción

A continuación, se describe el proceso para la configuración y puesta en marcha de SICEVED (Sistema De Información para el Censo Y Evaluación De Edificios en la Facultad de Ingeniería y Arquitectura de La Universidad de El Salvador Aplicando Tecnologías GIS), el cual permitirá instalar, configurar y capacitar a los profesionales de Ingeniería Civil e Ingeniería Estructural para el uso del sistema web y la aplicación móvil.

Este plan asegura la operatividad del sistema en el proceso de evaluación de edificaciones en la Facultad de Ingeniería y Arquitectura de la Universidad de El Salvador.

6.2 Objetivos

6.2.1 Objetivo general

Establecer el plan de implementación de SICEVED que permitirá la correcta instalación y puesta en marcha del mismo.

6.2.2 Objetivos específicos

- Establecer los recursos informáticos, humanos y económicos para la implementación del sistema.
- Establecer las actividades para la implementación del sistema.
- Instalar y configurar el software de acuerdo al manual de instalación y configuración del sistema en producción.
- Realizar pruebas del sistema para verificar su funcionalidad de acuerdo a los requerimientos.
- Garantizar la funcionalidad, integridad del sistema y consistencia de la información.

6.3 Marco Referencial del Plan de Implementación

6.3.1 Nombre del proyecto

SICEVED: Sistema De Información para el Censo Y Evaluación De Edificios en la Facultad de Ingeniería y Arquitectura de La Universidad de El Salvador Aplicando Tecnologías GIS.

6.3.2 Ubicación

Universidad de El Salvador.

6.3.3 Descripción del Proyecto

El sistema de información desarrollado es una herramienta de apoyo para los ingenieros civiles y estructurales para realizar evaluación de edificios que hayan sido afectados por algún fenómeno natural o deterioro por falta de mantenimiento, permitiendo determinar el estado del mismo y permitiendo llevar un registro de los cambios sufridos a lo largo del tiempo y las inspecciones realizadas sobre el mismo. Además, el sistema permite crear formularios y configurar la lógica de calificación, esta característica permitirá generalizar las evaluaciones ya no se verá limitado el tipo de evaluación que se debe realizar, sino que el administrador del sistema podrá ingresar y configurar cualquier tipo de formulario, por ejemplo: eficiencia energética, accesibilidad, etc.

6.3.4 Recursos

Se estima que la implementación del sistema se realizará en un periodo no mayor a un mes, por lo tanto todos los costos iniciales se contemplan para un mes.

A continuación, se detallan los recursos que serán necesarios para la implementación del sistema.

6.3.4.1 Recursos informáticos:

Los recursos informáticos definidos a continuación se dividen en dos: costos iniciales los cuales indican que solo se realizaran una vez al inicio del proyecto y costos fijos que son los recurrentes para mantener el sistema en marcha.

- **Costo inicial:**

Equipo	Cantidad	Costo Unitario	Costo mes de implementación
Computador de escritorio	2	\$ 600	\$1,200
Servidor en la nube	2	\$ 80.00	\$ 160.00
Volumen para servidor (1TB)	3	\$ 100.00	\$ 300.00
Cables UTP(5 metros)	5	\$1.50	\$7.50
Internet	1	\$37.99	\$37.99
Total			\$ 1705.49

Tabla 51: Costos iniciales de recursos informáticos para la implementación del sistema.

- **Costos fijos:**

Equipo	Cantidad	Costo/Mes	Depreciación mensual	Costo / mes
Computador de escritorio	2	-	\$12.48	\$24.96
Internet	1	\$37.99	-	\$37.99
Servidor en la nube	2	\$ 80.00	-	\$ 160.00
Volumen para servidor (1TB)	3	\$ 100.00	-	\$ 300.00
Total				\$ 522.95

Tabla 52 Costos fijos de recursos informáticos para la implementación del sistema.

Características

Servidor en la nube:

- **Memoria RAM:** 16 GB
- **CPU's:** 6
- **Disco SSD:** 320 GB
- **Transferencia de datos:** 6 TB

Servidor 1: Aplicación Web.

Servidor 2: Almacenamiento de archivos (Imágenes y base de datos.)

**Costo basado en Droplets estándar de Digital Ocean (\$80/mo, \$0.119/hr)

Internet

- **Velocidad de descarga:** 10Mbps
- **Velocidad de subida:** 2 Mbps

6.3.4.2 Recurso humano²¹:

A continuación, se describen los recursos humanos necesarios para la implementación del sistema dividiéndolos en dos: costos iniciales los cuales indican que solo se realizan al inicio del proyecto y costos fijos que son los recurrentes para mantener actualizados los datos del sistema.

- **Costo inicial:**

Equipo	Cantidad	Costo / día	Días	Costo mes de implementación
Ingeniero en Informática (Implementador)	1	\$ 80.00	20	\$ 1,600.00
Capacitador	2	\$ 80.00	5	\$ 800.00
Total				\$ 2,400.00

²¹ Costos obtenidos en <https://tusalarario.org/elsalvador/salario/>

Tabla 53: Costos iniciales de recurso humano para la implementación del sistema.

- **Costos fijos:**

Es necesario un Ingeniero en Informática que brinde soporte a problemas que el usuario puede tener en el uso de la aplicación o uso de los instrumentos de inspección, además será el encargado de la creación de formularios, agregar o modificar características del sistema.

Equipo	Cantidad	Costo / día	Días	Costo / mes
Ingeniero en Informática (Administrador del Sistema)	1	\$ 80.00	20	\$ 1,600.00
Total				\$ 1,600.00

Tabla 54 Costos fijos de recurso humano para la implementación del sistema.

6.3.4.3 Costo total de implementación

- **Costo inicial**

Costo	Costo total mes de implementación
Recursos informáticos	\$ 1,717.99
Recurso humano	\$ 2,400.00
Total	\$ 4,117.99

Tabla 55: Costos inicial total de la implementación del sistema.

- **Costo fijo**

Costo	Costo total / mes
Recursos informáticos	\$ 497.99
Recurso humano	\$ 1,600.00
Total	\$ 2,097.99

Tabla 56: Costos fijos totales de la implementación del sistema.

6.4 Actividades

ID	Actividad	Descripción	Tiempo (días)
1	Elaborar requerimientos de hardware	Se elabora un documento en el cual se incluyen los recursos a utilizar con las características descritas en la sección de recursos.	1
2	Enviar solicitud a proveedores	Enviar solicitud a proveedores de recursos informáticos para que provean de ofertas.	1
3	Analizar y evaluar las ofertas provistas por los proveedores	Revisar las propuestas presentadas por los proveedores para seleccionar la que se adapte a las necesidades y presupuesto asignado al proyecto	1
4	Realizar solicitud de comprar del equipo informático.	Elaborar la solicitud y orden de compra con el proveedor seleccionado.	1
5	Configuración del sistema informático.	Configurar el software requerido, siguiendo el proceso definido en el Manual de Instalación y Configuración del Sistema en Producción.	2
6	Compilar apk de producción.	Compilar apk con la ip del servidor de producción.	1
7	Instalación de aplicación móvil	Instalar en dispositivo móvil el apk.	1
8	Realizar pruebas	Verificar el correcto funcionamiento de la API, Sistema Web y aplicación móvil.	10
9	Capacitar al personal para el uso del sistema.	Capacitar al o los administradores (Encargado de instalar, configurar y administrar el sistema) del sistema web y los evaluadores en la aplicación móvil. ²²	5
10	Registrar usuarios.	Registrar usuarios administrador y evaluadores dentro del sistema	1
11	Puesta en producción	Cambiar el origen de datos de pruebas a producción.	1

²² Ver Apéndices A y B: Planes de capacitación para administradores y evaluadores.

Tabla 57: Actividades a desarrollar para la implementación del sistema.

6.5 Administración de riesgos

A continuación, se describen los posibles riesgos que pueden afectar el funcionamiento correcto de la aplicación del sistema y su implementación.

N°	Riesgo	Descripción	Tipo	Probabilidad de ocurrencia	Impacto	Plan de Contingencia
1	Daños en el servidor de base de datos	Daños en el servidor de base de datos, pérdida de información.	Tecnológicos	Baja	Alta	Respaldo y restauración de copia de seguridad.
2	Despido, renuncia de personal.	El personal es removido o renuncia a su cargo de administrador del sistema.	Ambiental	Bajo	Medio	Contratación y capacitación del personal.
3	Retraso en pago de servidor en la nube	El pago a realizar por el servidor no fue en el periodo correspondiente	Financiero	Media	Alta	Gestión de pago de servidor.

Tabla 58: Lista de riesgos contemplados en la implementación del sistema.

6.6 Planes de contingencia

6.6.1 Objetivo

Corregir y solventar los riesgos que puedan afectar la configuración, instalación y funcionamiento en producción del sistema.

6.6.2 Descripción

El plan de contingencia describe los procesos a seguir para corregir y solucionar los riesgos que pueden afectar el correcto funcionamiento del sistema

6.6.1 Plan de Respaldo y restauración de copia de seguridad.

Objetivo	Respaldo y restaurar las copias de seguridad de la base de datos.
Tiempo estimado	1 a 2 días
Responsable	Administrador del sistema.
Procesos	Respaldo de base de datos
	1. Consultar el manual de instalación y configuración del sistema, apartado Respaldo de Base de Datos.
	Restauración de base de datos
	1. Consultar el manual de instalación y configuración del sistema, apartado Restauración de Base de Datos

Tabla 59: Plan de contingencia de respaldo y restauración de copia de seguridad.

6.6.2 Plan de contratación de nuevo personal

Objetivo	Inducción y capacitación del nuevo empleado en la utilización y mantenimiento del sistema.
Tiempo estimado	15 días
Responsable	Líder del Proyecto
Proceso	<ol style="list-style-type: none"> 1. Informar a recursos humanos de la vacante disponible. 2. Entregar el perfil que describe las habilidades y conocimientos de la persona requerida. 3. Seleccionar y contratar al nuevo personal. 4. Inducción del personal. <ol style="list-style-type: none"> 1. Entrega del manual de instalación y configuración del sistema. 2. Entrega de manual de programador del sistema. 3. Entrega de manual de usuario del sistema. 4. Entrega de credenciales del servidor. 5. Entrega de credenciales del sistema.

Tabla 60: Descripción del plan de contingencia de contratación de nuevo personal.

6.6.2 Plan de Gestión de pago de servidor.

Objetivo	Permite conocer el proceso para gestionar el pago del servidor y verificar que las configuraciones realizadas no se hayan perdido.
Tiempo estimado	2 a 5 días
Responsable	Administrador del sistema

Procesos	<ol style="list-style-type: none"> 1. Solicitar pago de servidor a departamento de finanzas. 2. Luego de realizado el pago, verificar que el sistema no se hayan perdido configuraciones y realizar pruebas: <ol style="list-style-type: none"> 1. Si se perdieron configuraciones del sistema: <ol style="list-style-type: none"> 1. Consultar manual de instalación y configuración del sistema. 2. No se perdieron configuraciones <ol style="list-style-type: none"> 1. Notificar que el sistema está nuevamente activo.
-----------------	---

Tabla 61: Descripción del plan de contingencia de gestión de pago de servidor.

Conclusiones

La correcta descripción de los requerimientos para un Sistema Informático es de suma importancia ya que esto marca una pauta clara para los desarrolladores con respecto a las funcionalidades mínimas que el Sistema debe tener, es por tal motivo que debe contarse con una buena comunicación con el cliente y la aprobación de los requerimientos obtenidos de manera que se minimicen los cambios futuros al sistema en la etapa de Construcción.

La definición e implementación de estándares son de gran apoyo para los desarrolladores ya que permiten la creación de código legible para todos los integrantes del equipo de desarrollo, además brindan la base para que personas ajenas al equipo de desarrollo puedan comprender y modificar el sistema en caso sea necesario.

La construcción del “Sistema de información para el censo y evaluación de edificios en la facultad de ingeniería y arquitectura de la universidad de el salvador aplicando tecnologías gis” es de gran importancia para la Universidad El salvador ya que permitirá conocer el estado actual en que se encuentran las edificaciones de manera que se tomen las medidas pertinentes en cada caso para reducir el riesgo de la población universitaria.

Recomendaciones

- Utilizar herramientas de análisis para datos no estructurados, sobre la información recolectada con los diferentes módulos de evaluación, y de esta manera obtener valores estadísticos de cualquier índole, aprovechando así la captación de datos genérica del sistema.
- En la creación de nuevos módulos de evaluación evitar definir campos de ingreso de datos que ya han sido proporcionados durante la creación de una nueva edificación. Evitando así la duplicidad de datos.
- A la hora de crear nuevos módulos de evaluación, ocultar todos los campos que sean dependientes de campos no utilizados.
- Si se crea un módulo de evaluación que incluya múltiples campos de captura de foto, notificar con anticipación a los evaluadores que deben poseer una amplia cantidad de memoria para el almacenamiento de dichos archivos, evitando así que una, o muchas inspecciones, se vean interrumpidas por falta de espacio en el dispositivo móvil.
- Documentar la lógica de cálculo de bandera de cada módulo ingresado al sistema en un formato de diagrama de sintaxis, para posteriores ediciones.

Glosario

- **Sismo:** son movimientos convulsivos en el interior de la tierra y que generan una liberación repentina de energía que se propaga en forma de ondas provocando el movimiento del terreno.
- **Incendio:** es una ocurrencia de fuego no controlada que puede afectar o abrasar algo que no está destinado a quemarse.
- **Capa:** Conjunto de datos espaciales asociados a un contenido temático común.
- **Georreferenciar:** Asignar coordenadas geográficas a un objeto o estructura.
- **Mapa:** Representación gráfica del territorio, de acuerdo a determinadas convenciones o normas, en un modelo reducido y a escala, que establece una correspondencia matemática, continua y biunívoca, entre los distintos puntos de la superficie terrestre y los de un plano.
- **Prototipo:** es un modelo (representación, demostración o simulación) fácilmente ampliable y modificable de un sistema planificado, probablemente incluyendo su interfaz y su funcionalidad de entradas y salidas.
- **Estación de trabajo:** es un ordenador que facilita a los usuarios el acceso a los servidores y periféricos de la red. A diferencia de un ordenador aislado, tiene una tarjeta de red y está físicamente conectada por medio de cables u otros medios no guiados con los servidores.
- **Inspección:** es una técnica de evaluación formal en la cual una persona o grupo de personas examinan en detalle las características de un elemento.
- **Sistema de Alerta Temprana (SAT):** es una herramienta técnica que soporta la reducción de riesgos y la preparación ante desastres, con el objetivo de proteger a las personas y sus medios de vida expuestos a peligros.
- **Mapa de riesgo:** es una herramienta, basada en los distintos sistemas de información, que pretende identificar las actividades o procesos sujetos a riesgo, cuantificar la probabilidad de estos eventos y medir el daño potencial asociado a su ocurrencia.
- **Estructura:** Está formada por el miembro o conjunto de miembros estático-resistentes cuya función es soportar las cargas que sobre ellos actúan (vivas y muertas).
- **ASIA:** Es un gremio profesional sin fines de lucro, apolítico, con personería Jurídica y en su seno aglutina Ingenieros de todas las especialidades, Arquitectos y otras profesiones vinculadas con esas disciplinas.
- **CASALCO:** La Cámara Salvadoreña de la Industria de la Construcción (CASALCO) es una gremial que fue creada con el objetivo de integrar, unificar y coordinar esfuerzos que permitan la superación gremial y defensa de los intereses de la Industria de la construcción. Fue fundada el 3 de noviembre de 1964, desde esa fecha se creó como

una institución de utilidad pública, de conformidad con los lineamientos establecidos en el capítulo treinta del Código Civil.

- **Catastro:** es un registro administrativo dependiente del Estado en el que se describen los bienes inmuebles rústicos, urbanos y de características especiales. Entre las características del catastro podemos encontrar que es un registro estadístico para determinar la extensión geográfica y riqueza de alguna demarcación y que en materia hacendaria es un apoyo para determinar el cobro de las imposiciones del estado, según lo manifestado en los registros.
- **ACID:** son las características de los parámetros que permiten clasificar las transacciones de los sistemas de gestión de bases de datos. Cuando se dice que es *ACID compliant* indic que éste permite realizar transacciones. En concreto **ACID** es un acrónimo de **A**tomicity, **C**onsistency, **I**solation and **D**urability: Atomicidad, Consistencia, Aislamiento y Durabilidad.

Bibliografía

Libros

- Caio Ribeiro Pereira. (2016). Building APIs with Node.js. São Paulo, Brazil: Apress.
- Doug Garrett, Kyle Banker, Peter Bakkum, Shaun Verch, Tim Hawkins. (2016). MongoDB in Action. 20 Baldwin Road PO Box 761 Shelter Island, NY 11964: Manning Publications Co.
- Marc Garreau Will Faurot Foreword by Mark Erikson. (2016). Redux In Action. 20 Baldwin Road Po Box 761 Shelter Island, Ny 11964: Manning Publications Co.

Sitios web.

- Lupo Montero. (2017). Introducción a programación funcional en JavaScript—Parte 1. Sitio web: <https://medium.com/laboratoria-developers/introducci%C3%B3n-a-la-programaci%C3%B3n-funcional-en-javascript-parte-1-e0b1d0b2142e>
- Bruce Sun. (2011). Arquitectura multinivel para la construcción de servicios web RESTful. Sitio web: <https://www.ibm.com/developerworks/ssa/library/wa-aj-multitier/index.html>
- Fundacion Wikimedia. (2016). Astah. Sitio web: https://en.wikipedia.org/wiki/Astah*
- ERDPlus. (2015). ERD. Sitio web: <https://erdplus.com/>
- Facebook Inc. (2017). React. Sitio web: <https://reactjs.org/>
- Facebook Inc. (2017). React Native. Sitio web: <https://facebook.github.io/react-native/>
- MongoDB, Inc. (2017). NoSQL Databases Explained. Sitio web: <https://www.mongodb.com/nosql-explained>

Anexos

Anexo A: Formulario CASALCO²³

FORMULARIO ÚNICO DE INSPECCIÓN RÁPIDA
 Dirección de Protección Civil – Vice Ministerio de Vivienda y Desarrollo Urbano – OPAMSS – CASALCO

Formulario No. _____

Identificación de la Edificación

Nombre del Edificio: _____
 Dirección: _____
 Ciudad: _____ Municipio: _____ Departamento: _____
 Persona de Contacto: _____ Tel: _____

Descripción de la Edificación

Presentar esquema de la edificación al final de la página 4, indicando la dirección Norte (Use brújula).

Ubicación de la edificación en la manzana:
 Esquina Entre dos Edificaciones Libre o Aislada

Época de construcción:
 Antes de 1966 1966-1974 1975-1986 Después de 1986

Número de niveles sobre el terreno: _____

Sótanos: Si No Número: _____ Desconocido

Uso:

Casa de habitación Apartamento Comercio Oficina
 Industria Estadio o similar Estacionamiento Hotel
 Bodega Salud Educación Histórico
 Recreativo Gubernamental Servicio Emergencia Comunicaciones
 Otro: _____

Sistema Estructural

Material	Sistema Estructural					
Concreto	Marcos		Muros Estructurales		Sistema Dual	
	Dir. N-S <input type="radio"/>	Dir. E-O <input type="radio"/>	Dir. N-S <input type="radio"/>	Dir. E-O <input type="radio"/>	Dir. N-S <input type="radio"/>	Dir. E-O <input type="radio"/>
Acero	Marcos Arriostrados		Marcos No Arriostrados		Marcos Industriales	
	Dir. N-S <input type="radio"/>	Dir. E-O <input type="radio"/>	Dir. N-S <input type="radio"/>	Dir. E-O <input type="radio"/>	Dir. N-S <input type="radio"/>	Dir. E-O <input type="radio"/>
Mampostería	Confinada		Con Refuerzo Interior		No Reforzada (Adobe o Bahareque)	
	Dir. N-S <input type="radio"/>	Dir. E-O <input type="radio"/>	Dir. N-S <input type="radio"/>	Dir. E-O <input type="radio"/>	Dir. N-S <input type="radio"/>	Dir. E-O <input type="radio"/>

Marcos de Concreto combinados con Paredes Mampostería Marcos de Acero combinados con Paredes Mampostería
 Marcos Industriales Otro Sistema: _____

Sistema de Entrepiso

Concreto: Losa densa Losa nervada en una dirección Losa nervada en dos direcciones
 Acero: Metal deck
 Otro: _____

²³ Fuente: Phd. Edgar Armando Peña Figueroa.

FORMULARIO ÚNICO DE INSPECCIÓN RÁPIDA

Dirección de Protección Civil – Vice Ministerio de Vivienda y Desarrollo Urbano – OPAMSS – CASALCO

Sistema de Cubierta (Techo)

Lámina Teja Losa Estructura especial Otro: _____

Regularidad de la Estructura

En Planta: Regular Medio regular Irregular
En Altura: Regular Medio regular Irregular

Daños y Reparaciones Previos

Daños en sismos anteriores Sí Año _____ No No se sabe

Reparaciones realizadas en el pasado por sismo Sí Año _____ No No se sabe

Instrucciones

Revisar la edificación para las condiciones señaladas abajo. Con un **Sí** a cualesquiera de las preguntas 1 a 6, marcar la edificación como **Insegura. (Color Rojo)**. Con un **Sí** a las preguntas 7 ú 8 marcar **Área Insegura** y recomendar colocar barreras alrededor de la zona de peligro. Si en esta evaluación existen dudas se debe marcar la edificación como **Cuidado. (Color Amarillo)**.

Criterios	Sí	No	Existen Dudas	
1. Colapso total o parcial	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	}
2. Inclinación notoria de la edificación o de algún entrepiso o giro notorio de algún entrepiso (torsión).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
3. Edificación separada de su cimentación o falla de esta	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
4. Daños importantes (severos o fuertes) en miembros estructurales (columnas, vigas, uniones viga-columna y paredes estructurales)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
5. Daño severo en paredes no estructurales, escaleras, cubo de ascensores, cielos falsos, etc.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
6. Grandes grietas en el suelo, movimiento masivo del suelo, hundimientos en la proximidades de la edificación o falla de muros de retención próximos a la edificación	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
7. Pretilas, balcones u otros elementos en peligro de caer	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	}
8. Presencia de otro tipo de riesgo (derrames tóxicos, peligro de contaminación, líneas de energía caídas, roturas de tuberías de agua potable y servidas, etc.)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	

Si = Inseguro

Si = Área insegura

Tipo de Inspección

Inspección de la edificación

Exterior solamente Completa interior y exterior Parcial

No se realizó la inspección porque:

No se permitió Desocupada Colapso Demolida Otro motivo

FORMULARIO ÚNICO DE INSPECCIÓN RÁPIDA

Dirección de Protección Civil – Vice Ministerio de Vivienda y Desarrollo Urbano – OPAMSS – CASALCO

Recomendaciones de Urgencia

- No entrar Entrar solo con permiso Evacuar la edificación
- Hay áreas inseguras. Colocar barreras en las siguientes áreas: _____

- Hay que apuntalar o remover elemento por el peligro que representa en las siguientes áreas: _____

- Cubrir con plástico las grietas en el suelo Reparar las tuberías rotas en la edificación

Riesgos Globales

Riesgos Estabilidad Global (Criterios 1, 2 y 3)

- Bajo Bajo después de tomar medidas Alto Muy alto

Riesgos Geotécnicos (Criterios 3 y 6)

- Bajo Bajo después de tomar medidas Alto Muy alto

Riesgo Estructural (Criterio 4)

- Bajo Bajo después de tomar medidas Alto Muy alto

Riesgo de Elementos No Estructurales (Criterio 5 y 7)

- Bajo Bajo después de tomar medidas Alto Muy alto

Clasificación de la Habitabilidad

- Habitable (Verde)
- Cuidado (Amarillo)
- Insegura (Rojo)

Recomendaciones

- Evacuar la edificación
- No se requiere de una revisión futura
- Es necesario una revisita por aspectos: Estructurales Geotécnicos Otro: _____
- En caso de roturas de tuberías del sistema de agua potable y alcantarillado reportar a ANDA
- En caso de caída de líneas eléctricas reportar a CAESS
- Se requiere de un Estudio de Ingeniería a ser presentado en: OPAMSS VMVDU
- Posible demolición (La CER deberá comunicar al propietario del inmueble)

FORMULARIO ÚNICO DE INSPECCIÓN RÁPIDA

Dirección de Protección Civil – Vice Ministerio de Vivienda y Desarrollo Urbano – OPAMSS – CASALCO

Comentarios

Explicar los motivos principales de la clasificación: _____

Inspectores (indicar profesión y jefe de brigada -JB-)

1. _____
2. _____
3. _____

CER No. _____

CER No. _____

CER No. _____

Fecha de inspección: _____ Hora de inspección: _____ **0** a.m **0** p.m.

Persona que recibe el formulario en **CER**: _____ Fecha: _____

Anotar en que archivo se guardo la información de la evaluación: _____

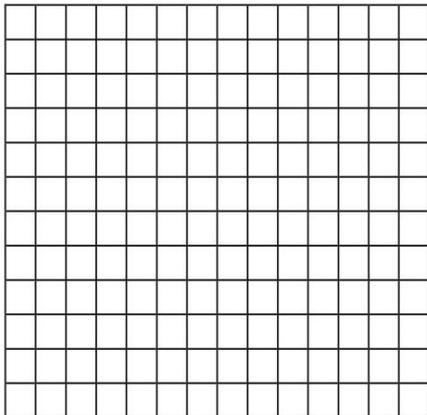
Nota: Anexar un mínimo de 3 fotos. Anotar en que archivo se guardaron las fotos: _____

Para uso oficial Código: _____ Sello

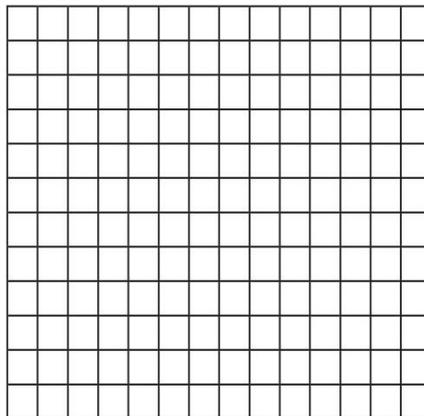
FORMULARIO ÚNICO DE INSPECCIÓN RÁPIDA

Dirección de Protección Civil – Vice Ministerio de Vivienda y Desarrollo Urbano – OPAMSS – CASALCO

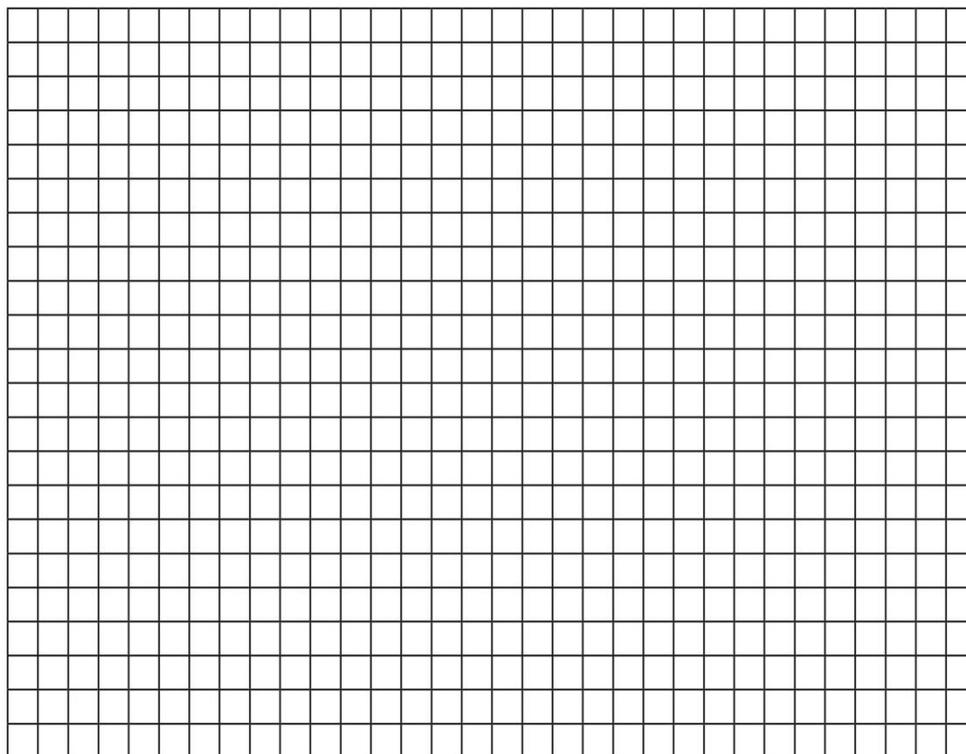
Croquis de localización



Croquis de Elevación



Croquis de Planta



Anexo B: Formulario ASIA²⁴

Guía Técnica para la Inspección de Edificaciones despues de un Sismo										
Formulario Número: _____										
Inspección de la Edificación:					Clasificación de Habitabilidad:					
Exterior e interior		No se pudo entrar			Verde		Amarillo		Naranja	Rojo
<input type="checkbox"/>		<input type="checkbox"/>			<input type="checkbox"/>		<input type="checkbox"/>		<input type="checkbox"/>	<input type="checkbox"/>
1.0- Identificación de la Edificación:										
Nombre del Edificio: _____										
Dirección: _____										
ciudad: _____			Municipio: _____			Departamento: _____				
Persona de Contacto: _____										
Coordenadas: _____										
1.1- Uso Predominante:										
1 Residencial		<input type="checkbox"/>		7 Industrial		<input type="checkbox"/>				
2 Comercial		<input type="checkbox"/>		8 Gubernamental		<input type="checkbox"/>				
3 Educacional		<input type="checkbox"/>		9 Estacionamiento		<input type="checkbox"/>				
4 Salud		<input type="checkbox"/>		10 Bodega		<input type="checkbox"/>				
5 Hotel		<input type="checkbox"/>		11 Entretenimiento		<input type="checkbox"/>				
6 Oficinas		<input type="checkbox"/>		12 Otros: _____		<input type="checkbox"/>				
1.2- Número de Niveles: _____										
Sótanos:		Si <input type="checkbox"/>		Número: _____		No <input type="checkbox"/>				
1.3- Dimensiones aproximadas de la Edificación: Frente: _____ (mt) Fondo _____ (mt)										
2.0- Descripción de la Estructura										
Sistema Estructural										
Concreto Reforzado:										
Marcos		Paredes Estructurales		Sistema Dual		Prefabricados				
2.01 <input type="checkbox"/>		2.02 <input type="checkbox"/>		2.03 <input type="checkbox"/>		2.04 <input type="checkbox"/>				
Mampostería:										
Confinada		Reforzada		No reforzada		Mixto				
Bloque concreto		2.05 <input type="checkbox"/>		2.06 <input type="checkbox"/>		2.07 <input type="checkbox"/>		2.08 <input type="checkbox"/>		
Confinada		Reforzada		No reforzada		Mixto				
Ladrillo de barro		2.09 <input type="checkbox"/>		2.10 <input type="checkbox"/>		2.11 <input type="checkbox"/>		2.12 <input type="checkbox"/>		
Acero:										
Marcos Arriostrados		Marcos No Arriostrados		Marcos Industriales						
2.12 <input type="checkbox"/>		2.13 <input type="checkbox"/>		2.14 <input type="checkbox"/>						
Adobe:										
Paredes		Bahareque		Paredes						
2.15 <input type="checkbox"/>				2.16 <input type="checkbox"/>						
Madera:										
Marcos		2.17 <input type="checkbox"/>								
Marco de concreto										
combinado con paredes de mamposteria		2.18 <input type="checkbox"/>								
Tapiales:										
Adobe		Mamposteria de barro reforzado		Bloque reforzado						
2.19 <input type="checkbox"/>		2.20 <input type="checkbox"/>		2.21 <input type="checkbox"/>						
Otro Sistema: _____										

²⁴ Fuente: Phd. Edgar Armando Peña Figueroa.

Año de Construcción:
 Antes de 1966 Entre 1966 -1974 Entre 1975 -1986 Entre 1986-2001
 Después de 2001

Sistema de Entrepiso:
Concreto: Losa densa Losa nervada 1 dirección Losa nervada 2 direcciones
Acero: Metal Deck
Otro: _____

3.0 Evaluacion del Estado de la Edificación

3.1 Estado General de la Edificación

Revisar la edificación en forma global para las condiciones señaladas a continuación y hacer las aclaraciones necesarias en la sección de comentarios:

1. Existe Colapso:	2. Desviación o inclinación de la edificación o de algún entrepiso:	3. Falla o Asentamiento de la cimentación:
No <input type="checkbox"/>	Si <input type="checkbox"/>	Si <input type="checkbox"/>
Parcial <input type="checkbox"/>	No <input type="checkbox"/>	No <input type="checkbox"/>
Total <input type="checkbox"/>	No se pudo determinar <input type="checkbox"/>	No <input type="checkbox"/>

3.2 Daños en elementos Arquitectónicos

4. Muros de fachadas	5. Muros divisorios o particiones	6. Cielos rasos y luminarias
Ninguno <input type="checkbox"/>	Ninguno <input type="checkbox"/>	Ninguno <input type="checkbox"/>
Leve <input type="checkbox"/>	Leve <input type="checkbox"/>	Leve <input type="checkbox"/>
Moderado <input type="checkbox"/>	Moderado <input type="checkbox"/>	Moderado <input type="checkbox"/>
Fuerte <input type="checkbox"/>	Fuerte <input type="checkbox"/>	Fuerte <input type="checkbox"/>
Severo <input type="checkbox"/>	Severo <input type="checkbox"/>	Severo <input type="checkbox"/>

7. Cubierta	8. Escaleras
Ninguno <input type="checkbox"/>	Ninguno <input type="checkbox"/>
Leve <input type="checkbox"/>	Leve <input type="checkbox"/>
Moderado <input type="checkbox"/>	Moderado <input type="checkbox"/>
Fuerte <input type="checkbox"/>	Fuerte <input type="checkbox"/>
Severo <input type="checkbox"/>	Severo <input type="checkbox"/>

9. Instalaciones:	Acueducto:	Alcantarillado:	Energía	Gas:	10. Tanques Elevados
Ninguno <input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Leve <input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Moderado <input type="checkbox"/>	<input type="checkbox"/>				
Fuerte <input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Severo <input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Problemas Geotécnicos

11. Falla en Talud o movimientos en masa	12. Asentamiento o Licuación
No <input type="checkbox"/>	No <input type="checkbox"/>
Puntual <input type="checkbox"/>	Puntual <input type="checkbox"/>
General <input type="checkbox"/>	General <input type="checkbox"/>

Daños en Elementos Estructurales en el piso de mayor afectación

Indique el nivel de entrepiso con el mayor daño: _____

Indique el porcentaje de los elementos afectados según su grado de daño:

	1 Ninguno	2. Leve	3.Moderado	4. Fuerte	5. Severo
13. Muros Mampostería	<input type="text"/>				
14. columnas o muros cortantes	<input type="text"/>				
15. Vigas	<input type="text"/>				
16. Nudos o puntos de conexión	<input type="text"/>				
17. Entrepisos	<input type="text"/>				

Porcentaje de Daños Global de la Edificación

Estimar el porcentaje de área afectada con relación al área total construída de la Edificación:

Rango:	%	Clasificación Global del daño
0%		Ninguno
0 - 10%		Leve
10 - 30%		Moderado
30 - 60%		Fuerte
60 - 100%		Severo
100%		Colapso Total

Clasificación Global del daño y habitabilidad de la edificación

1 Ninguno	Habitable (Verde)
2 Leve	Habitable (Verde)
3 Moderado	Uso Restringido (Amarillo)
4 Fuerte	No habitable (Anaranjado)
5 Severo	Peligro de colapso (Rojo)

Indique la clasificación del daño según la presente evaluación

Existe una clasificación Previa?

Si No

Cuál? _____

Recomendaciones y medidas de Seguridad

Se necesita visita especializada por aspectos:

Estructurales: _____ Geotécnicos: _____ Servicios Públicos: _____

Medidas de Seguridad:

Restringir paso de peatones	<input type="checkbox"/>
Evacuar parcialmente la Edificación	<input type="checkbox"/>
Evacuar totalmente la Edificación	<input type="checkbox"/>
Restringir tráfico Vehicular	<input type="checkbox"/>
Apuntalar	<input type="checkbox"/>
Demoler Elementos en peligro de caer	<input type="checkbox"/>
Evacuar Edificaciones vecinas	<input type="checkbox"/>
Desconectar 1. Energía 2. Gas 3. Agua	<input type="checkbox"/>

Especifique lugares de la edificación que requieran la aplicación de las medidas de seguridad:

Recomendaciones

Se recomienda intervención de:

Protección Civil	<input type="checkbox"/>
Policía-Ejército	<input type="checkbox"/>
Policia de Tránsito	<input type="checkbox"/>
Bomberos-Entidades de rescate	<input type="checkbox"/>

Comentarios:

Ampiar la evaluación con observaciones que ayuden a darle claridad al formulario.
Indicar los elementos donde los daños fueron más importantes. Amplie recomendaciones.

Condiciones Pre-Existentes

Calidad de la Construcción:	Posición de la Edificación en la manzana	Configuración en planta:	Configuración en Altura:
Buena <input type="checkbox"/>	Esquina <input type="checkbox"/>	Buena <input type="checkbox"/>	Buena <input type="checkbox"/>
Regular <input type="checkbox"/>	Intermedia <input type="checkbox"/>	Regular <input type="checkbox"/>	Regular <input type="checkbox"/>
Mala <input type="checkbox"/>	Libre por un costado <input type="checkbox"/>	Mala <input type="checkbox"/>	Mala <input type="checkbox"/>
	Libre por dos costados <input type="checkbox"/>		

Configuración Estructural:	Hay indicios de daños por Sismos anteriores?	Hubo reparación?
Buena <input type="checkbox"/>	Si <input type="checkbox"/>	Si <input type="checkbox"/>
Regular <input type="checkbox"/>	No <input type="checkbox"/>	No <input type="checkbox"/>
Mala <input type="checkbox"/>		

Efectos en los ocupantes

Hubo muertos o heridos:	Número de Fallecidos: <input type="text"/>	Número de Heridos: <input type="text"/>
Si <input type="checkbox"/>		
No <input type="checkbox"/>		
No se sabe <input type="checkbox"/>		

Anexo C. Visus

Las amenazas naturales pueden tener efectos negativos en las vidas de los estudiantes, como también pueden llegar a paralizar los sistemas de educación. La UNESCO, en el marco de la Alianza Global para la Reducción del Riesgo de Desastres y la Resiliencia en el Sector Educativo, está comprometida en proteger la seguridad de las comunidades educativas en el caso de un desastre.

Evaluar las instalaciones educativas para que sean seguras es parte del trabajo de la UNESCO. La Organización ha desarrollado una metodología de evaluación y fortalecimiento de los espacios de aprendizaje ante la posibilidad de enfrentar amenazas naturales de diversa índole. Con el nombre de VISUS²⁵, esta acción se realiza en conjunto con el laboratorio SPRINT de la Universidad de Udine (Italia), iniciativa que comenzó a aplicarse en 2014 en El Salvador y que se extendió a Laos e Indonesia.

Las amenazas naturales pueden tener efectos negativos en las vidas de los estudiantes, como también pueden llegar a paralizar los sistemas de educación. La UNESCO, en el marco de la Alianza Global para la Reducción del Riesgo de Desastres y la Resiliencia en el Sector Educativo, trabaja para proteger la seguridad de los estudiantes, los maestros y el personal administrativo en los centros educativos, garantizando la continuidad del sistema educativo en el caso de un desastre.

Proyecto piloto en El Salvador

El proyecto piloto de El Salvador tomó forma gracias a la colaboración de la Facultad de Ingeniería y Arquitectura de la Universidad de El Salvador. En dicho proceso, 60 personas fueron capacitadas sobre el uso de esta metodología, incluyendo a profesores, estudiantes universitarios, asociaciones de ingenieros y personal del Ministerio de Educación y fue evaluada la seguridad de 100 centros educativos de dicho país centroamericano. Este proceso proporcionó evidencia para los responsables en la toma de decisiones, como también herramientas para las comunidades educativas que pueden evaluar los riesgos que afectan la infraestructura y que ponen en peligro a quienes la ocupan.

La metodología VISUS para elaborar este tipo de asesoramiento surgió a partir de la última Conferencia Mundial sobre la Reducción del Riesgo de Desastres (WCDRR), realizada en Sendai, Japón en 2015. Allí, la UNESCO se comprometió con la Iniciativa Mundial para la Seguridad Escolar (WISS) para brindar apoyo técnico a los gobiernos bajo el marco integral de seguridad escolar. Esto implica crear políticas y planes para el sector educativo alineadas con los planes nacionales, subnacionales y locales de gestión de desastres.

²⁵ GRF Davos Planet@Risk, Volume 3, Number 1, Special Issue on the 5th IDRC Davos 2014, March 2015: VISUS Methodology: A i Assessment for Defining Safety Upgrading Strategies of Sool Facilities, 11 páginas.

En este sentido, la provechosa experiencia de El Salvador está siendo replicada en Laos e Indonesia y nuevos pilotos están previstos para ser implementados en 2016 en Haití, Perú y en otra región de Indonesia.

Anexo D: Cálculos de costos²⁶

A continuación se calculan los costos para la ejecución del proyecto SICEVED.

Administrador de Proyecto

Promedio mensual \$1200

Valor por Hora \$5

$\$5 * 20$ Horas semanales = \$100

$\$100 * 4$ semanas/mes = \$400

$\$400 * 9$ meses = \$3600

Programador / Analista

Promedio mensual = \$800

Valor por Hora = \$3.33

$\$3.33 * 20$ Horas semanales = \$66.66

$\$66.66 * 4$ semanas/mes = \$266.64

$\$266.64 * 9$ meses = \$2399.76

Administrador de Base de Datos

Promedio mensual = \$1163

Valor por Hora = \$4.85

$\$4.85 * 20$ Horas semanales = \$97

$\$97 * 4$ semanas/mes = \$388

$\$388 * 9$ meses = \$3492

Costos fijos por alquiler de local de trabajo: Escritorio

Costo mensual por persona = \$48

Número de persona = 5

Costo mensual= $\$48 * 5 = \240

Depreciación estaciones de trabajo

Valor total \$2000

Depreciación 20% (5 años)

$\$2000 / 5$ años = \$400 (anual)

$\$400 / 12$ meses = \$33.33 (mensual)

Depreciación servidor

Valor total \$3000

Depreciación 20% (5 años)

$\$3000 / 5$ años = \$600 (anual)

$\$600 / 12$ meses = \$50 (mensual)

Depreciación dispositivo móvil

Valor total \$250

Depreciación 20% (5 años)

$\$250 / 5$ años = \$50 (anual)

$\$50 / 12$ meses = \$4.16 (mensual)

Costo mes Windows 10

Valor total \$145

Depreciación 50% (2 años)

$\$145 / 2$ años = \$72.5 (anual)

$\$72.5 / 12$ meses = \$6.04 (mensual)

²⁶ Para calcular los salarios de los desarrolladores se ha tomado de referencia de la siguiente fuente <http://www.tusalario.org>

Anexo E: Cálculo de beneficios²⁷

El beneficio se ha calculado tomando como base la reducción del tiempo por evaluación mediante el uso de VISUS.

El tiempo de evaluación de la edificación base es de 90 horas previo a la utilización del sistema y 30 horas posterior a la utilización del sistema. Teniendo una reducción del tiempo de evaluación hasta en un 33% del tiempo original.

Cálculo de costos de evaluación sin la utilización de herramientas similares a VISUS

Especialista estructural

Promedio por hora = \$6.25

Promedio de duración de evaluación de 90 horas =

$\$6.25 * 90 = \562.5

Número de personas involucradas 2

$\$562.5 * 2 = \$1,125$

Personal de apoyo a Evaluación

Promedio por hora = \$2.5

Promedio por duración de evaluación de 90 horas =

$\$2.5 * 90 = \225

Número de personas involucradas 3

$\$225 * 3 = \67

Costo de evaluación por edificio sin utilizar herramientas de software = **\$1,800.**

Cálculo de costos de evaluación con la utilización de la herramienta VISUS.

Especialista estructural

Promedio hora = \$6.25

Promedio de duración de evaluación de 30 horas =

$\$6.25 * 30 = \187.5

Número de personas involucradas 2

$\$187.5 * 2 = \375

Personal de apoyo a Evaluación

Promedio hora = \$2.5

Promedio de duración de evaluación de 30 horas =

$\$2.5 * 30 = \75

Número de personas involucradas 3

$\$75 * 3 = \225

Costo de evaluación por edificio utilizando la herramienta VISUS = **\$600.**

Ahorro al utilizar VISUS para evaluar edificaciones:

$\$1,800 - \$600 = \$1,200.$

$\$15,042.57 / \$1,200/\text{edificio} = 13$ edificios

Tomando en cuenta el costo del desarrollo del proyecto \$15,042.57 la recuperación de la inversión mediante el uso del sistema para las evaluaciones de edificios se tendría con la evaluación de **13 edificios.**

²⁷<http://www.unesco.org/new/en/natural-sciences/special-themes/disaster-riskreduction/school-safety/safety-assessment-method-visus/visus-pilot-project-in-el-salvador/>

Anexo F: Software Libre²⁸

Software libre

Software libre (en inglés free software) es el software que, una vez obtenido, puede ser usado, copiado, estudiado, modificado y redistribuido libremente. El software libre suele estar disponible gratuitamente, pero no hay que asociar software libre a software gratuito, o a precio del costo de la distribución a través de otros medios; sin embargo, no es obligatorio que sea así y, aunque conserve su carácter de libre, puede ser vendido comercialmente. Análogamente, el software gratis o gratuito (denominado usualmente freeware) incluye en algunas ocasiones el código fuente; sin embargo, este tipo de software no es libre en el mismo sentido que el software libre, al menos que se garanticen los derechos de modificación y redistribución de dichas versiones modificadas del programa.

El término free, traducido al español, significa tanto libre como gratis, por eso muchas veces suelen confundirse el freeware con el software libre, aunque entre ambos existen notables diferencias.

Libertades del Software Libre

De acuerdo con tal definición, el software es "libre" si garantiza las siguientes libertades:

- Ejecutar el programa con cualquier propósito (privado, educativo, público, comercial, militar, etc.).
- Estudiar y modificar el programa (para lo cual es necesario poder acceder al código fuente).
- Copiar el programa de manera que se pueda ayudar al vecino o a cualquiera.
- Mejorar el programa y publicar las mejoras.

Es importante señalar que algunas de estas libertades obligan a que se tenga acceso al código fuente.

El término software no libre se emplea para referirse al software distribuido bajo una licencia de software más restrictiva que no garantiza estas cuatro libertades. Las leyes de la propiedad intelectual reservan la mayoría de los derechos de modificación, duplicación y redistribución para el dueño del copyright; el software dispuesto bajo una licencia de software libre rescinde específicamente la mayoría de estos derechos reservados.

La definición de software libre no contempla el asunto del precio, y es habitual ver a la venta CDs de software libre como distribuciones Linux. Sin embargo, en esta situación, el comprador del CD tiene el derecho de copiarlo y redistribuirlo. El software gratis puede incluir restricciones que no se adaptan a la definición de software libre —por ejemplo, puede no incluir el código fuente, puede prohibir explícitamente a los distribuidores recibir una compensación a cambio, etc.

Para evitar la confusión, algunas personas utilizan los términos "libre" (Libre software) y "gratis" (Gratis software) para evitar la ambigüedad de la palabra inglesa "free". Sin embargo, estos términos alternativos son usados únicamente dentro del movimiento del software libre, aunque

²⁸ <https://blog.desdelinux.net/que-es-el-software-libre/>

están extendiéndose lentamente hacia el resto del mundo. Otros defienden el uso del término open source software (software de código abierto, también llamado de fuentes abiertas). La principal diferencia entre los términos "open source" y "free software" es que éste último tiene en cuenta los aspectos éticos y filosóficos de la libertad, mientras que el "open source" se basa únicamente en los aspectos técnicos.

Anexo G: Ley de Fomento y Protección de la Propiedad Intelectual²⁹

Ley de Fomento y Protección de la Propiedad Intelectual, fue emitida según decreto legislativo del 15 de julio del año 93, publicada en el diario oficial Número 150, tomo 320 del 16 de agosto del año 93 y cuenta con un reglamento de conformidad con decreto ejecutivo Número 35 del 28 de septiembre de año 1994 número 190, tomo 325, del 6 de octubre del año 1994.

Sección E

PROGRAMAS DE ORDENADOR

Artículo 32, Ley de fomento y protección de la propiedad intelectual

Ya sea un programa fuente o programa objeto, es la obra literaria constituida por un conjunto de instrucciones expresadas mediante palabras, códigos, planes o en cualquier otra forma que, al ser incorporadas en un dispositivo de lectura automatizada, es capaz de hacer que un ordenador, o sea, un aparato electrónico o similar capaz de elaborar informaciones, ejecute determinada tarea u obtenga determinado resultado se presume que es productor del programa de ordenador, la persona que aparezca indicada como tal en la obra de manera acostumbrada, salvo prueba en contrario.

Artículo 33, Ley de fomento y protección de la propiedad intelectual

El contrato entre los autores del programa de ordenador y el productor, implica la cesión ilimitada y exclusiva a favor de éste de los derechos patrimoniales reconocidos en la presente ley, así como la autorización para decidir sobre la divulgación y la de ejercer los derechos morales sobre la obra, en la medida que ello sea necesario para la explotación de la misma, salvo pacto en contrario.

Posteriormente existe el convenio de Berna para la protección de los derechos de autor del cual El Salvador es parte desde el 19 de febrero del año 94. Y en dicho tratado en su artículo 2 protege el derecho de autor especialmente el programa de ordenador, luego aparece el acuerdo sobre los aspectos de los derechos de propiedad intelectual relacionados con el comercio de la ronda de Uruguay que en su artículo 10 protege los programas de ordenador y El Salvador es parte desde el 28 de abril de 1995.

Anexo H: JavaScript³⁰

JavaScript (a veces abreviado como JS) es un lenguaje ligero e interpretado, orientado a objetos con funciones de primera clase, más conocido como el lenguaje de script para páginas web, pero también usado en muchos entornos sin navegador, tales como node.js o Apache CouchDB. Es

²⁹ http://www.wipo.int/wipolex/es/text.jsp?file_id=129723

³⁰ <https://developer.mozilla.org/es/docs/Web/JavaScript>

un lenguaje script multi-paradigma, basado en prototipos, dinámico, soporta estilos de programación funcional, orientada a objetos e imperativa.

Características

Las siguientes características son comunes a todas las implementaciones que se ajustan al estándar ECMAScript, a menos que especifique explícitamente en caso contrario.

Imperativo y estructurado

JavaScript es compatible con gran parte de la estructura de programación de C (por ejemplo, sentencias if, bucles for, sentencias switch, etc.). Con una salvedad, en parte: en C, el ámbito de las variables alcanza al bloque en el cual fueron definidas; sin embargo, JavaScript no es compatible con esto, puesto que el ámbito de las variables es el de la función en la cual fueron declaradas. Esto cambia con la versión de ECMAScript 2015, ya que añade compatibilidad con block scoping por medio de la palabra clave let. Como en C, JavaScript hace distinción entre expresiones y sentencias. Una diferencia sintáctica con respecto a C es la inserción automática de punto y coma, es decir, en JavaScript los puntos y coma que finalizan una sentencia pueden ser omitidos.

Dinámico

- **Tipado dinámico**

Como en la mayoría de los lenguajes de scripting, el tipo está asociado al valor, no a la variable. Por ejemplo, una variable x en un momento dado puede estar ligada a un número y más adelante, religada a una cadena. JavaScript es compatible con varias formas de comprobar el tipo de un objeto, incluyendo duck typing. Una forma de saberlo es por medio de la palabra clave typeof.

- **Objetual**

JavaScript está formado casi en su totalidad por objetos. Los objetos en JavaScript son arrays asociativos, mejorados con la inclusión de prototipos (ver más adelante). Los nombres de las propiedades de los objetos son claves de tipo cadena: obj.x = 10 y obj['x'] = 10 son equivalentes, siendo la notación con punto azúcar sintáctico. Las propiedades y sus valores pueden ser creados, cambiados o eliminados en tiempo de ejecución. La mayoría de las propiedades de un objeto (y aquellas que son incluidas por la cadena de la herencia prototípica) pueden ser enumeradas a por medio de la instrucción de bucle for... in. JavaScript tiene un pequeño número de objetos predefinidos como son Function y Date.

- **Evaluación en tiempo de ejecución**

JavaScript incluye la función eval que permite evaluar expresiones expresadas como cadenas en tiempo de ejecución. Por ello se recomienda que eval sea utilizado con precaución y que se opte por utilizar la función JSON.parse() en la medida de lo posible, pues resulta mucho más segura.

Funcional

- **Funciones de primera clase**

A las funciones se les suele llamar ciudadanos de primera clase; son objetos en sí mismos. Como tal, poseen propiedades y métodos, como `.call()` y `.bind()`. Una función anidada es una función definida dentro de otra. Esta es creada cada vez que la función externa es invocada. Además, cada función creada forma una clausura; es el resultado de evaluar un ámbito conteniendo en una o más variables dependientes de otro ámbito externo, incluyendo constantes, variables locales y argumentos de la función externa llamante. El resultado de la evaluación de dicha clausura forma parte del estado interno de cada objeto función, incluso después de que la función exterior concluya su evaluación.

Prototípico

- **Prototipos**

JavaScript usa prototipos en vez de clases para el uso de herencia. Es posible llegar a emular muchas de las características que proporcionan las clases en lenguajes orientados a objetos tradicionales por medio de prototipos en JavaScript.

- **Funciones como constructores de objetos**

Las funciones también se comportan como constructores. Prefijar una llamada a la función con la palabra clave `new` crear una nueva instancia de un prototipo, que heredan propiedades y métodos del constructor (incluidas las propiedades del prototipo de `Object`).

ECMAScript 5 ofrece el método `Object.create`, permitiendo la creación explícita de una instancia sin tener que heredar automáticamente del prototipo de `Object` (en entornos antiguos puede aparecer el prototipo del objeto creado como `null`). La propiedad `prototype` del constructor determina el objeto usado para el prototipo interno de los nuevos objetos creados. Se pueden añadir nuevos métodos modificando el prototipo del objeto usado como constructor. Constructores predefinidos en JavaScript, como `Array` u `Object`, también tienen prototipos que pueden ser modificados. Aunque esto sea posible se considera una mala práctica modificar el prototipo de `Object` ya que la mayoría de los objetos en Javascript heredan los métodos y propiedades del objeto `prototype`, objetos los cuales pueden esperar que estos no hayan sido modificados.

Anexo I: MongoDB³¹

MongoDB

MongoDB es un sistema de base de datos NoSQL orientado a documentos, desarrollado bajo el concepto de código abierto; formando parte de la nueva familia de base de datos NoSQL, ya que en lugar de guardar los datos como tablas como se hace en las bases de datos relacionales, guarda estructuras de datos en documentos similares a JSON con un esquema dinámico (MongoDB utiliza una especificación llamada BSON), haciendo que la integración de los datos en ciertas aplicaciones sea más fácil y rápida.

MongoDB es escalable horizontalmente, lo que significa que para agregar capacidad, un administrador de base de datos puede simplemente agregar más servidores de productos básicos o instancias de nube. La base de datos distribuye automáticamente los datos entre servidores según sea necesario. Además, permite:

- Aprovechar los datos y la tecnología para maximizar la ventaja competitiva.
- Reduzca el riesgo de despliegues de misión crítica
- Acelere el tiempo de valoración
- Reducir drásticamente el costo total de propiedad

Con MongoDB, puede crear aplicaciones que nunca fueron posibles con las bases de datos relacionales tradicionales. Así es cómo.

- Desarrollo rápido e iterativo. El deslizamiento del alcance y los requerimientos cambiantes del negocio ya no se interponen entre usted y la entrega exitosa del proyecto. Un modelo de datos flexible, junto con el esquema dinámico y los controladores de lenguajes de programación hacen que sea rápido para los desarrolladores construir y evolucionar aplicaciones. El aprovisionamiento y la gestión automatizados permiten la integración continua y las operaciones altamente productivas. Contraste esto contra esquemas relacionales estáticos y operaciones complejas que te han obstaculizado en el pasado.
- Modelo de datos flexible. El modelo de datos de documentos de MongoDB facilita el almacenamiento y la combinación de datos de cualquier estructura, sin renunciar a las sofisticadas reglas de validación, al acceso a los datos y a la amplia funcionalidad de indexación. Puede modificar dinámicamente el esquema sin tiempo de inactividad. Pasa menos tiempo preparando sus datos para la base de datos, y más tiempo poniendo sus datos a trabajar.
- Multi-Datacenter Escalabilidad. MongoDB puede escalarse dentro y entre centros de datos distribuidos geográficamente, proporcionando nuevos niveles de disponibilidad y escalabilidad. A medida que sus implementaciones crecen en términos de volumen de datos y rendimiento, MongoDB se puede escalar fácilmente sin tiempo de inactividad y sin cambiar su aplicación. Y a medida que evolucionan sus objetivos de disponibilidad y

³¹ <https://docs.mongodb.com/manual/>

recuperación, MongoDB le permite adaptarse con flexibilidad, a través de centros de datos, con una consistencia sintonizable.

- Conjunto de funciones integrado. Análisis y visualización de datos, búsqueda de texto, procesamiento de gráficos, geoespacial, rendimiento en memoria y replicación global le permiten ofrecer una amplia variedad de aplicaciones en tiempo real en una tecnología, confiable y segura. Los sistemas RDBMS requieren tecnologías adicionales y complejas que exigen gastos generales de integración y gastos para hacerlo bien.
- Menor TCO. Los equipos de desarrollo de aplicaciones son más productivos cuando usan MongoDB. Gestión de un solo clic significa que los equipos de operaciones también lo están. MongoDB se ejecuta en hardware de productos básicos, reduciendo drásticamente los costos. Por último, MongoDB ofrece suscripciones anuales asequibles, incluido el soporte global 24x7x365. Sus aplicaciones pueden ser una décima parte del costo de entrega en comparación con el uso de una base de datos relacional.
- Compromiso a largo plazo. MongoDB Inc y el ecosistema MongoDB están detrás de la base de datos de más rápido crecimiento del mundo. 20 millones de descargas y más de 2.000 clientes, incluyendo más del 50% de Fortune 100. Más de 1.000 socios y una mayor financiación de inversores que cualquier otra base de datos de la historia. Puede estar seguro de que su inversión está protegida.

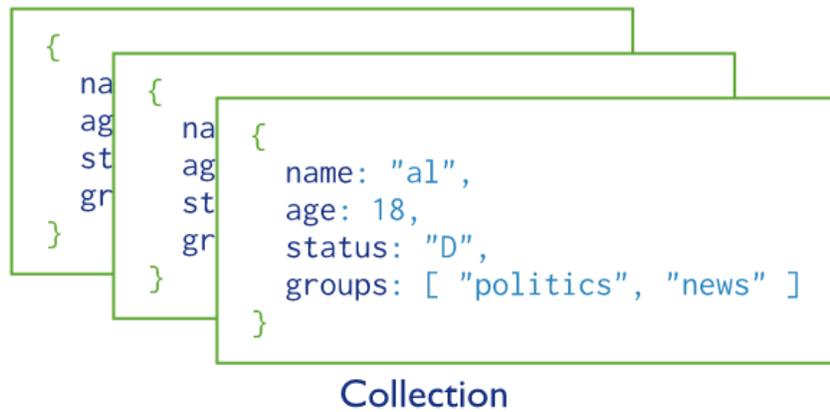
Terminología y conceptos:

La siguiente tabla presenta los diversos términos y conceptos de SQL y la terminología y los conceptos correspondientes de MongoDB.

SQL Terms/Concepts	MongoDB Terms/Concepts
database	<u>database</u>
table	<u>collection</u>
row	<u>document</u> or <u>BSON</u> document
column	<u>field</u>
index	<u>index</u>
table joins	<u>\$lookup</u> , embedded documents
primary key Specify any unique column or column combination as primary key.	<u>primary key</u> In MongoDB, the primary key is automatically set to the <u>_id</u> field.

Estructura:

- Bases de datos: conjunto de colecciones. Son dinámicas, solo existen si hay almacenadas en ellas, colecciones o documentos.
- Colecciones: MongoDB almacena documentos en colecciones. Las colecciones son análogas a las tablas de las bases de datos relacionales.



Documentos: MongoDB almacena registros de datos como documentos BSON. BSON es una representación binaria de documentos JSON , aunque contiene más tipos de datos que JSON. Para la especificación BSON, vea bsonspec.org . Consulte también Tipos BSON .

```

{
  name: "sue",
  age: 26,
  status: "A",
  groups: [ "news", "sports" ]
}

```

← field: value
 ← field: value
 ← field: value
 ← field: value

Estructura del documento

Los documentos de MongoDB se componen de pares de campo y valor y tienen la siguiente estructura:

```

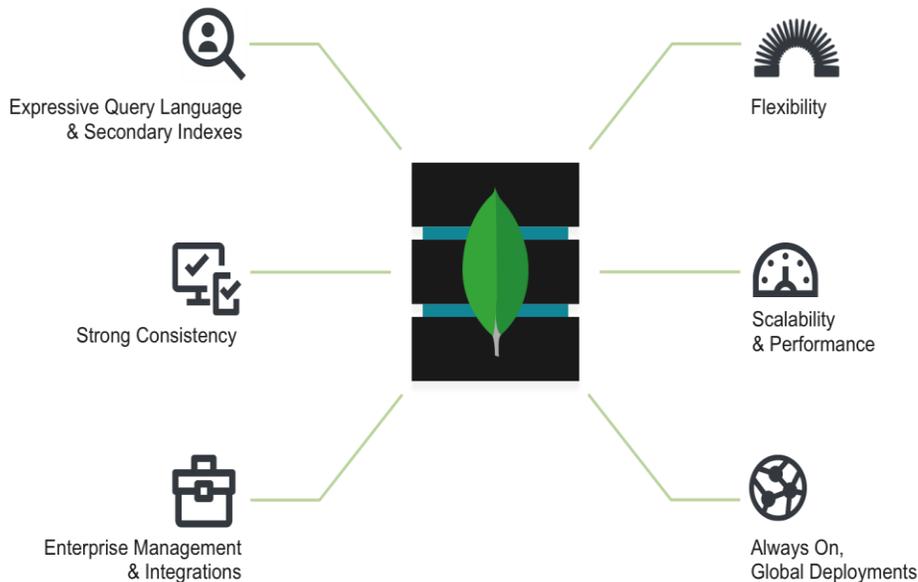
{
  Campo1 : valor1 ,
  campo2 : valor2 ,
  field3 : valor3 ,
  ...
  campoN : valorN
}

```

La Arquitectura Nexus

La filosofía de diseño de MongoDB se centra en combinar las capacidades críticas de las bases de datos relacionales con las innovaciones de las tecnologías NoSQL. Nuestra visión es

aprovechar el trabajo que Oracle y otros han hecho durante los últimos 40 años para hacer bases de datos relacionales lo que son hoy en día. En lugar de descartar décadas de probada madurez de la base de datos, MongoDB está recogiendo donde dejaron de combinar las capacidades de base de datos relacionales con el trabajo que los pioneros de Internet han hecho para satisfacer los requerimientos de las aplicaciones modernas.



Las bases de datos relacionales han servido de forma fiable las aplicaciones durante muchos años y ofrecen características que siguen siendo críticas hoy en día a medida que los desarrolladores crean la siguiente generación de aplicaciones:

- Lenguaje de consulta expresivo e índices secundarios. Los usuarios deben ser capaces de acceder y manipular sus datos de manera sofisticada para apoyar tanto las aplicaciones operacionales y analíticas. Los índices juegan un papel crítico en la provisión de acceso eficiente a los datos, soportados nativamente por la base de datos en lugar de mantenerse en el código de la aplicación.
- Fuerte consistencia. Las aplicaciones deben ser capaces de leer inmediatamente lo que se ha escrito en la base de datos. Es mucho más complejo construir aplicaciones alrededor de un modelo eventualmente consistente, imponiendo un trabajo significativo en el desarrollador, incluso para los equipos de ingeniería más sofisticados.
- Gestión Empresarial e Integraciones. Las bases de datos son sólo una pieza de la infraestructura de aplicaciones y deben encajar perfectamente en la pila de TI de la empresa. Las organizaciones necesitan una base de datos que pueda ser asegurada, supervisada, automatizada e integrada con su infraestructura tecnológica, procesos y personal existentes, incluyendo equipos de operaciones, DBAs y analistas de datos.

Sin embargo, las aplicaciones modernas imponen requisitos no abordados por las bases de datos relacionales, y esto ha impulsado el desarrollo de bases de datos NoSQL que ofrecen:

- **Modelo de datos flexible:** Las bases de datos NoSQL surgieron para abordar los requisitos de los datos que vemos dominando las aplicaciones modernas. Ya sea documento, gráfico, valor-clave o columna ancha, todos ellos ofrecen un modelo de datos flexible, lo que facilita el almacenamiento y la combinación de datos de cualquier estructura y permite la modificación dinámica del esquema sin tiempo de inactividad ni impacto en el rendimiento.
- **Escalabilidad y rendimiento:** Las bases de datos NoSQL fueron construidas con un enfoque de escalabilidad, por lo que todas incluyen alguna forma de fragmentación o partición. Esto permite a la base de datos escalar hacia fuera en el hardware de la materia puesto en uso en premisas o en la nube, permitiendo el crecimiento casi ilimitado con un rendimiento más alto y una latencia más baja que bases de datos relacionales.
- **Implementaciones globales siempre activas:** Las bases de datos NoSQL están diseñadas para sistemas de alta disponibilidad que proporcionan una experiencia consistente y de alta calidad para usuarios de todo el mundo. Están diseñados para ejecutarse en muchos nodos, incluida la replicación para sincronizar automáticamente los datos entre servidores, bastidores y centros de datos.

Al ofrecer estas innovaciones, los sistemas NoSQL han sacrificado las capacidades críticas que las personas han llegado a esperar y depender de las bases de datos relacionales. MongoDB ofrece un enfoque diferente. Con su Nexus Architecture, MongoDB es la única base de datos que aprovecha las innovaciones de NoSQL al tiempo que mantiene la base de las bases de datos relacionales.

Características principales

- **Consultas Ad hoc:** MongoDB soporta la búsqueda por campos, consultas de rangos y expresiones regulares. Las consultas pueden devolver un campo específico del documento pero también puede ser una función definida por el usuario.
- **Almacenamiento de archivos:** MongoDB puede ser utilizado como un sistema de archivos, tomando la ventaja de la capacidad que tiene MongoDB para el balanceo de carga y la replicación de datos utilizando múltiples servidores para el almacenamiento de archivos. Esta función se llama GridFS y es más bien una implementación en los drivers, no en el servidor, por lo que está incluida en los drivers oficiales que la compañía de MongoDB desarrolla. Estos drivers exponen funciones y métodos para la manipulación de archivos y contenido a los desarrolladores. En un sistema con múltiples servidores, los archivos pueden ser distribuidos y replicados entre los mismos y de una forma transparente, de esta forma se crea un sistema eficiente que maneja fallos y balanceo de carga.
- **Agregación:** MongoDB proporciona un framework de agregación que permite realizar operaciones similares a las que se obtienen con el comando SQL "GROUP BY". El framework de agregación está construido como un pipeline en el que los datos van pasando a través de diferentes etapas en las cuales estos datos son modificados, agregados, filtrados y formateados hasta obtener el resultado deseado. Todo este

procesado es capaz de utilizar índices si existieran y se produce en memoria. Asimismo, MongoDB proporciona una función MapReduce que puede ser utilizada para el procesamiento por lotes de datos y operaciones de agregación.

- **Ejecución de JavaScript del lado del servidor:** MongoDB tiene la capacidad de realizar consultas utilizando JavaScript, haciendo que estas sean enviadas directamente a la base de datos para ser ejecutadas.
- **Replicación:** MongoDB soporta el tipo de replicación primario-secundario. Cada grupo de primario y sus secundarios se denomina replica set. El primario puede ejecutar comandos de lectura y escritura. Los secundarios replican los datos del primario y sólo se pueden usar para lectura o para copia de seguridad, pero no se pueden realizar escrituras. Los secundarios tiene la habilidad de poder elegir un nuevo primario en caso de que el primario actual deje de responder.

Modelo de datos MongoDB

MongoDB Arquitectura de Multimodelos

MongoDB permite a los usuarios mezclar y combinar múltiples motores de almacenamiento dentro de un único despliegue. Esta flexibilidad proporciona un enfoque más simple y confiable para satisfacer las diversas necesidades de aplicación de datos. Tradicionalmente, se necesitaría administrar múltiples tecnologías de bases de datos para satisfacer estas necesidades, con un código de integración personalizado y complejo para mover datos entre las tecnologías y para garantizar un acceso coherente y seguro. Con la arquitectura de almacenamiento flexible de MongoDB, la base de datos gestiona automáticamente el movimiento de datos entre las tecnologías de los motores de almacenamiento mediante la replicación nativa.

- El modelo de datos de documentos flexible de MongoDB presenta un superconjunto de otros modelos de base de datos. Permite que los datos se representen como simples pares clave-valor y planos, estructuras similares a tablas, a través de documentos ricos y objetos con matrices y subdocumentos profundamente anidados.
- Con un expresivo lenguaje de consulta, los documentos se pueden consultar de muchas maneras: desde búsquedas sencillas hasta crear sofisticadas líneas de procesamiento para análisis de datos y transformaciones, hasta búsqueda con facetas, JOINS y recorridos de gráficos.
- Con una arquitectura de almacenamiento flexible, los propietarios de aplicaciones pueden implementar motores de almacenamiento optimizados para diferentes requisitos de carga de trabajo y operativos.

El diseño multimodelos de MongoDB reduce significativamente la complejidad de los desarrolladores y las operaciones en comparación con la ejecución de varias tecnologías de bases de datos distintas para satisfacer las diferentes necesidades de las aplicaciones. Los usuarios pueden aprovechar el mismo lenguaje de consulta MongoDB, modelo de datos, escalabilidad, seguridad y herramientas operativas en diferentes partes de su aplicación, cada una de ellas con el motor de almacenamiento óptimo.

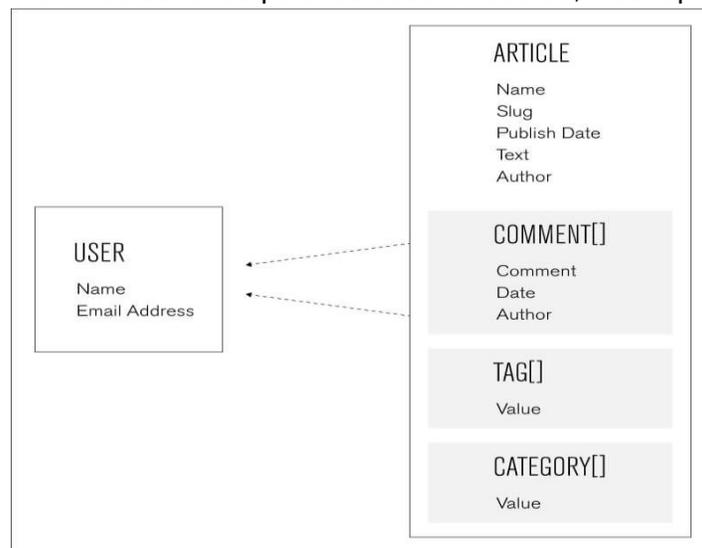
Datos como documentos

MongoDB almacena los datos como documentos en una representación binaria llamada BSON (Binary JSON). Los documentos que comparten una estructura similar se organizan típicamente como colecciones. Se puede pensar que las colecciones son análogas a una tabla en una base de datos relacional: los documentos son similares a las filas y los campos son similares a las columnas.

Los documentos MongoDB tienden a tener todos los datos de un registro dado en un solo documento, mientras que en una base de datos relacional la información para un registro dado suele estar repartida entre muchas tablas.

Por ejemplo, considere el modelo de datos para una aplicación de blogs. En una base de datos relacional, el modelo de datos comprendería varias tablas como Categorías, Etiquetas, Usuarios, Comentarios y Artículos. En MongoDB los datos podrían ser modelados como dos colecciones, una para los usuarios, y la otra para los artículos. En cada documento de blog puede haber varios comentarios, múltiples etiquetas y varias categorías, cada una de ellas expresada como una matriz incrustada.

Datos como documentos: más sencillos para los desarrolladores, más rápidos para los usuarios.



Como resultado del modelo de documento, los datos en MongoDB son más localizados, lo que reduce drásticamente la necesidad de JOIN tablas separadas. El resultado es un rendimiento y una escalabilidad dramáticamente mayores en el hardware de productos básicos, ya que una única lectura a la base de datos puede recuperar todo el documento.

A diferencia de muchas bases de datos NoSQL, los usuarios no necesitan renunciar completamente a JOIN. Para mayor flexibilidad analítica, MongoDB preserva la semántica izquierda-externa JOIN con el operador \$ lookup, lo que permite a los usuarios obtener lo mejor del modelado de datos relacional y no relacional.

Además, los documentos MongoDB están más alineados con la estructura de los objetos en el lenguaje de programación. Esto hace que sea más sencillo y rápido para los desarrolladores modelar cómo los datos de la aplicación se asignan a los datos almacenados en la base de datos.

Esquema dinámico de MongoDB con control de datos

Los documentos de MongoDB pueden variar en estructura. Por ejemplo, todos los documentos que describen a los usuarios pueden contener el ID de usuario y la última fecha en que se conectaron al sistema, pero sólo algunos de estos documentos pueden contener la identidad del usuario para una o más aplicaciones de terceros.

Los campos pueden variar de un documento a otro; No hay necesidad de declarar la estructura de los documentos al sistema - los documentos son auto-descriptivos. Si se necesita añadir un nuevo campo a un documento, el campo puede crearse sin afectar a todos los demás documentos del sistema, sin actualizar un catálogo central del sistema y sin desconectar el sistema.

MongoDB permite a los desarrolladores diseñar y evolucionar el esquema a través de un enfoque iterativo y ágil.

Los desarrolladores pueden comenzar a escribir código y persistir los objetos a medida que se crean. Y cuando los desarrolladores añaden más funciones, MongoDB sigue almacenando los objetos actualizados sin necesidad de realizar costosas operaciones ALTER_TABLE, o peor: tener que volver a diseñar el esquema desde cero.

Los esquemas dinámicos brindan gran agilidad, pero también es importante que se puedan implementar controles para mantener la calidad de los datos. A diferencia de las bases de datos NoSQL que empujan la aplicación de estos controles de nuevo en código de aplicación, MongoDB proporciona validación de documentos dentro de la base de datos. Los usuarios pueden hacer cumplir controles sobre la estructura del documento, los tipos de datos, rangos de datos y la presencia de campos obligatorios. {Como resultado, los DBA pueden aplicar estándares de gestión de datos, mientras que los desarrolladores mantienen los beneficios de un modelo de documento flexible.

¿Cómo se acumula el modelo de datos MongoDB hasta las bases de datos relacionales y los almacenes de valores clave? Echa un vistazo a la siguiente tabla:

	MongoDB	Relacional	Valor clave
Modelo de datos enriquecidos	Sí	No	No
Esquema dinámico	Sí	No	Sí
Validación de datos	Sí	Sí	No
Datos mecanografiados	Sí	Sí	No

Localidad de datos	Sí	No	Sí
Actualizaciones de campo	Sí	Sí	No
Fácil para programadores	Sí	No	No cuando se modelan estructuras de datos complejas

Gestión de esquemas

El shell mongo es un shell interactivo de JavaScript que se incluye con todas las distribuciones de MongoDB. Además MongoDB Compass es una GUI sofisticada e intuitiva para MongoDB. Ofreciendo una rica exploración y administración de esquemas, Compass permite a los DBA modificar documentos, crear reglas de validación y optimizar el rendimiento de las consultas visualizando los planes de explicación y el uso del índice. Las consultas sofisticadas pueden ser construidas y ejecutadas simplemente seleccionando elementos del documento desde la interfaz de usuario, con los resultados vistos tanto gráficamente como como un conjunto de documentos JSON. Todas estas tareas se pueden realizar desde una interfaz de punto y clic, y todas con cero conocimientos del lenguaje de consulta de MongoDB.

MongoDB Compass se incluye con las suscripciones MongoDB Professional y MongoDB Enterprise Advanced usadas con las instancias autogestionadas o las instancias de MongoDB Atlas alojadas. MongoDB Compass es libre de usar para la evaluación y en entornos de desarrollo.

MongoDB Modelo de consulta y visualización de datos

Drivers de lenguajes de programación

MongoDB proporciona controladores nativos para todos los lenguajes de programación populares y marcos para hacer que el desarrollo sea natural. Los controladores soportados incluyen Java, Javascript, .NET, Python, Perl, PHP, Scala y otros, además de 30 + drivers desarrollados por la comunidad. Los controladores MongoDB están diseñados para ser idiomáticos para el idioma dado.

Con el intuitivo modelo de datos de documentos, el esquema dinámico y los controladores idiomáticos, puede crear aplicaciones y llegar al mercado más rápidamente con MongoDB.

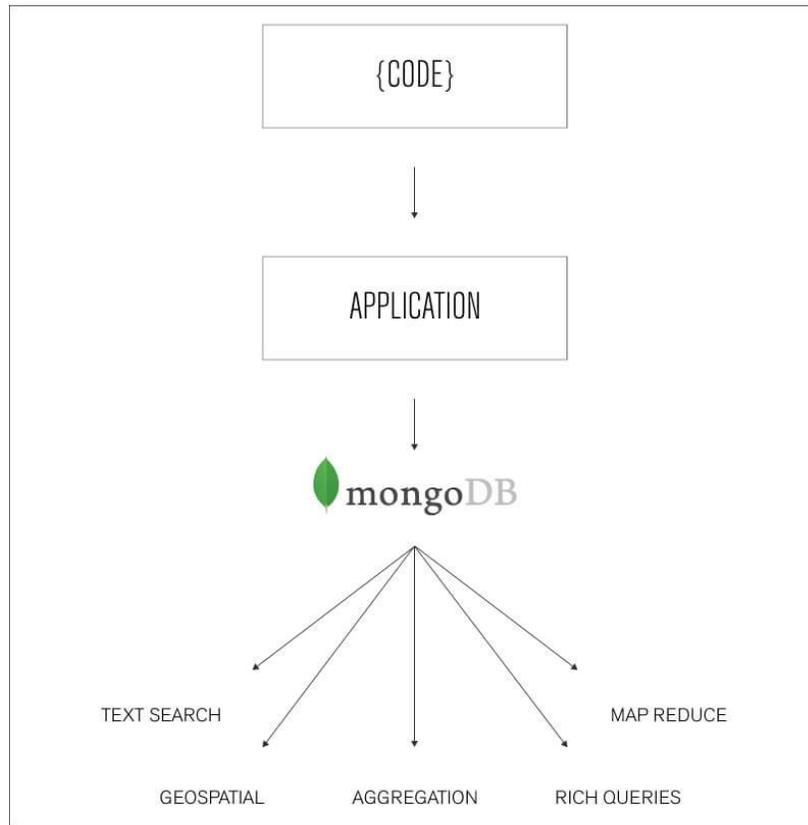
Tipos de consulta

A diferencia de las bases de datos NoSQL, MongoDB no se limita a simples operaciones de valor clave. Los desarrolladores pueden crear aplicaciones complejas utilizando consultas complejas, agregaciones e índices secundarios que desbloquean el valor en datos estructurados, semiestructurados y no estructurados.

Un elemento clave de esta flexibilidad es el apoyo de MongoDB para muchos tipos de consultas. Una consulta puede devolver un documento, un subconjunto de campos específicos dentro del documento o agregaciones complejas y la transformación de muchos documentos:

- Las consultas de valor-clave devuelven resultados basados en cualquier campo del documento, a menudo la clave principal.
- Las consultas de rangos devuelven resultados basados en valores definidos como desigualdades (por ejemplo, mayor que, menor o igual que entre).
- Las consultas geoespaciales devuelven resultados basados en criterios de proximidad, intersección e inclusión según lo especificado por un punto, una línea, un círculo o un polígono.
- Las consultas de búsqueda: devuelven los resultados en orden de relevancia y en grupos, basándose en argumentos de texto que utilizan operadores booleanos (por ejemplo, `AND`, `OR`, `NOT`) ya que a través de bucketing, agrupación y conteo de resultados de consultas. Con el apoyo de intercalaciones, la comparación de datos y orden de clasificación se puede definir para más de 100 idiomas y locales diferentes.
- Framework de agregación: devuelven las agregaciones y las transformaciones de los valores devueltos por la consulta (por ejemplo, count, min, max, average, similar a una sentencia SQL GROUP BY).
- JOINS y recorridos de gráficos. A través del pipeline de agregación \$lookup, los documentos separados por colecciones se pueden combinar a través de una operación JOIN. \$graphLookup brinda el procesamiento nativo de gráficos dentro de MongoDB, permitiendo recorridos eficientes a través de árboles, gráficos y datos jerárquicos para descubrir patrones y superficies de conexiones previamente no identificadas.
- Las consultas MapReduce ejecutan procesamiento de datos complejos que se expresa en JavaScript y se ejecuta a través de datos en la base de datos.

Además, el Conector MongoDB para Apache Spark expone las bibliotecas Scala, Java, Python y R de Spark. Los datos de MongoDB se materializan como DataFrames y conjuntos de datos para el análisis a través de aprendizaje automático, gráficos, streaming e APIs de SQL.



Indexación

Los índices son un mecanismo crucial para optimizar el rendimiento y la escalabilidad del sistema, a la vez que proporciona un acceso flexible a sus datos. MongoDB incluye soporte para muchos tipos de índices secundarios que se pueden declarar en cualquier campo del documento, incluidos los campos dentro de arrays:

- Puede definir índices compuestos, únicos, de matriz, parcial, TTL, geoespacial, escasa, hash y de texto para optimizar para múltiples patrones de consulta, tipos de datos multi-estructurados y restricciones.
- La intersección de índices permite que MongoDB utilice más de un índice para optimizar una consulta ad-hoc en tiempo de ejecución.

¿Cómo se acumula la consulta y el modelo de indexación de MongoDB en las bases de datos relacionales y en los almacenes de valores clave? Echa un vistazo a la siguiente tabla:

	MongoDB	Relacional	Valor clave
Consultas de valor-clave	Sí	Sí	Sí
Índices secundarios	Sí	Sí	No
Intersección del índice	Sí	Sí	No

Consultas de rango	Sí	Sí	No
Geoespacial	Sí	Add-on caro	No
Facetas de búsqueda	Sí	No	No
Agregación y transformación	Sí	Sí	No
Mapa reducido	Sí	No	Sí
Conductores idiomáticos	Sí	No	No
Izquierda JOINS (\$ Lookup)	Sí	Sí	No
Procesamiento de gráficos (\$ graphLookup)	Sí	No	No

Gestión de datos MongoDB

Auto-sharding para escalabilidad lineal

MongoDB proporciona escala horizontal para las bases de datos de bajo costo, hardware de productos básicos mediante una técnica llamada sharding, que es transparente para las aplicaciones. Sharding distribuye datos a través de múltiples particiones físicas llamadas fragmentos. El Sharding permite que las implementaciones de MongoDB resuelvan las limitaciones de hardware de un solo servidor, como cuellos de botella en RAM o E/S de disco, sin agregar complejidad a la aplicación. MongoDB equilibra automáticamente los datos del clúster a medida que crecen los datos o aumenta o disminuye el tamaño del clúster.

El desprendimiento es transparente para las aplicaciones; Si hay uno o cien fragmentos, el código de aplicación para consultar MongoDB es el mismo.



A diferencia de las bases de datos relacionales, el sharding es automático y se integra en la base de datos. Los desarrolladores no se enfrentan a la complejidad de construir la lógica de sharding en su código de aplicación, que luego necesita ser actualizado a medida que migran fragmentos. Los equipos de operaciones no necesitan desplegar software de clústeres adicional para administrar la distribución de procesos y datos.

A diferencia de otras bases de datos distribuidas, existen varias políticas de sharding que permiten a los desarrolladores y administradores distribuir datos a través de un clúster según los

patrones de consulta o la localidad de datos. Como resultado, MongoDB ofrece escalabilidad mucho mayor en un conjunto diverso de cargas de trabajo:

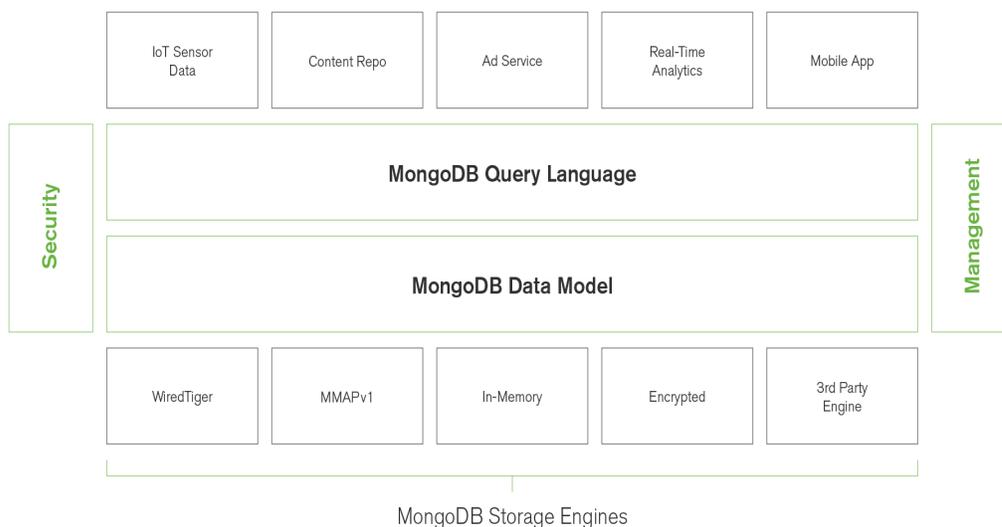
- **Range Sharding:** Los documentos se dividen en «shards» (particiones) en función del valor clave de la partición. Los documentos con un valor de clave de partición próximo es probable que se coloquen en la misma partición. Este enfoque es muy adecuado para las aplicaciones que deben optimizar consultas basadas en rangos. Hash Sharding. Los documentos se distribuyen según un hash MD5 del valor de clave de la partición. Este enfoque garantiza una distribución uniforme de las escrituras a través de las particiones, pero no funciona tan bien en consultas basadas en rangos.
- **Hash Sharding:** Los documentos se distribuyen de acuerdo con un hash MD5 del valor de clave de fragmentos. Este enfoque garantiza una distribución uniforme de escrituras entre fragmentos, pero es menos óptimo para las consultas basadas en rangos.
- **Zone Sharding:** Proporciona a los DBA y equipos de operaciones la posibilidad de definir reglas específicas que rijan la colocación de datos en un clúster fragmentado. Las zonas admiten una serie de escenarios de implementación, por ejemplo, localización de datos por región geográfica, configuración de hardware para arquitecturas de almacenamiento en niveles o función de aplicación. Los administradores pueden perfeccionar continuamente las reglas de colocación de datos modificando los rangos de claves de shard y MongoDB migrará automáticamente los datos a su nueva zona.

¿Cómo se acumulan las capacidades de escalado MongoDB en las bases de datos relacionales y en los almacenes de valores clave? A continuación se muestra la siguiente tabla:

	MongoDB	Relacional	Valor clave
Escala-hacia fuera del hardware de la materia	Sí	No	Sí
Sharding automático	Sí	No	Sí
Shard by Hash	Sí	Manual	Sí
Fragmento por rango	Sí	Manual	No
Fragmento por zona	Sí	Manual	No
Reequilibrio Automático de Datos	Sí	Manual	Limitado

Arquitectura de almacenamiento conectable para flexibilidad de aplicaciones

Con MongoDB, las organizaciones pueden abordar diversas necesidades de aplicaciones, recursos de hardware y diseños de implementación con una única tecnología de base de datos. Mediante el uso de una arquitectura de almacenamiento conectable, MongoDB puede ser ampliado con nuevas capacidades y configurado para el uso óptimo de arquitecturas de hardware específicas. Este enfoque reduce significativamente la complejidad del desarrollador y la operacional en comparación con la ejecución de varias bases de datos para alimentar aplicaciones con requisitos únicos. Los usuarios pueden aprovechar el mismo lenguaje de consulta MongoDB, el modelo de datos, la escalabilidad, la seguridad y las herramientas operativas en diferentes aplicaciones, cada uno de ellos con diferentes motores de almacenamiento MongoDB.



MongoDB se suministra con cuatro motores de almacenamiento compatibles, los cuales pueden coexistir dentro de un único conjunto de réplicas MongoDB. Esto hace que sea fácil evaluar y migrar entre ellos y optimizar los requerimientos específicos de las aplicaciones, por ejemplo, combinar el motor en memoria para operaciones de latencia ultrabaja con un motor basado en disco para la persistencia. Los motores de almacenamiento compatibles incluyen:

- El motor de almacenamiento WiredTiger predeterminado. Para muchas aplicaciones, el control de concurrencia granular de WiredTiger y la compresión nativa proporcionarán el mejor rendimiento y eficiencia de almacenamiento para el más amplio rango de aplicaciones.
- El motor de almacenamiento encriptado protege los datos altamente confidenciales, sin el rendimiento o la sobrecarga de administración del cifrado de sistema de archivos independiente. (Requiere MongoDB Enterprise Advanced)
- El motor de almacenamiento en memoria proporciona el rendimiento extremo junto con análisis en tiempo real para las aplicaciones más exigentes y sensibles a la latencia. (Requiere MongoDB Enterprise Advanced)
- El motor MMAPv1, una versión mejorada del motor de almacenamiento original utilizado en las versiones pre-3.x de MongoDB.

Almacenamiento y eficiencia de red con compresión

MongoDB soporta la compresión nativa cuando se configura con los motores de almacenamiento WiredTiger o Encrypted, reduciendo el almacenamiento físico en un 80%. Además de reducir el espacio de almacenamiento, la compresión permite una escalabilidad de E/S de almacenamiento mucho mayor a medida que se leen menos bits del disco. Los administradores tienen la flexibilidad de configurar algoritmos de compresión específicos para colecciones, índices y el diario.

Como base de datos distribuida, MongoDB se basa en el transporte eficiente de la red durante el enrutamiento de consultas y la replicación entre nodos. MongoDB ofrece la opción de comprimir el protocolo de cable utilizado para comunicaciones intra-cluster. Basado en el algoritmo de compresión rápido, el tráfico de red puede comprimirse hasta en un 70%, proporcionando grandes beneficios de rendimiento en entornos de ancho de banda restringido y reduciendo los costos de red.

Consistencia y disponibilidad de MongoDB

Modelo de transacción

MongoDB proporciona propiedades ACID en el nivel de documento. Uno o más campos se pueden escribir en una sola operación, incluyendo actualizaciones a varios subdocumentos y elementos de un arreglo. Las garantías de ACID proporcionadas por MongoDB aseguran el aislamiento completo cuando se actualiza un documento; Cualquier error provoca que la operación se vuelque para que los clientes reciban una vista coherente del documento.

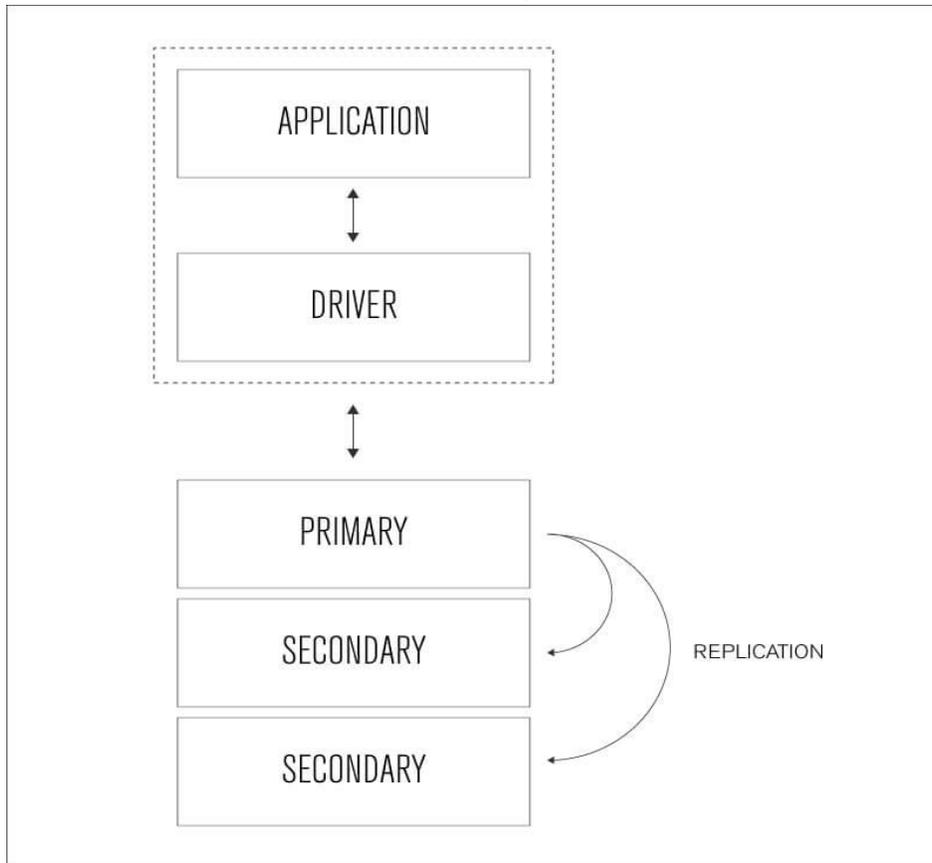
Los desarrolladores pueden utilizar las prioridades de escritura de MongoDB para configurar las operaciones para que se comprometan con la aplicación sólo después de que se hayan descargado al archivo de diario en disco. Este es el mismo modelo utilizado por muchas bases de datos relacionales tradicionales para proporcionar garantías de durabilidad. Como un sistema distribuido, MongoDB presenta flexibilidad adicional que ayuda a los usuarios a lograr sus SLAs de disponibilidad deseada. Cada consulta puede especificar el problema de escritura adecuado, como por ejemplo escribir en al menos dos réplicas en un centro de datos y una réplica en un segundo centro de datos.

Conjuntos de réplicas

MongoDB mantiene múltiples copias de los datos llamados conjuntos de réplicas utilizando replicación nativa. Un conjunto de réplicas es un 'shard' de auto-curación que ayuda a prevenir caídas en la base de datos. El intercambio de réplicas es totalmente automatizado, eliminando la necesidad de que los administradores intervengan manualmente.

El número de réplicas en un conjunto de réplicas MongoDB es configurable: un número mayor de réplicas proporciona una mayor disponibilidad de datos y protección contra el tiempo de inactividad de la base de datos (por ejemplo, en caso de múltiples fallos de la máquina, fallas en el bastidor, fallas en el centro de datos o particiones de red). Opcionalmente, las operaciones

pueden configurarse para escribir en varias réplicas antes de volver a la aplicación, proporcionando así una funcionalidad similar a la replicación síncrona.



"Los conjuntos de réplicas MongoDB ofrecen tolerancia a fallos y recuperación ante desastres. La conciencia de los centros de datos múltiples permite la distribución y separación de datos globales entre cargas de trabajo operacionales y analíticas. Los conjuntos de réplicas también proporcionan flexibilidad operativa proporcionando una forma de actualizar el hardware y el software sin necesidad de que la base de datos se desconecte".

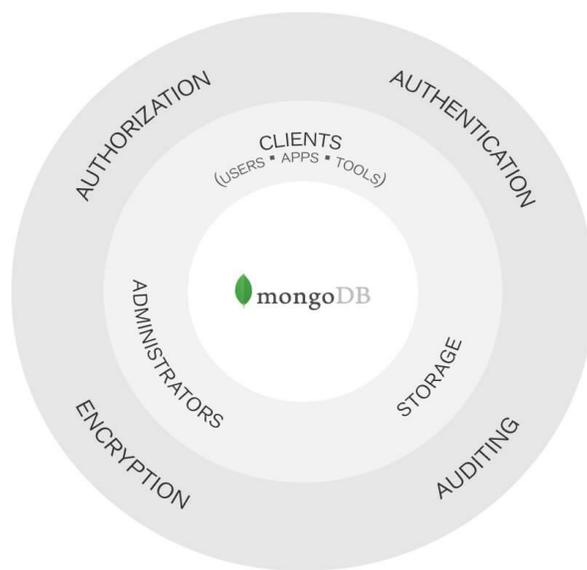
Rendimiento en memoria con capacidad en disco

Con el motor de almacenamiento en memoria, los usuarios de MongoDB pueden darse cuenta de las ventajas de rendimiento de la computación en memoria para cargas de trabajo analíticas operacionales y en tiempo real. El motor de almacenamiento en memoria ofrece el rendimiento extremo y la latencia predecible exigida por las aplicaciones más intensivas en AdTech, finanzas, telecomunicaciones, IoT, comercio electrónico y más, eliminando la necesidad de capas de caché separadas.

El conjunto de réplicas MongoDB permiten implementaciones de bases de datos en memoria y en disco híbridas. Los datos gestionados por el motor en memoria se pueden procesar y analizar en tiempo real, antes de ser replicados automáticamente a las instancias de MongoDB configuradas con uno de los motores de almacenamiento de disco permanente. Se evitan largos ciclos ETL típicos cuando se mueven datos entre diferentes bases de datos, y los usuarios ya no tienen que cambiar la capacidad escalable ni las garantías de durabilidad ofrecidas por el almacenamiento en disco.

Seguridad

La frecuencia y la dificultad de las brechas de datos siguen aumentando. Los analistas de la industria predicen que la ciberdelincuencia costará a la economía global 6 billones de dólares anuales para 2021. Las organizaciones enfrentan un ataque de nuevas clases de amenazas y actores de amenazas con robo de phishing, ransomware y propiedad intelectual creciendo más del 50% año tras año. Con las bases de datos que almacenan los activos de información más importantes de una organización, asegurarlos es una prioridad para los administradores.



- **Autenticación:** para simplificar el control de acceso a la base de datos, MongoDB ofrece integración con mecanismos de seguridad externos, incluyendo LDAP, Windows Active Directory, Kerberos y certificados PKI x.509.
- **Autorización:** las funciones definidas por el usuario permiten a los administradores configurar permisos granulares para un usuario o una aplicación basándose en los privilegios que necesitan para realizar su trabajo. Estos se pueden definir en MongoDB, o centralmente dentro de un servidor LDAP. Además, los administradores pueden definir vistas que exponen sólo un subconjunto de datos de una colección subyacente, es decir, una vista que filtra o enmascara campos específicos, como la información personalmente identificable (PII) de los datos del cliente o registros de salud.
- **Revisión de cuentas:** para el cumplimiento normativo, los administradores de seguridad pueden utilizar el registro de auditoría nativo de MongoDB para rastrear el acceso y las operaciones realizadas en la base de datos.
- **Cifrado:** Los datos MongoDB se pueden cifrar en la red, en disco y en copias de seguridad. Con el motor de almacenamiento encriptado, la protección de datos en reposo es una característica integral dentro de la base de datos. Mediante la encriptación nativa de archivos de base de datos en disco, los administradores eliminan tanto la gestión como

el rendimiento de los mecanismos de cifrado externo. Sólo el personal que tiene las credenciales de autorización de base de datos adecuadas puede acceder a los datos cifrados, proporcionando niveles adicionales de defensa.

Data geoespacial

MongoDB soporta datos geoespaciales e índices especializados que hacen que las aplicaciones de construcción con características geoespaciales sean fáciles y escalables. Una de las características más populares, el operador `$geoNear`, que devuelve los documentos en orden, de la más cercana a la más lejana con respecto a un punto determinado. Para evitar clasificar toda la colección de una vez, el `$geoNear` algoritmo amplía iterativamente su búsqueda en intervalos de distancia (el anillo rojo mostrado a continuación), con el objetivo de tener unos pocos cientos de documentos por intervalo.

La búsqueda de todos los documentos en un intervalo se logra mediante la búsqueda de una celda de índice que cubre (el conjunto de celdas de índice gris mostradas a continuación). Este recubrimiento asegura que todos los documentos en el intervalo se pueden encontrar mediante una exploración de índice. Los documentos en la cubierta pero no en el intervalo se filtran después. Después de encontrar todos los documentos en un intervalo, se ordenan y se devuelven.

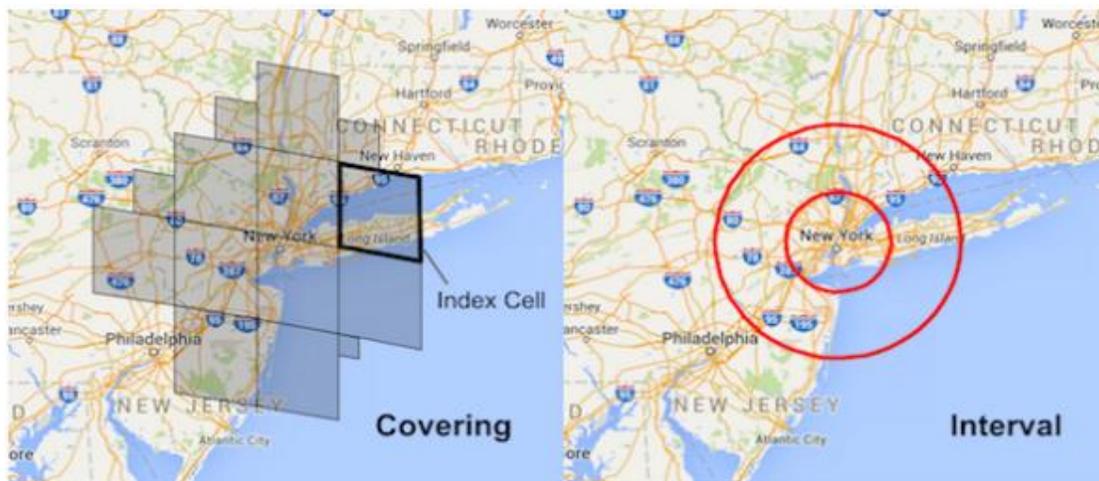


Figure 1: Interval and its corresponding Index Cell Covering

Este procedimiento de capa-filtro-y-ordenación se repite para cada intervalo durante la expansión del área de búsqueda, hasta que se alcanza un límite o se escanea todo el mundo. La siguiente imagen muestra las tres primeras etapas / intervalos de este algoritmo.

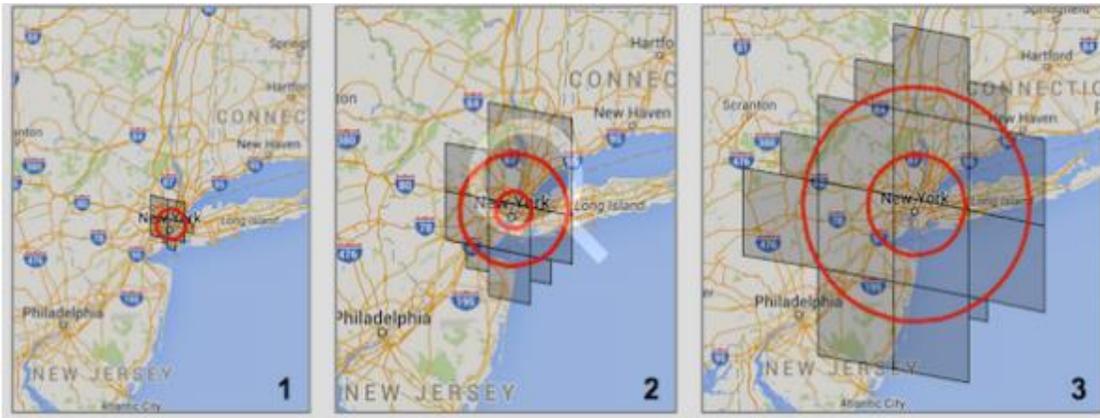


Figure 2: First three stages of a \$geoNear search.

Instrumentos de MongoDB

Los siguientes comandos pueden ser instalados para el manejo y la administración del sistema de base de datos:

- **mongo:** es un Shell interactivo que permite a los desarrolladores y administradores ver, insertar, eliminar y actualizar datos en su base de datos. Este también permite entre otras funciones la replicación de datos, configuración de sharding, apagar los servidores, ejecutar JavaScript y todos los comandos que se puedan realizar.
- **mongostat:** es un instrumento de línea de comandos que muestra en resumen una lista de estadísticas de una instancia de MongoDB en ejecución. Esto te permite visualizar cuantas inserciones, actualizaciones, eliminaciones, consultas y comandos se han ejecutado, pero también cuánta memoria está utilizando y cuánto tiempo ha estado cerrada la base de datos.
- **mongotop:** es un instrumento de línea de comandos que provee un método para dar seguimiento a la cantidad de tiempo que dura una lectura o escritura de datos en una instancia. También provee estadísticas en el nivel de cada colección.
- **mongosniff:** es un instrumento de línea de comandos que provee un sniffing en la base de datos haciendo un sniffing en el tráfico de la red que va desde y hacia MongoDB.
- **mongoimport/mongoexport:** es un instrumento de línea de comandos que facilita la importación exportación de contenido desde JSON, CSV o TSV. También tiene el potencial de importar o exportar hacia otros formatos.
- **mongodump/mongorestore:** es un instrumento de línea de comandos para la creación de una imagen binaria del contenido de la base de datos. Estos comandos son utilizados para la estrategia de copias de seguridad en MongoDB.

Limites

- Tamaño máximo de nombre de base de datos = 64 caracteres
- Tamaño máximo del documento BSON = 16 MB
- Número máximo de niveles de anidamiento = 100 niveles.
- Tamaño máximo del nombre del archivo Namespace=2047 MB
- Tamaño máximo de Base de datos =32 TB
- Tamaño máximo del Shard Key= 512 bytes.

- Data Size: Utilizando el motor de almacenamiento MMAPv1, una instancia de mongod no puede administrar un conjunto de datos que exceda el espacio de direcciones de memoria virtual máximo proporcionado por el sistema operativo subyacente.

Sistema operativo	Registrado en diario	No registrado
Linux	64 terabytes	128 terabytes
Windows Server 2012 R2 y Windows 8.1	64 terabytes	128 terabytes
Windows	4 terabytes	8 terabytes

Anexo J: Node.js³²

Node.js es un entorno en tiempo de ejecución multiplataforma, de código abierto, para la capa del servidor (pero no limitándose a ello) basado en el lenguaje de programación ECMAScript, asíncrono, con I/O de datos en una arquitectura orientada a eventos y basado en el motor V8 de Google. Fue creado con el enfoque de ser útil en la creación de programas de red altamente escalables, como por ejemplo, servidores web.

Concurrencia

Node.js funciona con un modelo de evaluación de un único hilo de ejecución, usando entradas y salidas asíncronas las cuales pueden ejecutarse concurrentemente en un número de hasta cientos de miles sin incurrir en costos asociados al cambio de contexto. Este diseño de compartir un único hilo de ejecución entre todas las solicitudes atiende a necesidades de aplicaciones altamente concurrentes, en el que toda operación que realice entradas y salidas debe tener una función callback. Un inconveniente de este enfoque de único hilo de ejecución es que Node.js requiere de módulos adicionales como *cluster* para escalar la aplicación con el número de núcleos de procesamiento de la máquina en la que se ejecuta.

V8

V8 es el entorno de ejecución para JavaScript creado para Google Chrome. Es software libre desde 2008, está escrito en C++ y compila el código fuente JavaScript en código de máquina en lugar de interpretarlo en tiempo real.

Node.js contiene libuv para manejar eventos asíncronos. Libuv es una capa de abstracción de funcionalidades de redes y sistemas de archivo en sistemas Windows y sistemas basados en POSIX como Linux, Mac OS X y Unix.

El cuerpo de operaciones de base de Node.js está escrito en JavaScript con métodos de soporte escritos en C++.

Módulos

Node.js incorpora varios "módulos básicos" compilados en el propio binario, como por ejemplo el módulo de red, que proporciona una capa para programación de red asíncrona y otros módulos fundamentales, como por ejemplo Path, FileSystem, Buffer, Timers y el de propósito más general Stream. Es posible utilizar módulos desarrollados por terceros, ya sea como archivos ".node" precompilados, o como archivos en javascript plano. Los módulos Javascript se implementan siguiendo la especificación CommonJS para módulos, utilizando una variable de exportación para dar a estos scripts acceso a funciones y variables implementadas por los módulos.

Los módulos de terceros pueden extender node.js o añadir un nivel de abstracción, implementando varias utilidades middleware para utilizar en aplicaciones web, como por ejemplo los frameworks connect y express. Pese a que los módulos pueden instalarse como archivos simples, normalmente se instalan utilizando el Node Package Manager (npm) que nos facilitará la compilación, instalación y actualización de módulos así como la gestión de las dependencias.

³² <https://nodejs.org/en/docs/>

Además, los módulos que no se instalen en el directorio por defecto de módulos de Node necesitarán la utilización de una ruta relativa para poder encontrarlos.

Desarrollo homogéneo entre cliente y servidor

Node.js puede ser combinado con una base de datos documental (por ejemplo, MongoDB o CouchDB) y JSON lo que permite desarrollar en un entorno de desarrollo JavaScript unificado. Con la adaptación de los patrones para desarrollo del lado del servidor tales como MVC y sus variantes MVP, MVVM, etc. Node.js facilita la reutilización de código del mismo modelo de interfaz entre el lado del cliente y el lado del servidor.

Bucle de eventos

Node.js se registra con el sistema operativo y cada vez que un cliente establece una conexión se ejecuta un callback. Dentro del entorno de ejecución de Node.js, cada conexión recibe una pequeña asignación de espacio de memoria dinámico, sin tener que crear un hilo de ejecución. A diferencia de otros servidores dirigidos por eventos, el bucle de gestión de eventos de Node.js no es llamado explícitamente sino que se activa al final de cada ejecución de una función callback. El bucle de gestión de eventos se termina cuando ya no quedan eventos por atender.

Anexo K: Ventajas de usar Debian³³

Estabilidad

Existen muchos casos de máquinas que trabajan durante más de un año seguido sin reiniciarse. De la misma forma, hay equipos que tan sólo son reiniciados debido a un fallo en el suministro de corriente o a una actualización del hardware. Compare esto con otros sistemas que se colapsan varias veces al día.

Rápido y ligero en memoria

Otros sistemas operativos pueden ser rápidos en una o dos áreas, pero, estando basado en GNU/Linux o GNU/kFreeBSD, Debian es ligero y humilde. El software para Windows se ejecuta bajo GNU/Linux usando un emulador a veces más rápido que en su ambiente original.

Los controladores para la mayoría del hardware están escritos por usuarios de GNU/Linux / GNU/kFreeBSD, no por el fabricante.

Mientras que esto puede significar retrasos antes de que el nuevo hardware sea soportado y la no existencia de soporte para algún hardware, permite que continúe el soporte mucho después de que el fabricante haya detenido su producción o haya quebrado. La experiencia ha demostrado que los controladores de fuentes abiertas son usualmente mejores que los controladores propietarios.

Buena seguridad del sistema

Debian y la comunidad del software libre son muy sensibles a asegurarse de que las correcciones de problemas de seguridad entren en la distribución rápidamente. Normalmente, los paquetes arreglados se hacen disponibles a los pocos días. La disponibilidad del código fuente permite que la seguridad en Debian se evalúe de forma abierta, lo que evita que se implementen modelos de seguridad pobres. Además, la mayoría de los proyectos de software libre tienen sistemas de revisión por terceras partes, que, como primera medida, evitan que se introduzcan en el sistema problemas de seguridad potenciales.

Software de seguridad

Muchos desconocen que cualquier cosa enviada por la red puede ser leída por cualquier máquina entre usted y el receptor. Debian tiene paquetes del famoso software GPG (y PGP) que permite enviar correo entre usuarios preservando su privacidad. Además, ssh permite crear conexiones seguras a otras máquinas que tengan ssh instalado.

Soporte incomparable

El correo enviado a las listas de correo frecuentemente obtiene respuesta en quince minutos (o menos), gratuitamente, y por las personas que lo desarrollaron. Compare esto al típico soporte telefónico: horas gastadas en el teléfono, pagando dinero, sólo para tener a alguien que no conoce el sistema lo suficientemente bien como para entender su pregunta.

³³ https://www.debian.org/intro/why_debian.es.html

Increíble cantidad de software

Debian viene con más de 51000 elementos de software diferentes. Cada bit de éstos es libre. Si tiene software propietario que corre bajo GNU/Linux o GNU/kFreeBSD, puede usarlo (de hecho, puede que incluso exista un instalador en Debian que automáticamente instale y configure todo por usted).

Paquetes bien integrados

Debian sobrepasa a todas las otras distribuciones en lo bien integrados que están sus paquetes. Como todo el software lo empaqueta un grupo coherente, no sólo puede encontrar todos los paquetes en un mismo sitio sino que puede estar seguro de que hemos eliminado todos los problemas al respecto de complejas dependencias. Aunque creemos que el formato deb tiene algunas ventajas sobre el rpm, es la integración entre paquetes lo que hace a un sistema Debian más robusto.

Código fuente

Si usted es un desarrollador de software, apreciará el hecho de que haya cientos de herramientas y lenguajes de desarrollo, además de millones de líneas de código fuente en el sistema base. Todo el software en la distribución principal es conforme al criterio de las Directrices de Software Libre de Debian (DFSG). Esto significa que usted puede usar libremente este código para estudiarlo o para incorporarlo a un nuevo proyecto de software libre. También hay una buena cantidad de herramientas y código apropiado para el uso en proyectos propietarios.

Actualizaciones fáciles

Actualizarse a una nueva versión de Debian es muy fácil gracias a nuestro sistema de empaquetamiento. Sólo tiene que ejecutar `apt-get update`; `apt-get dist-upgrade` (o `aptitude update`; `aptitude dist-upgrade`, según la versión) y puede actualizarse desde un CD en cuestión de minutos o configurar `apt` para que utilice alguno de los trescientos espejos de Debian y actualizarse desde la red.

Múltiples arquitecturas y kernels

Actualmente Debian soporta un impresionante número de arquitecturas CPU: alpha, amd64, armel, hppa, i386, ia64, mips, mipsel, powerpc, s390, y sparc. También corre con los kernels GNU Hurd y FreeBSD además de Linux, y con la utilidad de bootstrap es difícil que encuentre un dispositivo que no pueda correr Debian.

Sistema de seguimiento de errores

El sistema de seguimiento de errores de Debian es público. No intentamos esconder la realidad de que el software no siempre trabaja de la manera que los usuarios desean. Aconsejamos a los usuarios que envíen informes de errores y serán notificados de cuándo y cómo el error ha sido solucionado. Este sistema permite que Debian responda a los problemas rápida y honestamente.

Apéndices

Apéndice A: Instrumentos de recolección de datos.

Entrevista.

Minutas de reuniones durante el proceso de levantamiento de requerimientos.

Reunión Preliminar

Información general	
Fecha: 27/02/2017	Hora: 5:00pm
Lugar: F.I.A.	Moderador: ing. Edgar Peña
Proyecto: SICEVED	Objetivos: Determinar la necesidad del sistema informático.

Participantes		
#	Nombre y Apellido	Rol
1	Oscar Raúl Pleitez Trujillo	Alumno
2	Castellanos, Edgar	Docente
3	Stefhanie Jeszerelita Rizo Rodriguez	Alumno
4	Peña Figueroa, Edgar Armando	Dueño del proyecto

Síntesis de temas tratados		
#	Tema	Situación/Pasos a seguir
1	Planteamiento de la problemática	<ul style="list-style-type: none">• Se habla de la necesidad de automatizar el proceso de recolección de datos de edificios dañados.• Existen iniciativas como VISUS que han construido aplicaciones informáticas para dicho fin.• En el país existen instituciones como ASIA y CASALCO que poseen formularios impresos con los cuales se recoge la información.• Existen también dos trabajos de graduación de la escuela de ingeniería civil trabajando en una metodología de evaluación post y pre sísmica.• Se plantea un sistema prototipo con la capacidad de soportar el ingreso de mapas con medidas catastrales y el ingreso de evaluaciones de edificios mediante formularios. Además, dicho sistema debe mostrar un mapa con los edificios evaluados y su nivel de daño.
2	Involucrados en el proyecto	<ul style="list-style-type: none">• En principio se pretende que la Escuela de Civil Junto al Vicerrector sean los encargados el sistema. y el Ing. Edgar Peña sea el dueño del proyecto.
3	Manejo de información de evaluación de edificios	<ul style="list-style-type: none">• Por el momento la recolección de datos no es tabulada y solo se generan los formularios llenados por cada voluntario sin ser procesados.

Notas

- Se acordó que ing. Peña enviaría trabajos de graduación para tomarlos de base. y también los formularios de ASIA para tomar de referencia.

Reunión #1

Información general

Fecha: 28/03/2017	Hora: 5:30pm
Lugar: F.I.A.	Moderador: ing. Edgar Peña
Proyecto: SICEVED	Objetivos: Conocer los requerimientos del sistema.

Participantes

#	Nombre y Apellido	Rol
1	Cruz Cordero, José Giovanni	Estudiante
2	Peña Figueroa, Edgar Armando	Dueño del proyecto
3	Pleitez Trujillo, Oscar Raúl	Estudiante
4	Rizo Rodríguez, Stefhanie Jeszerelita	Estudiante

Síntesis de temas tratados

#	Tema	Situación/Pasos a seguir
1	Funcionalidades del sistema	<ul style="list-style-type: none">• Se expresa que es necesaria una app móvil por ser recolección de datos en campo.• También se menciona que tiene que funcionar sin internet por si los servicios se ven interrumpidos debidos al sismo.• Debe haber un mapa que muestre las edificaciones que representen riesgos a la población.
2	Formularios	<ul style="list-style-type: none">• Se presenta el formulario enviado por Ing. Peña y se hacen observaciones sobre el ingreso de esquemas de edificios.• Se acuerda presentar una opción donde los usuarios no tengan que dibujar sobre la tablet todo el esquema sino más bien facilitar dicho proceso de ingreso de información.• Se acuerda separar el formulario en secciones para no mostrar todas las preguntas, sino que se muestren según las respuestas que el usuario vaya ingresando.

Notas

- **Se acordó que ing. Peña enviaría trabajos de graduación para tomarlos de base. y también los formularios de ASIA para tomar de referencia.**

Reunión #2

Información general

Fecha:25/04/2017	Hora: 5:00pm
Lugar: C.C. San Luis	Moderador: ing. Edgar Peña
Proyecto: SICEVED	Objetivos: Determinar los diferentes tipos de evaluación a edificios.

Participantes		
#	Nombre y Apellido	Rol
1	Oscar Raúl Pleitez Trujillo	Estudiante
2	Cruz Cordero, José Giovanni	Estudiante
3	Stefhanie Jeszerelita Rizo Rodriguez	Estudiante
4	Peña Figueroa, Edgar Armando	Dueño del proyecto

Síntesis de temas tratados		
#	Tema	Situación/Pasos a seguir
1	Formularios de evaluación	<ul style="list-style-type: none"> • Se muestra un formulario que no solo implica una evaluación estructural sino también de daños y costos. • Se acuerda que se debe permitir en el sistema el ingreso de futuros formularios que evalúan diferentes aspectos del edificio no solo el estructural.

Cuestionario.

Título: SISTEMA DE EVALUACIÓN POST-SISMO

Objetivo: Recopilar información sobre la metodología actual utilizada para la realización de la evaluación de edificios en caso de sismos.

Indicaciones: Conteste las siguientes preguntas de la manera correspondiente.

Preguntas:

1- ¿Posee alguna especialización en el área estructural?

Si _____

No _____

Si su respuesta fue positiva, pase a la pregunta 2, si no pase a la pregunta 3

2- Escriba la especialización que posee:

3- ¿Ha participado en evaluaciones de edificios posterior a un sismo u otro tipo de desastre natural?

Si _____

No _____

Si su respuesta fue positiva pase a la pregunta 4, si no pase a la pregunta 11

4- Describa brevemente el proceso realizado para la evaluación de edificios:

5- ¿Qué instrumento/s (formulario/s) ha utilizado para la inspección de edificios?

6- ¿Quién facilitó dichos instrumentos?

7- Si ha participado en más de una evaluación ¿Variaron los instrumentos utilizados en cada evaluación?

Si _____

No _____

Si su respuesta fue positiva pase a la pregunta 8, si no pase a la pregunta 9

8- ¿A qué iba orientada dicha variación?

Al objetivo de la evaluación _____

Al contenido de la evaluación _____

Otro _____ Especifique _____

9- ¿A la hora de realizar una evaluación, que elementos considera prioritarios en la inspección preliminar?

10- ¿Qué deficiencias considera que tienen los formularios utilizados en la evaluación de edificios?

11- ¿Considera necesario estandarizar la evaluación de edificios para evitar sobreestimación o subestimación del riesgo?

Si _____

No _____

12- ¿Considera de importancia el establecer prioridad de inspección a algunos edificios?

Si _____

No _____

Si su respuesta fue positiva pase a la pregunta 13, si no pase a la pregunta 14

13- ¿Cuáles deben ser prioritarios?

Escuelas _____

Centros Comerciales _____

Alcaldías _____

Edificios de gobierno _____

Estación de bomberos _____

Hospitales _____

Otras _____ Especifique _____

14- ¿Cómo se organizan las jornadas de evaluación ante una emergencia?

Por convocatoria _____

Por voluntariado _____

Por afinidad _____

Otro _____ Especifique _____

15- ¿Considera eficiente la organización para las evaluaciones de edificios en caso de emergencia?

Si _____

No _____

16- ¿Considera necesaria la realización de evaluaciones periódicas de las edificaciones?

Si _____

No _____

17- ¿Con qué frecuencia deberían realizarse las evaluaciones a edificios?

18- ¿Considera útil que se almacene un histórico de evaluaciones anteriores de las edificaciones?

Si _____

No _____

Si su respuesta fue positiva pase a la pregunta 19, si no pase a la pregunta 20

19- ¿Qué utilidad tendría dicho histórico?

20- ¿Qué dispositivos utiliza como ayuda para tomar medidas, distancias, ángulos, ubicación, etc.?

21- ¿Conoce alguna aplicación o software para la evaluación de edificios?

Si _____

No _____

Si su respuesta fue positiva pase a la pregunta 22, si no pase a la pregunta 27

22- Mencione las aplicaciones que conoce:

23- ¿Ha utilizado alguna de estas aplicaciones para evaluación de edificios?

Si _____

No _____

Si su respuesta fue positiva pase a la pregunta 24, si no pase a la pregunta 27

24- ¿Cuáles utilizas?

25- ¿Qué ventajas observa de este tipo de aplicaciones?

26- ¿Qué desventajas observa de este tipo de aplicaciones?

27- ¿Qué apoyo visual sería útil mientras realiza una evaluación en un área extendida de terreno?

28- ¿Qué uso podría tener la información obtenida en las evaluaciones?

29- ¿Conoce alguna institución que se encargue de revisar y tomar medidas de acuerdo a los resultados obtenidos en las evaluaciones de edificios?

Si _____

No _____

Si su respuesta fue positiva pase a la pregunta 30, si no pase a la pregunta 31

30- Mencione las instituciones

31- ¿Considera útil que se almacene la información recolectada en las jornadas de evaluación?

Sí _____

No _____

32- ¿Quién debería almacenar la información resultante de las evaluaciones realizadas a los edificios?

33- ¿Conoce alguna entidad que posea la información recolectada en las evaluaciones realizadas hasta la fecha?

Si _____

No _____

Si su respuesta fue positiva pase a la pregunta 34, si no pase a la pregunta 35

34- Mencione la o las instituciones que poseen dicha información

35- ¿Por qué cree que no se han centralizado los datos de las inspecciones realizadas?

36- ¿Cuál es el procedimiento en caso un edificio sea considerado inhabitable?

Informar a los dueños o residentes _____

Evacuación inmediata obligatoria _____

Clausura obligatoria _____

Clausura y demolición _____

Otra _____ Especifique _____

37- Si existe un sitio web para mostrar la información recolectada en las evaluaciones ¿Cómo cree conveniente que se despliegue dicha información?

Fotografías _____

Mapas 2D _____

Mapas 3D _____

Puntos de referencia _____

38- ¿Quién debería tener acceso a dicha información?

Análisis de documentos.

El análisis de documentos. Es la técnica de investigación donde los analistas de sistemas y diseñadores deben tratar de encontrar la información necesaria para comenzar las investigaciones. En los documentos se puede encontrar la historia de la temática a tratar, estadísticas, iniciativas anteriores o vigentes sobre la temática.

Para el presente trabajo se han tomado como base los trabajos de graduación de la escuela de ingeniería civil que a continuación se detallan.

Evaluación Pre-Sismo

Título: "PROPUESTA METODOLÓGICA PARA LA CREACIÓN DE UN SISTEMA DE GESTIÓN DE INFORMACIÓN DE ESTUDIOS DE VULNERABILIDAD ESTRUCTURAL". Por: JOSÉ ROBERTO ALVARADO CERÓN, OSCAR ORLANDO JAVIER REYES. JUNIO 2017.

Evaluación Post Sismo

Título: "PROPUESTA DE ÍNDICE DE DAÑO ESTRUCTURAL PARA CUANTIFICAR EL NIVEL DE AFECTACIÓN EN ESTRUCTURAS EN ZONAS URBANAS DESPUÉS DE UN SISMO" Por: MARISSA SCARLETT ALVARENGA CARDOZA, CARLOS ENRIQUE GARCÍA RAMÍREZ, KARINA LISETTE PÉREZ DE LÉON. JUNIO 2017.

De los cuales se ha tomado como base lo siguiente:

- Es necesaria la parametrización de toda la evaluación ya que al permitir el ingreso de nuevos formularios al sistema este deberá permitir ingresar la forma de evaluación de las estructuras es decir los cálculos necesarios para determinar el estado del edificio.
- Es necesario llevar un registro histórico de todas las evaluaciones, esto debido a que se pretende tener un inventario pre-sismo de los edificios para posteriormente agregar evaluaciones estructurales.

Se consideraron también de gran aporte los formularios de evaluación de edificios:

- ASIA
- CASALCO³⁴
- IEE FOMENTO ESPAÑA³⁵
- Asociación Colombiana de Ingeniería Sísmica – AIS³⁶

Estos documentos nos han ayudado a determinar los siguientes:

- Es necesario separar el sistema en módulos que permitan el ingreso de múltiples formularios para que un edificio pueda ser evaluado de diferentes puntos de vista y con diferentes criterios.
- Se debe permitir la configuración lógica de cálculos para determinar el estado de un edificio.

³⁴ Manual de Evaluación Post - Sísmica de Edificaciones de El Salvador Abril 2008
http://www.evivienda.gob.sv/Emergencias/Descargas/MaterialCER/Manual/PARTE_1.pdf

³⁵ Ministerio de Fomento. España Informe de evaluación del edificio
<https://iee.fomento.gob.es/>

³⁶ Gestión del Riesgo Manizales- Colombia. MANUAL DE EVALUACIÓN DE DAÑOS
http://www.gestiondelriesgomanzales.com/index.php?option=com_content&view=article&id=38%3Amanu-al-de-evaluacion-de-danos&catid=41%3Amanejo-de-desastres&Itemid=198

Apéndice B: Plan de capacitación para administradores.

Alcance.

El presente plan de capacitación es de aplicación para los administradores de la plataforma SICEVED.

Objetivos.

Objetivo general.

Preparar a los administradores de sistema para instalar, configurar y administrar la plataforma SICEVED.

Objetivos específicos.

- Proporcionar orientación en la instalación y configuración de la plataforma SICEVED.
- Proveer métodos para la realización de respaldos de información.
- Ayudar en la preparación del personal que administrará la plataforma.
- Mostrar métodos para la realización de consultas complejas sobre la data recolectada.

Recursos.

Humanos.

Participantes: Futuros administradores de la plataforma SICEVED.

Facilitador: Experto familiarizado con el funcionamiento interno y uso de la plataforma SICEVED.

Materiales.

Infraestructura: Centro de computo proporcionado por la institución capacitadora, con servicios básicos y acceso a internet.

Mobiliario, equipo y otros: Estaciones de trabajo con terminal para conectarse a servidores virtuales, material para anotaciones, pantalla de proyección, proyector.

Documentos técnicos:

- Manual de instalación y configuración de la plataforma.
- Manual de usuario de la aplicación web.
- Manual de usuario de la aplicación móvil.
- Documentación oficial MongoDB Compass.

Contenido de la capacitación.

Contenido	Tiempo necesario (Minutos)
Instalación de Web API	
Instalación y configuración de MongoDB	60
Respaldos de seguridad	30
Preparación de entorno de ejecución	30
Ejecución de Web API en producción	20
Instalación de Aplicación Web	
Preparación de entorno de ejecución	30
Ejecución de Aplicación Web en producción	20
Aplicación web	
Descripción de interfaces	20
Administración de catálogos	60
Creación de módulos de inspección	180
Utilización del mapa	40
Aplicación móvil	
Descripción de interfaces	20
Tipos de entrada	60
Administración de catálogos	30
Utilización del mapa	20
Realización de inspecciones	60
Generación de consultas complejas sobre data recolectada.	
MongoDB Compass	40
Introducción a aggregation framework	120
Creación de pipelines	180
Total horas	17

Apéndice C: Plan de capacitación para evaluadores.

Alcance.

El presente plan de capacitación es de aplicación para los evaluadores que hará uso de la plataforma SICEVED.

Objetivos.

Objetivo general.

Preparar a los evaluadores para el correcto uso de la plataforma SICEVED.

Objetivos específicos.

- Ayudar a los evaluadores a familiarizarse con las interfaces que provee la plataforma SICEVED.
- Mostrar el uso de la aplicación móvil en el campo.
- Proporcionar orientación sobre el funcionamiento de subida de recursos de apoyo a las inspecciones.
- Mostrar el uso de la aplicación web, para visualizar inspecciones realizadas y en desarrollo.

Recursos.

Humanos.

Participantes: Evaluadores que harán uso de la plataforma SICEVED.

Facilitador: Experto familiarizado con el uso de la plataforma SICEVED.

Materiales.

Infraestructura: Centro de computo proporcionado por la institución capacitadora, con servicios básicos y acceso a internet. Edificaciones para simular evaluaciones de campo.

Mobiliario, equipo y otros: Estaciones de trabajo con computadoras para acceder a la aplicación web, dispositivos móviles con Android 5+, material para anotaciones, pantalla de proyección, proyector.

Documentos técnicos:

- Manual de usuario de la aplicación web.
- Manual de usuario de la aplicación móvil.

Contenido de la capacitación.

Contenido	Tiempo necesario (Minutos)
Uso de la Aplicación web	
Descripción de interfaces	20
Utilización del mapa	40
Uso de la Aplicación móvil	
Descripción de interfaces	20
Tipos de entrada	60
Utilización del mapa	20
Uso de la plataforma en campo	
Sistema de mensajería	30
Realización de inspecciones grupales	180
Manejo de colas de subida	20
Visualización de inspecciones en tiempo real	120
Total minutos	510
Total horas	8.5