

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



**HERRAMIENTA PARA EL DESARROLLO ASISTIDO DE  
DIAGRAMAS DE FLUJO**

PRESENTADO POR:

**JEANNETTE MELLANY AGUILAR ZEPEDA**

**CHRISTIAN ALEXANDER MARTÍNEZ ARTEAGA**

**ELVIS MAURICIO RAMÍREZ**

**MIGUEL ISAAC SÁNCHEZ RAMOS**

PARA OPTAR AL TÍTULO DE:

**INGENIERO(A) DE SISTEMAS INFORMÁTICOS**

CIUDAD UNIVERSITARIA, FEBRERO DE 2010

**UNIVERSIDAD DE EL SALVADOR**

**RECTOR :**

**MSc. RUFINO ANTONIO QUEZADA SÁNCHEZ**

**SECRETARIO GENERAL :**

**LIC. DOUGLAS VLADIMIR ALFARO CHÁVEZ**

**FACULTAD DE INGENIERIA Y ARQUITECTURA**

**DECANO :**

**ING. MARIO ROBERTO NIETO LOVO**

**SECRETARIO :**

**ING. OSCAR EDUARDO MARROQUÍN HERNÁNDEZ**

**ESCUELA DE INGENIERIA DE SISTEMAS INFORMÁTICOS**

**DIRECTOR :**

**ING. CARLOS ERNESTO GARCÍA**

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

Trabajo de Graduación previo a la opción al Grado de:  
**INGENIERO(A) DE SISTEMAS INFORMÁTICOS**

Título

:

**HERRAMIENTA PARA EL DESARROLLO ASISTIDO DE  
DIAGRAMAS DE FLUJO**

Presentado por

:

**JEANNETTE MELLANY AGUILAR ZEPEDA  
CHRISTIAN ALEXANDER MARTÍNEZ ARTEAGA  
ELVIS MAURICIO RAMÍREZ  
MIGUEL ISAAC SÁNCHEZ RAMOS**

Trabajo de Graduación Aprobado por:

Docentes Directores

:

**ING. EDGAR WILLIAM CASTELLANOS SÁNCHEZ**

San Salvador, Febrero de 2010.

Trabajo de Graduación Aprobado por:

Docentes Directores :

**ING. EDGAR WILLIAM CASTELLANOS SÁNCHEZ**

## **DEDICATORIA**

### **A DIOS NUESTRO SEÑOR**

Te dedico este trabajo de graduación ya que sin la fortaleza y sabiduría que de Tí viene, no sería capaz de lograr esta meta, que sea mi Señor para tu Santa Gloria.

### **A MIS PADRES**

Agradezco a mi mami porque siempre estuvo a mi lado durante toda mi vida y que sin sus consejos y valiosa ayuda no estaría donde estoy ahora. A mi papi, que siempre ha sido un gran apoyo y que me ha enseñado siempre cual es el mejor camino a seguir con su ejemplo y el amor que siempre me ha dado.

### **A MIS HERMANOS**

Les agradezco mucho por su infinito apoyo, por que han corrido a mi lado no solamente durante el desarrollo de la carrera, sino en estos años de mi vida... Gracias a mi hermana, que siempre ha estado ahí para apoyarme y brindarme siempre su cariño; por todos esos favores que a última hora le solicitaba pero que siempre estaba dispuesta a ayudarme.

### **A MI AMOR**

A ti, Christian, porque sin tu apoyo y tú amor probablemente no estaría finalizando este trabajo de graduación, porque siempre me ayudaste a aclarar el camino y a no tomar decisiones sin pensarlas, te agradezco mucho por escucharme y guiarme cuando no tenía la razón, porque me ayudaste a guardar la calma y seguir luchando.....TE AMO.

### **A LA FAMILIA DE CHRISTIAN**

A toda esa linda familia, Doña Carmencita, Memito y Don Guillermo, a todos los tíos y primos porque siempre nos apoyaron y nos animaron a seguir adelante con este trabajo, les agradezco mucho sus palabras y su cariño.

### **A NUESTRO ASESOR**

Gracias, infinitas gracias, Ing. Castellanos, por que sin su apoyo y sin sus palabras de aliento para slir adelante no estaríamos finalizando este trabajo de graduación, nuestro Señor Jesucristo derramará muchas bendiciones sobre usted y sobre toda su familia por la gran persona que es...

### **A MIS QUERIDOS AMIGOS Y AMIGAS**

Gracias a todos mis amigos y amigas, los que están cerca y lejos, por que siempre tuvieron ese tiempo de escucharme o me ayudaron para desahogarme en los momentos más difíciles del desarrollo de este proyecto.

### **A MIS COMPAÑEROS DE TRABAJO DE GRADUACIÓN**

Por todos esos buenos y no tan buenos momentos que compartimos a lo largo del desarrollo de este trabajo, ya que sin ustedes no sería posible finalizar este objetivo.

A todas las personas que han participado de manera directa e indirecta en el desarrollo de este trabajo de graduación por que en alguna medida, contribuyeron a que se llevara a cabo....Les agradezco y les dedico este trabajo a todos, por aguantarme, por tenerme paciencia y apoyarme siempre...LOS QUIERO MUCHO.

*Mellany Aguilar*

## **Dedicatoria**

*Primeramente le agradezco a Dios por haberme dado el milagro de vivir, por estar siempre cuidándome y por darme esa fortaleza de seguir adelante, cada vez que se lo pedí el nos ayudo dándome consuelo y sabiduría.*

*Además Diosito se encargo de darme una madre excepcional la cual desde el primer día que nací ha creído en mi incondicionalmente y me ha dado ese calor maternal importante para seguir adelante, gracias a cada esfuerzo y sacrificio, gracias a ese gran amor que siempre me ha dado María del Carmen Arteaga Álvarez es que yo soy lo que soy. La amo mucha mamá y gracias a ti y a Dios yo puedo cumplir una meta tan importante. Eres el mejor ejemplo a seguir en mi vida ya que siempre me diste lo mejor de ti para que yo estuviera bien.*

*Le agradezco a Don Guillermo que ha sido como un padre el cual me ha apoyado en mi carrera y también a Don Francisco ya que creyó mucho en mí.*

*Gracias a mi familia, a mis tíos Leonel, Jesús, Armando y mis tías Margarita, Estela, Gladys y Nora a mi abuelo José Arteaga porque siempre me dieron ese consejo o esas palabras de aliento, siempre estuvieron pendiente de mí y siempre oraron por mis triunfos.*

*También a ti Mellany Aguilar, te agradezco mucho mi amor, por estar a mi lado en los momentos más difíciles, por ser mi apoyo, mi amiga, por darme un segundo punto de vista el cual siempre me ayudo a superar cualquier dificultad. La amo mucho mi amorcito lindo.*

*Gracias a mi hermano Memito y mis primos Tito y Katherine ya que por ellos he tratado de dar lo mejor de mí para que puedan ver en mí un ejemplo a seguir.*

*Gracias a los padres y hermanos de mi novia, Lilian de Aguilar, René Aguilar, Lisseth y Edwin por sus palabras de aliento, por su comprensión y por su cariño que me brindaron a lo largo de este trabajo de graduación, por sus oraciones y por el interés que mostraron cada día.*

*A todos mis amigos que me escucharon, y me ayudaron cuando más lo necesite, aquellos que me dieron un consejo y otros que me dieron esperanzas para seguir adelante.*

*Le agradezco a mi asesor el Ing. William Castellanos que más que un asesor fue un amigo el cual supo guiarnos por el camino del éxito, también muchas gracias al Ing. Balmore Ortiz ya que siempre nos dio palabras de aliento y nunca dudaron de nuestra capacidad.*

*No me puedo olvidar de mi equipo de trabajo gracias Mell, Elvis, Miguel ya que dieron lo mejor de ustedes, gracias por su esfuerzo es que todos estamos aquí, esto no lo ganamos 1 o 3 lo ganamos todos y si triunfamos es porque todos dimos lo mejor de nosotros. Gracias muchachos saben que más que un equipo para mí son muy buenos amigos.*

*En general gracias a todas aquellas personas que de alguna u otra manera contribuyeron a este importante éxito que primero Dios sea el primero de muchos que deseo lograr.*

***Christian Martínez***

## DEDICATORIA

### A DIOS TODOPODEROSO.

Por haber dado la oportunidad de alcanzar este logro y darme la sabiduría necesaria para poder alcanzarlo.

### A MIS PADRES.

Por darme la oportunidad de estudiar en la Universidad de El Salvador teniendo siempre en mente un mejor futuro profesional para mí, por darme su apoyo incondicional tanto económica como emocionalmente en los buenos y en los malos momentos a lo largo de toda mi formación profesional. Los quiero mucho.

### A MI ABUELO (Q.D.D.G.)

Por haberme dado un enorme ejemplo de persona durante mi infancia y adolescencia. Gracias por haberme enseñado a ser como soy y darme siempre tu confianza y apoyo.

### A MI FUTURA ESPOSA.

Por estar siempre a mi lado como mi apoyo incondicional desde que te conocí, en las buenas y en las malas, por hacerme ver muchas veces que estaba cometiendo errores y ayudarme a salir de ellos y por querer siempre lo mejor para mí. Hoy soy yo mañana serás tú quien alcance sus metas y ahí estaré contigo como tu estuviste siempre conmigo.

### A MIS HERMANOS.

Por estar siempre conmigo en todo momento directa o indirectamente a lo largo de toda mi carrera. Siempre esfuércense para alcanzar sus sueños y sus metas.

### A MIS COMPAÑEROS DE TRABAJO DE GRADUACION.

Felicitarlos por haber logrado este objetivo que tanto cuesta, por haber tenido paciencia durante todo este último paso de la carrera, por no desfallecer en momentos duros durante el trabajo de graduación y por haber sido parte este logro. Gracias por todo.

### A MIS COMPAÑEROS Y AMIGOS

Que durante toda la carrera estuvieron siempre conmigo, teniendo siempre en mente lograr el objetivo que hoy estamos logrando nosotros. A ellos los que ya finalizaron este largo camino, Felicidades; y a los que aún faltan no desfallezcan por nada.

A MIS FAMILIARES.

Por todo el apoyo que me dieron de una u otra manera a lo largo de todo este tiempo. Gracias especiales a mi tía y mis primas por el apoyo brindado durante el tiempo que estuve con ellas.

A todos infinitas Gracias por siempre!!!!

Elvis Mauricio Ramírez

## Agradecimientos

Primeramente le doy las gracias a Dios todo poderoso porque por su misericordia me ha permitido llegar al final de una meta más en esta vida y la honra y la gloria sea para él y su hijo Jesucristo.

A mi grupo de trabajo de graduación, que a pesar de los problemas que surgieron como grupo, pudimos salir adelante con este proyecto sin importar las discusiones y enojos que en un momento pudieron resaltar, pero la gran amistad que nos une fue la encargada de que el grupo se mantuviera de principio a fin, sin importar el desgaste físico y mental que un trabajo de graduación puede generar.

A todos mis amigos que siempre tuvieron la fe de que pudiera concluir con esta meta, sus ánimos apoyo y cariño han sido de un valor sin igual y personas como ustedes son difíciles de encontrar en la vida y agradezco a Dios por cada una de esas amistades que hasta el día de hoy sigo manteniendo y espero mantener por muchos años mas (incluyendo a los mompirris y la Cipotada F.C.).

A todas aquellas tardes y noches “futeboleras” en el hoyo, en periodismo, en el complejo deportivo, en las canchas aledañas a la UES, en las canchas de la gambeta etc. en las cuales botábamos el estrés generado por la carga académica, a todos aquellas asistencias, goles, paredes, sombreritos, tunelitos, patadas, trompones etc. Ahhh! cuantos recuerdos sin igual jajajajaja.

A todos mis hermanos en Cristo Jesús que siempre me han apoyado en todo momento, ustedes valen la sangre de nuestro salvador, gracias por estar para mi en esos momentos de desesperación y desaliento... A cada uno de ustedes les pertenece un pedacito de mi corazón, los amo mucho.

Al asesor de mi trabajo de graduación que Dios puso en nuestro camino, sin su ayuda y confianza no hubiéramos concluido con este proyecto, Dios le bendiga a usted y su familia y lo colme de ricas y abundantes bendiciones.

A la familia que Dios me ha regalado, a cada uno de mis hermanos que en una u otra forma han sacrificado algo para que yo haya alcanzado este punto en mi carrera, contentos o enojados a todos los amo por igual. Al padre que Dios me ha regalado que en todo momento me brinda su ayuda y consejos y uno que otro regaño por haragán y perezoso jajaja lo amo mucho “jefe”.

Por ultimo, pero no menos importante, a esa grandiosa madre que Dios me ha dado, gracias por cada uno de sus consejos, por sus castigos y por su amor, mujeres en el mundo hay muchas pero como tu ninguna madrecita y sin ti no seria ni la mitad de lo que soy, te amo con todo mi corazón.

**Miguel Isaac Sánchez Ramos**

*“Mira que te mando que te esfuerces y seas valiente; no temas ni desmayes, porque Jehová tu Dios estará contigo en dondequiera que vayas” Josué 1:9*

## Índice

Índice .....	ix
Introducción .....	i
Objetivos.....	iii
Objetivo General.....	iii
Objetivos Específicos.....	iii
<b>CAPÍTULO I: INVESTIGACIÓN PRELIMINAR.....</b>	<b>1</b>
<b>1. Antecedentes .....</b>	<b>1</b>
1.1 Información recabada mediante encuestas a docentes de la EISI.....	1
<b>2. Alcances y Limitaciones.....</b>	<b>4</b>
2.1 Alcances .....	4
2.2 Limitaciones .....	4
<b>3. Importancia y Justificación.....</b>	<b>4</b>
3.1 Importancia.....	4
3.2 Justificación.....	7
<b>4. Estudio de Factibilidades.....</b>	<b>9</b>
4.1 Factibilidad Técnica .....	9
4.1.1 Software de desarrollo.....	9
4.1.2. Hardware de desarrollo .....	9
4.1.3. Software de implementación .....	10
4.1.4. Hardware de implementación.....	11
4.1.5. Equipo informático actual de la EISI .....	12
4.1.6. Conclusión de factibilidad técnica .....	12
4.2 Factibilidad Económica .....	13
4.2.1. Costo por el desarrollo del software.....	13
4.2.2. Costo por recurso informático para la implementación de la herramienta .....	13
4.2.3. Beneficios tangibles .....	13
4.2.4. Análisis Costo/Beneficio.....	14
4.2.5. Conclusiones de la factibilidad económica .....	15
4.3. Factibilidad Operativa.....	15

4.3.1.	Aceptación de los usuarios.....	15
4.3.2.	Utilización del Software .....	16
4.3.3.	Conclusión de la factibilidad operativa .....	16
5.	Metodología utilizada .....	17
5.1.	Ciclo de vida de desarrollo.....	17
5.1.1.	Ventajas, Desventajas y utilización del ciclo de vida en espiral. ....	18
5.2.	Herramientas de investigación.....	18
5.2.1.	Técnicas.....	18
5.2.2.	Herramientas y equipo.....	20
5.2.3.	Recursos .....	22
5.3.	Definición de la situación problemática .....	22
5.3.1.	Identificación del problema.....	22
5.3.2.	Planteamiento del problema.....	23
5.4.	Gestión de proyectos.....	23
5.5.	Análisis y Diseño .....	24
5.6.	Codificación.....	25
5.7.	Pruebas .....	26
5.8.	Mantenimiento .....	26
CAPITULO II: ANÁLISIS DE LA HERRAMIENTA.....		27
1.	Situación problemática .....	27
1.1.	Identificación del Problema .....	27
1.1.1.	Descripción de causas principales del problema .....	28
1.2.	Definición del problema .....	33
1.3.	Planteamiento del problema .....	33
1.4.	Enfoque de Sistemas .....	34
1.4.1.	Componentes del Enfoque de sistemas.....	34
1.4.2.	Esquema del Enfoque de sistemas de la situación actual .....	35
1.4.3.	Descripción del Enfoque de Sistemas de la Situación Actual .....	36
2.	Definición de Requerimientos.....	38
2.1.	Requerimientos Funcionales.....	40
2.2.	Requerimientos No Funcionales .....	50

2.3.	Requerimientos de seguridad .....	51
2.4.	Requerimientos legales. ....	52
2.5.	Requerimientos de Desarrollo .....	52
2.5.1.	Software de desarrollo.....	52
2.5.2.	Selección del entorno de desarrollo. ....	54
2.5.3.	Requerimientos mínimos para la instalación del paquete Java SE y NetBeans .....	55
2.6.	Requerimientos de Implementación .....	56
2.6.1.	Software de implementación .....	56
2.6.2.	Hardware de implementación.....	57
2.6.3.	Equipo informático actual de la EISI .....	57
3.	Diagramas de casos de uso .....	58
3.1.	Nomenclatura .....	58
3.1.1	Descripción del Personal Involucrado.....	58
3.1.2	Objetivos de Nivel de Usuario .....	59
3.1.3	Perspectiva del sistema.....	59
3.1.4	Resumen de beneficios .....	60
3.1.5	Resumen de las características del sistema .....	60
3.1.6	Modelo de caso de Uso.....	61
CAPÍTULO III: DISEÑO DE LA HERRAMIENTA .....		63
1.	Diseño de estándares .....	63
1.1	Ventajas de la estandarización.....	63
1.2	Estándares de la aplicación .....	64
1.2.1	Estándares para Generación de pseudocódigo .....	64
1.3	Estándares de programación .....	68
Métodos .....		68
Clases.....		68
Interfaces.....		69
1.4	Estándares de documentación.....	69
1.4.1	Documentación Interna .....	69
2.	Arquitectura de la herramienta .....	74
2.1.	Diagrama Jerárquico de Procesos .....	74

Descripción de módulos .....	74
3. Diagrama de Clases .....	76
3.1. Diagrama de Clases.....	76
4. Diseño de interfaces.....	78
4.1. Interfaz externa .....	78
4.1.1. Aspectos a considerar en cuanto a la interfaz de usuario.....	78
4.1.2. Elementos gráficos a utilizar para el diseño de interfaces .....	79
4.1.3. Diseño de Menú.....	80
4.1.4. Diseño de Mensajes .....	81
<b>CAPITULO IV: PLAN DE IMPLEMENTACIÓN .....</b>	<b>83</b>
1. Planeación .....	83
1.1 Consideraciones Previas a la Etapa de Planeación.....	83
1.1.1 ¿Qué es la Etapa de Planeación? .....	83
1.1.2 ¿Por qué es Conveniente Realizar la Etapa de Planeación?.....	83
1.1.3 ¿Cómo se Realiza la Etapa de Planeación?.....	84
1.2 Método de Implementación de la Aplicación EISI - FlowChart.....	84
1.3 Listado de Actividades .....	85
1.3.1 Preparación .....	85
1.3.2 Recursos a Utilizar.....	87
1.3.3 Aprobar los Resultados de la Planeación .....	98
2. Organización.....	99
2.1 Descripción de Funciones y Perfiles de Puestos.....	99
3. Control.....	105
3.1 Canales de Comunicación .....	105
3.2 Documentos de Control.....	105
3.2.1 Control de Actividades de Preparación e Instalación .....	106
3.2.2 Control de Actividades Generales.....	108
3.2.3 Control de Capacitaciones.....	114
4. Plan de Capacitación .....	116
4.1. Contenido de la Capacitación.....	116

4.1.1.	Duración de la Capacitación .....	120
4.1.2.	Usuarios a Capacitar .....	121
4.1.3.	Entorno para la Capacitación .....	123
4.1.4.	Equipo Tecnológico y Material Didáctico .....	124
4.1.5.	Métodos para Impartir la Capacitación .....	125
4.1.6.	Evaluación del Conocimiento .....	127
Conclusiones.....		129
Recomendaciones.....		130
Bibliografía.....		131
1.1.	Libros. ....	131
1.2.	Páginas Web.....	131
Anexos .....		132
Anexo 1: Presupuesto del proyecto .....		132
Estimación del costo total del proyecto.....		132
Costos directos .....		132
Recurso humano .....		132
Salarios promedios de empresas privadas para programadores en Java .....		132
Estimación de sueldo promedio por hora .....		132
Estimación de duración del proyecto.....		133
Estimación de sueldo promedio por hora del docente director del proyecto .....		133
Recurso tecnológico .....		134
Costos indirectos.....		135
Recursos consumibles .....		135
Recursos de operación .....		136
Estimación de gasto en telefonía celular .....		136
Estimación de gastos de transporte.....		137
Resumen de costos .....		138
Anexo 2: Encuestas realizadas a docentes .....		139
Encuesta a docentes universitarios 1.....		139
Encuesta para docentes univesitarios 2 .....		141

Anexo 3: Figuras estadarizadas de la ANSI.....	148
Anexo 4: Métodos de Conversión de Sistemas .....	149
Método Paralelo. ....	149
Método Directo .....	150
Método Piloto.....	150
Método por Etapas .....	150

## Introducción

En la Escuela de Ingeniería de Sistemas Informáticos (EISI) se imparten las primeras asignaturas acerca de informática y se inicia el estudio de el proceso solucionador de problemas, el cual consiste en una secuencia de pasos lógicos para solventar cualquier situación problemática planteada. Generalmente esta compuesto por 5 pasos que son: planteamiento del problema, análisis del problema (definición de variables), diseño de la solución (construcción del flujograma), construcción de pseudocódigo y construcción de código fuente.

Este proceso se realiza de manera manual para docentes al momento de impartir las clases teóricas o de laboratorio y para estudiantes al momento de recibir clases, realizar actividades evaluadas o prácticas de laboratorio.

El desarrollo del proceso solucionador de problemas consume mucho tiempo cuando crece la complejidad de los ejercicios propuestos, esto se debe a que el docente debe explicar y escribir en la pizarra las soluciones a los ejercicios vistos en clase. A este inconveniente se le suma la falta de recursos didácticos para todos los períodos de clases.

Por otra parte, el desarrollo de tareas relacionadas al proceso solucionador de problemas, consume mucho tiempo a los estudiantes y no le permite conocer si el desarrollo de la misma tiene una validez lógica.

Para solventar esta situación, se ha desarrollado una *Herramienta para el desarrollo asistido de diagramas de flujo EIS-FlowChart* la cual tiene como objetivo contribuir a realizar el proceso solucionador de problemas de manera más eficiente para el proceso de enseñanza aprendizaje.

El contenido de este documento se ha dividido en cuatro capítulos y son los siguientes:

### **Capítulo I: Investigación preliminar**

Muestra los antecedentes que permiten comprender el medio ambiente de la herramienta a desarrollar, la situación actual y todas las áreas involucradas. Además se incluyen los alcances, limitaciones, importancia y justificación del proyecto, así como el desglose del estudio de factibilidades y la metodología a utilizar.

### **Capítulo II: Análisis de la herramienta**

En este capítulo se presentarán todos los elementos que fueron considerados para desarrollar la herramienta. El análisis permite hacer una clasificación e interpretación de hecho,

llevándonos a obtener un diagnóstico de problemas y empleo de la información para recomendar mejoras al proceso actual.

Se mostrarán en un listado de requerimientos funcionales y no funcionales para cada etapa del proceso solucionador de problemas, requerimientos de seguridad y requerimientos legales, validados a medida se avanza en el proceso.

### **Capítulo III: Diseño de la herramienta**

En este capítulo se presentan los estándares utilizados para el desarrollo de la herramienta tanto para la construcción como para la utilización y aplicación de la misma. Es decir, se detallan los estándares de interfaces, documentación y programación. Además se detalla la arquitectura de la herramienta mediante un diagrama jerárquico de procesos y una explicación de cada uno de los módulos que componen la aplicación.

### **Capítulo IV: Plan de implementación**

El objetivo de este documento es ofrecer a los usuarios la descripción de las tareas necesarias, para asegurar la puesta en marcha o puesta en producción de la herramienta EISI-FlowChart.

## Objetivos

### Objetivo General

- ✓ Desarrollar todos los elementos requeridos en las etapas de análisis, diseño y construcción de la *Herramienta para desarrollo asistido de diagramas de flujo* para las fases del proceso solucionador de problemas, así como también la documentación para el uso, mantenimiento y puesta en marcha de la herramienta.

### Objetivos Específicos

- ✓ Realizar un estudio de la situación actual del proceso enseñanza – aprendizaje del proceso solucionador de problemas de la EISI en las fases de *Planteamiento del problema, Análisis del problema, Construcción del flujograma, Ejecución del flujograma, Generación de pseudocódigo y código fuente*.
- ✓ Definir y analizar de forma precisa y clara todos los requerimientos informáticos para las fases del proceso solucionador de problemas.
- ✓ Diseñar las interfaces de cada una de la fases del proceso solucionador de problemas.
- ✓ Construir el software correspondiente a todas las fases del proceso solucionador de problemas para obtener un software 100% funcional.
- ✓ Elaborar la documentación de las funcionalidades de la herramienta EISI-FlowChart como manual de usuario, manual de instalación y manual técnico.
- ✓ Elaborar el plan de implementación y plan de capacitaciones para poner en funcionamiento la herramienta EISI-FlowChart.

## CAPÍTULO I: INVESTIGACIÓN PRELIMINAR

### 1. Antecedentes

#### 1.1 Información recabada mediante encuestas a docentes de la EISI<sup>1</sup>

Estas son todas las etapas que se deben completar para dar por finalizado el proceso solucionador de problemas, el cual es realizado manualmente, principalmente utilizando herramientas ofimáticas (Microsoft Word) para apoyarse en las primeras fases del proceso (formulación del problema, análisis del problema y diseño de la solución) quedando la utilización de herramientas como Microsoft Visio y el compilador de Turbo C 3.0 para desarrollar las etapas de construcción del diagrama de flujo y la codificación respectivamente. En la figura 1.1 se pueden observar las herramientas que se utilizan actualmente para el desarrollo de la temática en Introducción a la Informática.

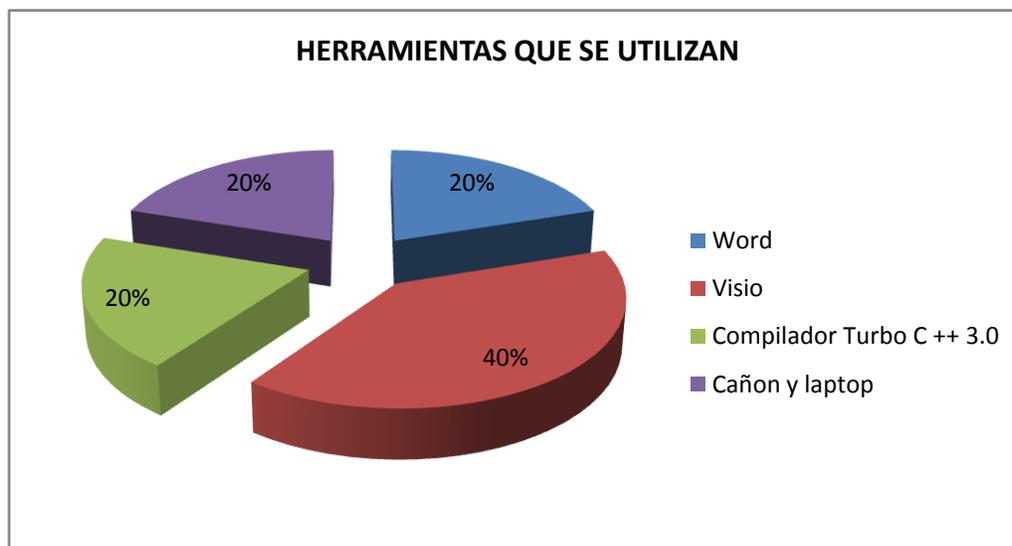


Figura 1.1 – Herramientas que se utilizan para elaborar diagramas de flujo.

El desarrollo del proceso solucionador de problemas se complica debido a que se hace necesario que los estudiantes utilicen diversas herramientas para presentar sus tareas y demás actividades evaluadas y que tengan que llevar una secuencia prácticamente manual del proceso como tal. Adicional a esto, la dificultad de los alumnos para comprender la lógica de desarrollo de la metodología del proceso solucionador de problemas por diversas razones, las cuales muchas veces están con los estudiantes

<sup>1</sup> La información fue obtenida del resultado de las encuestas contestadas por docentes de la EISI, ver anexo 2.

mismos, ya sea por deficiencias académicas previas al estudio universitario y desinterés en aprender<sup>2</sup>.

Al consultar a los docentes que imparten actualmente la asignatura acerca de los inconvenientes que se les presentan para desarrollar el contenido, expresaron que la falta de equipo y herramientas para la aplicación de la metodología, la falta de una estandarización del proceso solucionador de problemas que permita guiar al estudiante en la aplicación del mismo y no pierda la secuencia de lógica a medida se avance en el desarrollo del proceso para que aprenda correcta y adecuadamente la aplicación del proceso. Estas causas se expresan gráficamente en la figura 1.2<sup>3</sup>:

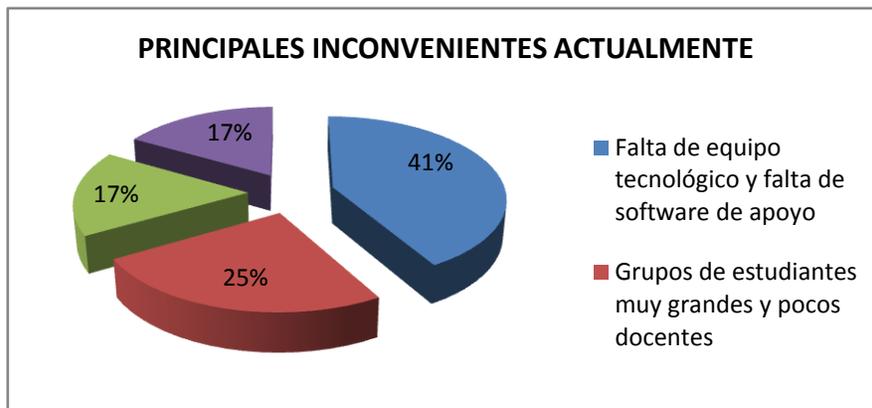


Figura 1.2 – Principales inconvenientes en la materia introducción a la informática

En cuanto a la población estudiantil de los últimos 2 años, para Introducción a la Informática, los datos estadísticos reflejan que se han inscrito un promedio de 1,052 estudiantes de los cuales un 52.3% ha reprobado la materia, 36.8% aprobaron y el 10.9% restante lo componen los alumnos que retiraron la asignatura y los que desertaron, es decir, fueron aproximadamente 550 estudiantes reprobados, 387 estudiantes aprobados y 114 estudiantes retirados y desertores por cada año, tal como se muestra en la figura 1.3. Los años correspondientes para la obtención de esta información son los años 2007 y 2008:

<sup>2</sup> Causas expresadas por los actuales docentes de Introducción a la Informática para más detalle ver anexo 2 – Encuesta a docentes 1, pregunta 3.

<sup>3</sup> Causas expresadas por los docentes de Introducción a la Informática, para más detalle ver anexo 2 – Encuesta a docentes 1, pregunta 8.



Figura 1.3 – Datos estadísticos promedio de introducción a la informática

En el caso de Programación I se inscribieron en total 865 estudiantes en ambos años, 2007 y 2008, de los cuales el 29.82% reprobaron, 66.82% aprobaron y el 3.36% restante lo componen los alumnos que retiraron la materia y los que desertaron, es decir, fueron aproximadamente 258 estudiantes reprobados, 578 estudiantes aprobados y 29 estudiantes retirados y desertores por cada año<sup>4</sup>, como se expresa en la siguiente Figura 1.4.

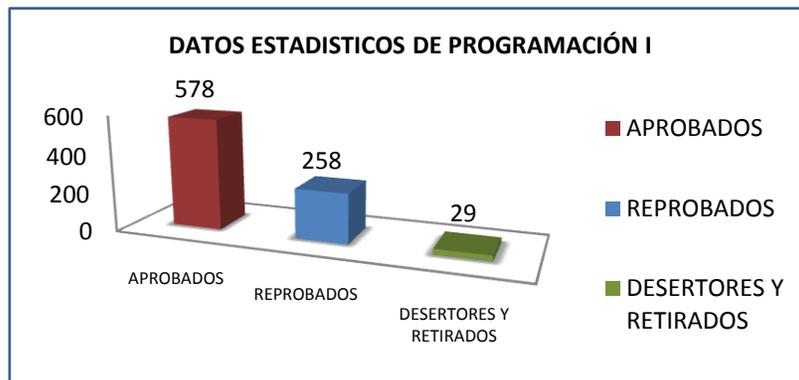


Figura 1.4 – Datos estadísticos promedio de alumnos de programación I

En el año 2009 en Introducción a la Informática se inscribieron un total de 1,130 estudiantes, incluyendo a quienes cursan la asignatura en segunda y tercera matrícula.

<sup>4</sup> Fuente: Coordinadores de la materia de los últimos 2 años, 2007 y 2008.

## **2. Alcances y Limitaciones**

### **2.1 Alcances**

- Al finalizar este proyecto los módulos que se incluirán en la construcción de la herramienta son: Planteamiento del problema, Análisis del problema, Construcción del flujograma, Validación y Ejecución del flujograma, Generación de pseudocódigo, Generación de código fuente, Generación de flujograma a partir de pseudocódigo y Generación de versiones de la aplicación.
- El desarrollo del proyecto comprenderá las siguientes etapas: Validación y actualización de listado de requerimientos, Análisis, Diseño, Programación, Documentación interna y Externa (Manual de usuario, manual de instalación y manual técnico en medios digitales y físicos), pruebas y propuesta del plan de implementación.
- La aplicación es multiplataforma, lo cual facilita que sea utilizado en cualquier sistema operativo.

### **2.2 Limitaciones**

Para el desarrollo de este proyecto no se tuvieron limitaciones que impidieran el avance del mismo.

## **3. Importancia y Justificación**

### **3.1 Importancia**

La construcción de la herramienta de desarrollo asistido de diagramas de flujo permitirá a las asignaturas de la EISI que imparten como parte del contenido la metodología para resolver problemas, un apoyo en el proceso enseñanza aprendizaje, adicional a esto, se facilitaría el desarrollo de guías de laboratorio, actividades evaluadas y ejercicios con mayor rapidez.

Los principales beneficios que se podrían obtener para cada elemento participante se detallan a continuación:

Algunos beneficios que obtendrán los estudiantes son:

- ✓ Una herramienta de que brinde las funcionalidades necesarias para la el desarrollo de la metodología para la resolución de problemas de forma dinámica, a pesar de no tener bases sólidas sobre el tema.
- ✓ Disminuir el tiempo de desarrollo de diagramas de flujo
- ✓ Disminuir el tiempo de resolución de las guías de laboratorio y ejercicios
- ✓ Apoyo de la ingeniería en el proceso enseñanza – aprendizaje
- ✓ Simplificar el desarrollo de diagramas de flujo, facilitando la comprensión de la lógica utilizada en su construcción.
- ✓ Modernizar el desarrollo de las prácticas de laboratorio
- ✓ Proporcionar al estudiante una herramienta interactiva y de fácil distribución.

Estos beneficios se podrían replicar a las siguientes generaciones todas las carreras que utilicen esta técnica.

Adicional a esto, se beneficiaran a los docentes aportándoles una herramienta que les proporcione ventajas como:

- ✓ Disminución de tiempos de desarrollo de guías de laboratorio
- ✓ Disminución de algunas dudas básicas de la construcción de diagramas de flujo con el uso de asistentes.
- ✓ Disminuir la carga académica y administrativa de EISI.
- ✓ Satisfacción de brindar a los estudiantes más herramientas para la construcción de diagramas de flujo

Los beneficios que tendrá la Facultad de Ingeniería y Arquitectura, serán:

- ✓ Estudiantes con una mejor herramienta de trabajo
- ✓ Docentes con una herramienta de apoyo en las asignaturas correspondientes
- ✓ Trabajos de mejor calidad
- ✓ Fácil acceso a una herramienta útil para el desarrollo curricular
- ✓ Promover la utilización de software de libre distribución
- ✓ Disminuir la carga administrativa de la FIA.

Como apoyo estadístico del impacto de la herramienta principalmente en Introducción a la informática por ser una de las asignaturas con mayor población estudiantil y que brinda las bases de la metodología para resolver problemas, sin embargo, esta herramienta, de igual manera traería beneficios a estudiantes de otras asignaturas en las que se imparta dicha metodología.

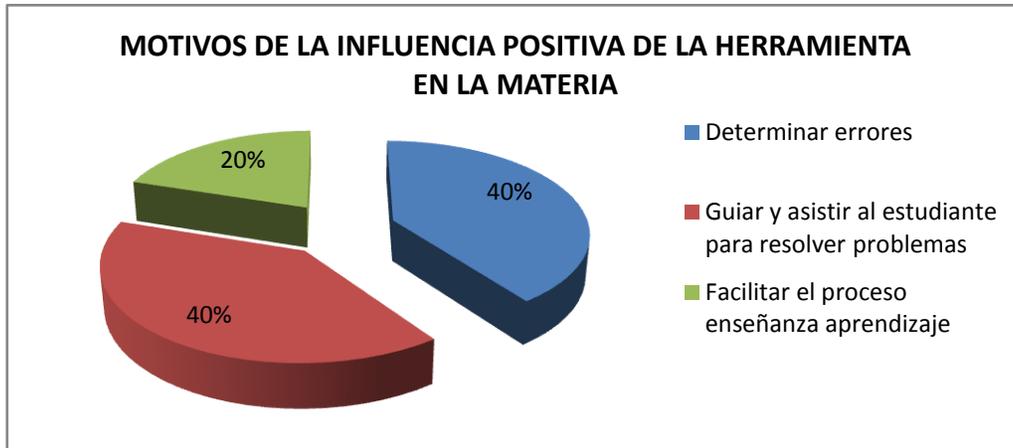


Figura 3.1 – Influencia de la herramienta a desarrollar

Además de esto, a continuación se muestran algunos de los inconvenientes que actualmente se presentan para desarrollar la metodología para la resolución de problemas en las diferentes asignaturas, los cuales podrían ser subsanados, tal es el caso del equipo informático disponible para desarrollo de laboratorios pues como se especifico en la sección 7 Factibilidades, se cuenta y se contara, a mediano plazo con mejor equipo para lleve a cabo la implementación de esta herramienta<sup>5</sup>.

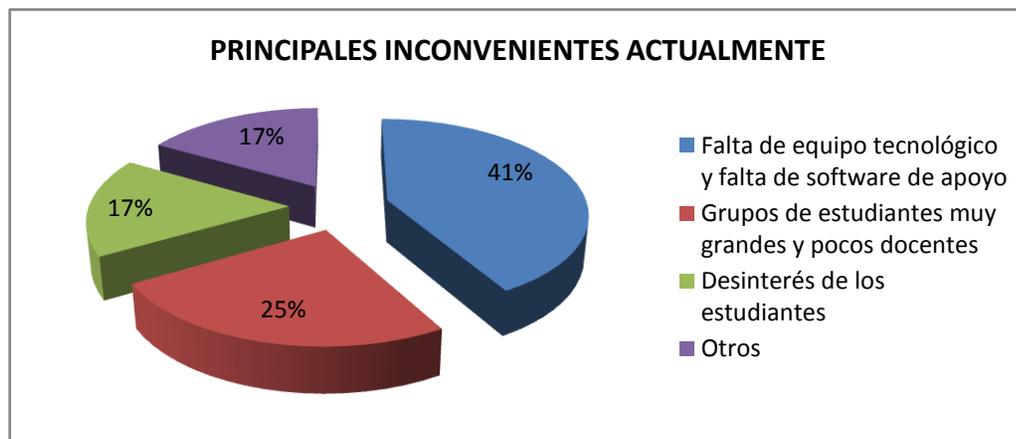


Figura 3.2 – Principales inconvenientes identificados por los docentes

<sup>5</sup> Datos obtenidos como resultado de la Encuesta a Docentes que se cita en el anexo 15.6.2

## 3.2 Justificación

El desarrollo del proceso solucionador de problemas, no es sencillo de realizarse, sobre todo cuando no se cuentan con herramientas que proporcionen las facilidades para llevarlo a cabo. Sin embargo, actualmente se hace uso de herramientas que suplen las necesidades básicas de su desarrollo. Estas herramientas son Microsoft Word utilizadas para realizar la formulación del problema, el análisis, el pseudocódigo y las pruebas sobre el diagrama. Además, se utiliza Microsoft Visio para elaborar el diagrama de flujo y el compilador Turbo C ++ 3.0 para desarrollar el código fuente.

Todo esto resulta muy complicado, pues es necesario tener a disposición todas estas herramientas para facilitar un poco el desarrollo del proceso solucionador de problemas, y en caso de no tener acceso a ellas, realizar todo el proceso manualmente, invirtiendo más tiempo y esfuerzo en llevarlo a cabo.

La construcción de la herramienta para el soporte al desarrollo del proceso solucionador de problemas pretende ahorrar tiempo, facilitar las actividades y proporcionar a los estudiantes una ayuda básica para el desarrollo de su aprendizaje, sin dejar de lado su estudio y su asistencia a clases presenciales para conocer adecuadamente las reglas que deben seguir para un mejor desarrollo de los mismos. Este software no garantizará que los estudiantes aprobarán alguna asignatura, únicamente pretende facilitar el aprendizaje del proceso solucionador de problemas de manera interactiva para se simplifique el desarrollo del estudiante en este tópico.

La principal causa que motivó el desarrollo de la herramienta es proporcionar un software de propósito específico para brindar soporte al desarrollo del proceso solucionador de problemas para contribuir al aprendizaje de manera fácil y sencilla, ahorrando tiempo y esfuerzo en la elaboración de cada una de sus fases. Adicional a esto, se tendrá la oportunidad de utilizar esta herramienta pues por ser de carácter interactivo e intuitivo, apoyará el desarrollo de esta temática y por supuesto contribuirá a llevar a cabo las actividades de laboratorio, tareas, ejercicios y otras evaluaciones que beneficiarán a cualquier persona que desee utilizar el proceso solucionador de problemas.

Además, se pretende promover la utilización de herramientas de libre distribución, de manera que se disminuya la utilización de software con copias ilegales, contribuyendo así a crear una conciencia y una integridad profesional para ser capaces de contrarrestar todas aquellas actividades que vayan en contra de principios y valores sociales y morales.

Según los resultados de encuestas realizadas a estudiantes que cursan la asignatura de Introducción a la Informática de la Universidad de El Salvador, el 87% de los estudiantes consideran que la herramienta sería de beneficio para la asignatura y que estarían dispuestos a utilizarla, principalmente para facilitar las actividades que lleva inmersas el proceso solucionador de problemas.

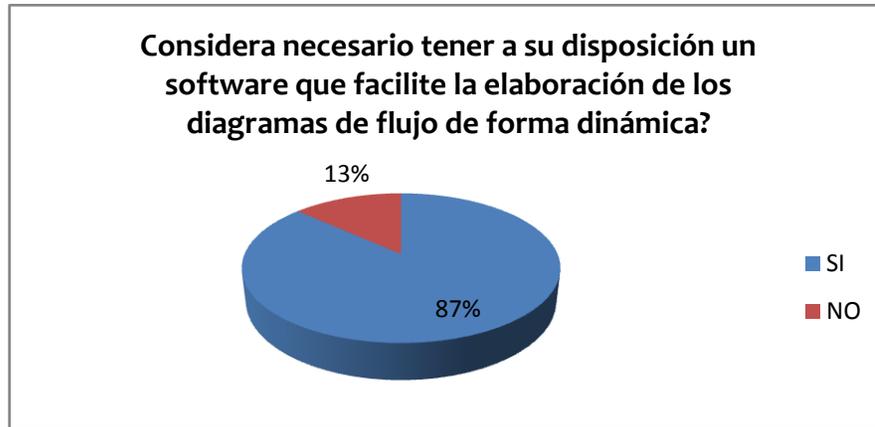


Figura 3.3 – Resultados de encuesta a los estudiantes

Además, según resultados obtenidos de encuestas realizadas a docentes que imparten la asignatura de Introducción a la Informática de las causas de la aceptación para utilizar la herramienta son los siguientes:

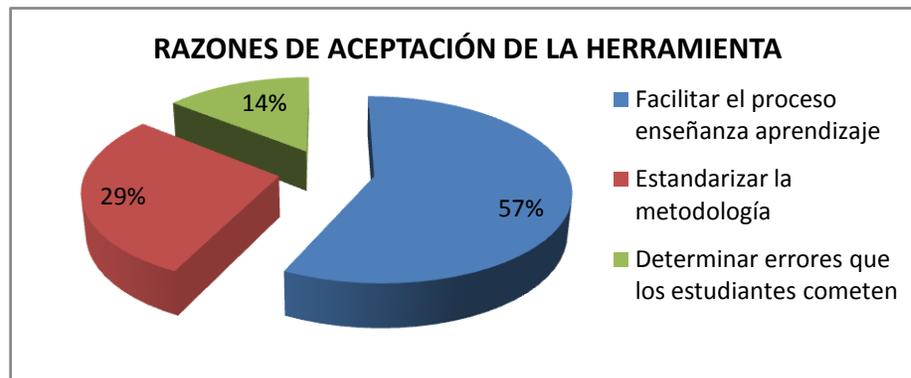


Figura 3.4 – Resultados de aceptación de la herramienta

Los docentes de la cátedra expresaron que facilitaría, sobre todo a los estudiantes, el contar con estándares para el desarrollo del proceso solucionador de problemas en cada una de sus etapas, de igual manera, permitiría facilitar el proceso enseñanza - aprendizaje, aumentando el desarrollo de ejercicios prácticos debido a que se podrían reducir los tiempos de realización de los mismos. La determinación de errores podría disminuir también, ya que la herramienta pretende advertir a quien la utilice respecto a los errores comunes cometidos al momento de llevar a cabo el proceso, principalmente en el desarrollo de los diagramas de flujo.

## 4. Estudio de Factibilidades

### 4.1 Factibilidad Técnica

#### 4.1.1 Software de desarrollo

Para la etapa de desarrollo se utilizará el lenguaje de programación Java en su versión 1.5 y el entorno de desarrollo NetBeans 6.5.

En los equipos de desarrollo se instalará el paquete Java SE y NetBeans (JDK 5.0 Update 18 y NetBeans 6.5.1) el cual requiere los mismos recursos de hardware para la operatividad de los dos software.

##### 4.1.1.1 Requerimientos mínimos para la instalación del paquete Java SE y NetBeans

En la tabla 4.1 se detallan los requerimientos mínimos para la instalación del paquete Java SE y NetBeans (JDK 5.0 Update 18 y NetBeans 6.5.1).

		Requerimiento mínimo por Sistema Operativo (32 bits)	
		Windows XP	Linux/Solaris
Elemento	RAM	512MB	512MB
	Espacio en Disco duro	750MB	650MB
	Velocidad del procesador	800MHz	450Mhz

Tabla 4.1 - Requerimiento mínimo de hardware para la instalación del paquete Java y NetBeans (JDK 5.0 Update 18 y NetBeans 6.5.1)<sup>6</sup>

#### 4.1.2. Hardware de desarrollo

La información que contiene la tabla 4.2 proporciona las características generales de los 4 equipos con los que se cuenta para el desarrollo del proyecto.

		Equipos			
		Equipo 1	Equipo 2	Equipo 3	Equipo 4
Elemento	Tipo y Modelo	Laptop, Toshiba Tecra	Laptop, HP Pavilion	Desktop, Clon	Desktop, Clon

<sup>6</sup> Java. Download J2SE5.0, Recuperado el 26 de mayo de 2009, de [http://java.sun.com/j2se/1.5.0/install\\_jdk1.5\\_18-nb6.5.1.html](http://java.sun.com/j2se/1.5.0/install_jdk1.5_18-nb6.5.1.html)

	A8 S8513	zv6000		
Procesador	Intel Core 2 Duo T5600	AMD Athlon 64	Celeron Dual Core	Intel Dual Core
Velocidad de procesador	1.83GHz	3200+ (2GHz)	2.0GHz	2.4GHz
Memoria física	1GB	1GB	1GB	1GB
Disco duro	100GB, 5400rpm	80GB, 5400rpm	80GB, 7200rpm	80GB, 7200rpm
Unidad de disco	CD/CD-RW/DVD-RW	CD/CD-RW/DVD-RW	CD/CD-RW/DVD-RW	CD/CD-RW/DVD-RW
Sistema operativo	Windows XP, profesional SP2	Windows XP, profesional SP2	Windows XP, profesional SP2	Windows XP profesional SP2

*Tabla 4.2 - Descripción del Hardware*

Estos equipos están conectados por medio de un Wireless Router TP-Link TL-WR542G7 con 4 puertos el cual provee de conexión a Internet de forma alámbrica e inalámbrica.

Además se cuenta con un impresor Canon IP 1800, el cual servirá para imprimir la documentación generada del proyecto.

#### 4.1.3. Software de implementación

La maquina virtual de Java (JVM por sus siglas en ingles) proporciona la facilidad de ejecución del software a desarrollar en diferentes Sistemas Operativos, lo que permitirá fácilmente la distribución de la herramienta a construir. Con esto, cualquier estudiante que reciba el software no se deberá de preocupar por el Sistema Operativo que posea su equipo de trabajo.

En la tabla 4.3 se detallan los requerimientos mínimos para la instalación de la Maquina virtual de Java.

Elemento	Requerimiento mínimo			
	Windows 2000, XP, Vista, Windows 2003	Linux	Mac OS X	Solaris
RAM	64MB	64MB	Utilizar la función de	64MB

Disco Duro	98MB	58MB	<b>actualización de software</b> (disponible en el menú de Apple).	53MB
Versión	6.0	6.0		5.0

Tabla 4.3 - Requerimiento mínimo de hardware para la instalación de la JVM<sup>7</sup>

#### 4.1.4. Hardware de implementación

Para determinar el hardware requerido para la implementación del software, se considero los requerimientos mínimos para el sistema operativo y la maquina virtual de java, ambos serán el software mínimo para el buen funcionamiento de la herramienta a construir.

La tabla 4.4 contiene la información de los requerimientos de hardware de los sistemas operativos con soporte de Java.

		Requerimientos mínimos por Sistema Operativo			
		Windows XP Pro <sup>8</sup>	Linux, Distribución Ubuntu 8.04 <sup>9</sup>	Mac OS X <sup>10</sup>	Solaris 10 <sup>11</sup>
Elemento	Microprocesador	Pentium 233MHz	300MHz	867MHz	Sistemas de 32 y 64 bits basados en CPU de AMD, Intel y VIA x86
	RAM	64MB	64MB	512MB	512MB
	Disco Duro	1.5GB	4GB	9GB	2GB

Tabla 4.4 – Requerimientos mínimos de hardware por Sistema Operativo

En la sección 4.1.3 Software de Implementación, se detalla los requerimientos mínimos de hardware para la instalación de la maquina virtual de Java.

<sup>7</sup> Java. *Maquina Virtual de Java*, Recuperado el 1 de mayo de 2009, de <http://www.java.com/es/download/manual.jsp>

<sup>8</sup> Microsoft, Ayuda y Soporte, Recuperado el 29 de mayo de 2009, de <http://support.microsoft.com/kb/314865>

<sup>9</sup> Ubuntu, Documentación, Recuperado el 29 de mayo de 2009, de <https://help.ubuntu.com/community/Installation/SystemRequirements>

<sup>10</sup> Apple, Mac OS X Leopard, Recuperado el 29 de mayo de 2009, de <http://www.apple.com/macosx/techspecs/>

<sup>11</sup> Sun, System Requirements, Recuperado el 29 de mayo de 2009, de <http://es.sun.com/practice/software/solaris/specs.jsp>

#### 4.1.5. Equipo informático actual de la EISI

La tabla 4.5 contiene información referente a las descripciones técnicas de las computadoras con las que cuenta actualmente la Escuela de Ingeniería en Sistemas informáticos.

Procesador	Memoria	Disco Duro	Sistema Operativo	Cantidad
AMD Athlon k7	512MB	60GB	Windows XP Pro	10
Pentium 4 de 1.8GHz	512MB	40GB	Windows XP Pro	20
Pentium 4 de 1.8GHz	256MB	40GB	Windows XP Pro	20
IBM Pentium 4 de 1.8GHz	256MB	40GB	Windows XP Pro	5
Pentium I, II y otras de baja frecuencia.	64MB	50MB	MS-DOS	19

*Tabla 4.5 – Descripción técnica de las computadoras con las que cuenta la EISI.*

En base a la información anterior, se concluye que:

Para la utilización del software a construir, la EISI debe actualizar al menos 19 computadoras que actualmente cuentan con poca capacidad de hardware.

Actualmente existe un plan de renovación de los equipos informáticos de los centros de cómputo de la EISI, con lo cual se espera recibir el próximo año 111 equipos con las características que se presentan en la tabla 4.6.

Elemento	Descripción
Procesador	Core 2 Duo de 3GHz
RAM	2GB
Disco Duro	200GB

*Tabla 4.6 – Descripción de equipo informático a renovar en la EISI<sup>12</sup>.*

#### 4.1.6. Conclusión de factibilidad técnica

Al contar con cada una de las herramientas descritas anteriormente se concluye que técnicamente el proyecto es factible, ya que se cuenta tanto con el hardware y software de desarrollo e implementación para dicho proyecto.

<sup>12</sup> La información anterior fue proporcionada por la Jefatura de la Comunicación y Ciencias de la computación de la EISI.

## 4.2 Factibilidad Económica

El estudio de la factibilidad económica se basará en la estimación del costo del desarrollo del software, adquisición de recursos informáticos para la implementación del la herramienta a desarrollar (hardware y software) y beneficios tangibles, por medio del análisis del costo/beneficio. Este estudio se basará en la teoría de Kendall & Kendall<sup>13</sup>.

### 4.2.1. Costo por el desarrollo del software

En base a estimaciones y cálculos del costo total del desarrollo de software se estima un monto de **\$ 9,123.27** para el desarrollo del proyecto. Ya que este proyecto forma parte del proceso de graduación dentro de la Escuela de Ingeniería de Sistemas Informáticos, dicha escuela no incurrirá en este costo.

### 4.2.2. Costo por recurso informático para la implementación de la herramienta

#### 4.2.2.1. Costo de hardware

La EISI cuenta con el equipo necesario para poder implementar la herramienta a desarrollar, las descripciones de estos equipos se puede ver en la sección 5.1.4 *Hardware de implementación*.

#### 4.2.2.2. Costo de software

En la tabla 4.7 se detalla el costo para adquirir por la utilización de la Máquina Virtual de Java, elemento indispensable para la ejecución del software a desarrollar.

Software	Tipo de Licencia	Costo (\$ USD)
Java Virtual Machine 6 Update 13	GPL	0.00

Tabla 4.7 – Detalle del costo del software de operación<sup>14</sup>

### 4.2.3. Beneficios tangibles

Para poder enseñar la metodología para la resolución de problemas, la EISI no cuenta con un software que de soporte a todas las fases que componen tal metodología. A continuación se describen dos alternativas que brinde soporte a la metodología para la resolución de problemas.

<sup>13</sup> Kendall, K. E. y Kendall, J. E. (2005), Análisis y diseño de sistemas (6ª. ed.). México: Pearson, p.56

<sup>14</sup> Java, Licencia de la tecnología Java, Recuperado el 29 de mayo de 2009, de <http://www.java.com/es/download/license.jsp>

**4.2.3.1. Alternativa 1**

Si la EISI deseara obtener una herramienta de apoyo a los estudiantes para la elaboración de los diagramas de flujo, necesitaría recursos financieros para la adquisición de licencias de un software con las características necesaria. Debido a que en el mercado no se encuentra con una herramienta que reúna todas las funcionalidades que incluirá el software a desarrollar, la EISI debería de elegir un software en base a las particularidades y el costo que implicaría la adquisición de una herramienta de este tipo. En la tabla 4.8 se presenta un listado de este tipo de software.

Funcionalidad	Software	Costo (\$ USD)
Planteamiento del problema.	Microsoft Visio Estándar 2007 <sup>15</sup>	0.0
Análisis del problema.		
Diseño de la solución	DFD V 1.1	0.0
Codificación	Flujo de cristal para C++ V3.83	279.00
<b>Costo total</b>		<b>\$ 279.00</b>

*Tabla 4.8 – Precio de herramientas para soporte al proceso solucionador de problemas.*

Tomando en cuenta que se implementaría en un centro de computo con un promedio de 25 computadoras, al multiplicar \$279 \* 25 = 6975

Con lo cual, el costo de licenciamiento para la EISI seria de **\$ 6,975.00** para un centro de cómputo con 25 máquinas.

**4.2.3.2. Alternativa 2**

Para poder satisfacer la necesidad de la EISI se propone desarrollar una herramienta que brinde soporte en el desarrollo del proceso solucionador de problemas con un costo de **\$ 9,123.27**, pero para la EISI tendrá un costo de \$0.0 por tratarse de un trabajo de graduación.

**4.2.4. Análisis Costo/Beneficio**

No se ha considerado un costo de actualización de equipo informático, debido a que se cuenta con un plan de renovación del mismo. Para ver detalle de las características de los nuevos equipos a adquirir verificar la sección 4.1.5, tabla 4.6.

En la tabla 4.9 se detallan los costos y beneficios del proyecto.

<sup>15</sup> Debido a un convenio entre la UES y Microsoft, el costo de Microsoft Visio Estándar 2007 es \$ 0.0

<b>Costo/Beneficio</b>	<b>Total \$ USD</b>
Costos alternativa 1	<b>\$ 6,975.00</b>
Costos alternativa 2	\$0.0

*Tabla 4.9 – Comparación de Costos y Beneficios*

El resultado genera un ahorro estimado de **\$ 6,975.00**

#### **4.2.5. Conclusiones de la factibilidad económica**

En base a lo anterior se concluye que el proyecto es económicamente factible, debido a que se genera un ahorro de \$ 6,975.00 para la EISI.

### **4.3. Factibilidad Operativa**

Para el desarrollo del proyecto propuesto, se necesita determinar si la realización del mismo es operativamente factible.

Para determinar si la operatividad de la herramienta será factible es importante conocer la aceptación de los usuarios que pueden ser beneficiados con el desarrollo del sistema propuesto. Se realizó una encuesta entre los estudiantes de la carrera de Ingeniería de Sistemas Informáticos y docentes de la asignatura de Introducción a la Informática (Ver anexo 2 – Encuesta a docentes 2). Se escogió esta población ya que es la que más se relaciona con la metodología de resolución de problemas, sin embargo el software será propiedad a la EISI por lo que estará disponible para cualquier docente y/o alumno de cualquier materia con conocimientos del proceso solucionador de problemas. En base a los resultados de la encuesta elaboramos un análisis considerando los siguientes rubros:

- ✓ Aceptación de los usuarios
- ✓ Utilización del Software

#### **4.3.1. Aceptación de los usuarios**

Para determinar si los usuarios están de acuerdo con la utilización del software se evaluaron dos poblaciones que fueron: los Docentes y los alumnos

Por parte de los docentes se entrevisto un total de 8 docentes, se tomo esta población ya que estos docentes están relacionados con la enseñanza del proceso solucionador de problemas. El 100% de los docentes encuestados, están de acuerdo con la elaboración de un software con las características propuestas, este resultado se debe a que los docentes consideran apropiado contar con una herramienta para impartir la metodología de solución de problemas.

Desde el punto de vista de los alumnos, en total se entrevistaron 400 estudiantes de población de 1117 inscritos, los cuales representan el 35% de la población inscrita, El 90% está de acuerdo con el desarrollo de una herramienta que le ayude desarrollar la metodología de resolución de problemas.

A continuación se muestra el gráfico donde se puede observar el porcentaje de aceptación que tendría la herramienta en los estudiantes.



*Figura 4.1 – Grafica de aceptación del usuario*

#### **4.3.2. Utilización del Software**

Según los resultados de la factibilidad técnica y económica, se puede determinar que la herramienta es operativamente factible, ya que la Escuela de sistemas informáticos cuenta con los equipos necesarios para utilizar el software propuesto (Para verificar el hardware con el que cuenta la EISI ver la sección 5.1.4. Hardware de implementación). Por otra parte, la Jefatura de Comunicación y Ciencias de la Computación cuenta con un proyecto de renovación de centros de cómputo para la EISI, dicho proyecto actualmente está en marcha y se estima que para el año 2010 se implemente dicha renovación. La realización de este proyecto fortalece la operatividad del software propuesto, ya que se contará con más recursos para utilizar la herramienta.

#### **4.3.3. Conclusión de la factibilidad operativa**

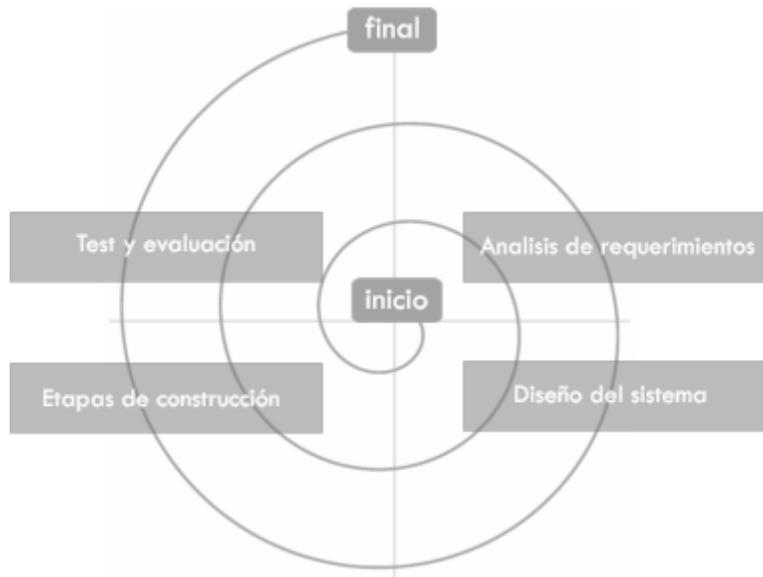
Debido a que la EISI cuenta con el equipo informático necesario para poder operar el software propuesto y que los usuarios consideran apropiado una herramienta que les apoye en el desarrollo de la metodología para la solución de problemas, se determina que el software es operativamente factible.

## 5. Metodología utilizada

### 5.1. Ciclo de vida de desarrollo

La metodología a utilizar para el desarrollo del proyecto es el modelo de ciclo de vida en espiral, el cual reconoce la necesidad de pasar por la secuencia de Análisis de requerimientos, Diseño, Implementación y Pruebas más de una vez, principalmente para eliminar los riesgos. Es decir, se trabaja en la construcción de versiones parciales para obtener como resultado partes funcionales del sistema completo.

La estructura del ciclo de vida en espiral es como el siguiente:



*Figura 5.1 – Esquema del ciclo de vida en espiral*

Para aplicar esta metodología, se realizará una fusión desarrollando el modelo de ciclo de vida en cascada en cada iteración del espiral, llevando a cabo las etapas de análisis, diseño, construcción, pruebas y documentación. En la primera etapa se realizará una investigación para identificar los requerimientos de cada módulo del proyecto. La segunda etapa servirá para realizar el diseño del sistema, detallando las interfaces a utilizar, los comportamientos de los elementos y la interrelación entre estos. En la etapa de construcción se desarrolla la programación como tal del módulo que se ha analizado y diseñado previamente. Luego de ello, se lleva a cabo la documentación de la parte del sistema que se ha desarrollado.

### **5.1.1. Ventajas, Desventajas y utilización del ciclo de vida en espiral.**

#### ***5.1.1.1. Ventajas del modelo de ciclo de vida en espiral***

Las ventajas con las que cuenta este modelo son:

- ✓ Disminuye el riesgo de rechazar el proyecto porque se construyen sistemas pequeños.
- ✓ Se desarrollan las funcionalidades independientemente.
- ✓ Si se detecta un error grave, solo desecharmos la última versión.
- ✓ Apto para desarrollos en los que no se conoce a priori sus especificaciones o la tecnología a utilizar.
- ✓ Permite iniciar un proyecto con un alto grado de incertidumbre.
- ✓ Bajo riesgo de retraso en caso de detección de errores.
- ✓ Posibilidad de recolectar métricas en la primera iteración para diseñar e implementar las iteraciones subsecuentes.

#### ***5.1.1.2. Desventajas del modelo de ciclo de vida en espiral***

- ✓ Costoso respecto a tiempo y dinero invertido a lo largo del desarrollo del proyecto.
- ✓ Dificultad para evaluar riesgos por la falta de habilidad en el equipo de trabajo para realizar esta actividad.
- ✓ Necesidad de la presencia o la comunicación continua con el cliente o usuario.

#### ***5.1.1.3. Utilización del modelo de ciclo de vida en espiral***

La principal característica de este modelo es que no es lineal sino cíclico y es utilizado principalmente para desarrollar proyectos grandes o sistemas de gran complejidad. Es un modelo de ciclo de vida poco utilizado ya que en nuestro país, poco se desarrollan proyectos novedosos y/o se utilizan tecnologías nuevas. Otra de las causas por la que no es utilizado es porque no se tienen completamente definidos los requerimientos o bien son diversas las solicitudes de los usuarios que participan. Muchas veces dentro de este modelo se puede hacer uso de cualquier ciclo de vida como parte de las iteraciones que se realizan.

## **5.2. Herramientas de investigación**

### **5.2.1. Técnicas**

Para el desarrollo del proyecto de construcción de la herramienta de desarrollo asistido de diagramas de flujo se utilizarán las siguientes técnicas:

- ✓ Recopilación y análisis de requerimientos
  - Entrevistas: conversación entre dos o más personas la cual es guiada por una serie de preguntas previamente elaboradas para obtener información sobre los requerimientos y que es relevante para el desarrollo del proyecto.

- Cuestionarios: Método utilizado para la recolección de información, el cual consiste en una serie de preguntas abiertas o cerradas y generalmente permiten cuantificar el resultado de una entrevista.
  - Encuestas: Técnica de investigación que consiste en formular una serie de preguntas recogidas en un cuestionario, para conocer la opinión del público sobre un asunto determinado y reflejarla mediante estadísticas.
  - Tormenta de ideas: Técnica que consiste en identificar y anotar todas las ideas que el equipo de trabajo mencione para dar alternativas de solución a una situación propuesta.
  - Diagrama de Causa – Efecto: Es una forma para ordenar y presentar de forma muy concentrada, todas las causas que supuestamente pueden contribuir a un determinado efecto.
  - UML: es un lenguaje de modelado capaz de describir sistemas como conjuntos de elementos y relaciones. Los diagramas utilizados para el análisis de requerimientos son el diagrama de caso de uso y los diagramas de secuencia.
- ✓ Diseño del sistema
- UML: El principal diagrama que se elabora es el diagrama de clases.
- ✓ Desarrollo o construcción del sistema
- Construcción en base a análisis y diseño realizados: Seguir los modelos creados mediante la técnica de desarrollo de diagramas UML.
  - Reutilización de componentes<sup>16</sup>: Es el uso de software existente para desarrollar un nuevo software. La reutilización de código ha sido empleada desde los primeros días de la programación. La reutilización de códigos programados es una técnica común que intenta ahorrar tiempo y energía, reduciendo el trabajo redundante.
- ✓ Implementación y pruebas
- Utilización de método de programación lineal, en base al análisis y diseño realizado y creación de scripts de pruebas unitarias e integrales del sistema.

---

<sup>16</sup> Exception!, Diccionario de informática, internet, tecnologías y computación, Recuperado el Viernes 23 de Mayo de 2009, <http://www.alegsa.com.ar/Dic/reutilizacion%20de%20codigo.php>

✓ Documentación

- Estándares de documentación: Técnica utilizada para obtener información de fuentes bibliográficas virtuales, páginas web, enciclopedias en línea, diccionarios, libros e incluso artículos de revistas.

## 5.2.2. Herramientas y equipo

### 5.2.2.1. Herramientas

Para cada una de las etapas del modelo de ciclo de vida que se utilizará se han identificado las herramientas necesarias para el desarrollo de las mismas, a continuación se detallan:

✓ Análisis de requerimientos

- Método de formulación de problemas de la “Caja Negra”: Es un método que permite mostrar las entradas y salidas de un sistema, ya sean estas materia, energía o información. Sin embargo los procesos permanecen fuera del alcance total del analista.
- Enfoque de sistemas (Teoría de Sistemas): Son las actividades que determinan un objetivo general y la justificación de cada uno de los subsistemas, las medidas de actuación y estándares en términos del objetivo general, el conjunto completo de subsistemas y sus planes para un problema específico.
- Diagrama Ishikawa o Diagrama Causa – Efecto: Es una forma para ordenar y presentar de forma muy concentrada, todas las causas que supuestamente pueden contribuir a un determinado efecto. Nos permite, por tanto, lograr un conocimiento común de un problema complejo, sin ser nunca sustitutivo de los datos.
- Diagramas de casos de uso: Los diagramas de casos de uso son importantes para modelar el comportamiento de un sistema, un subsistema o una clase. Cada uno muestra un conjunto de casos de uso, actores y sus relaciones.

✓ Diseño del sistema

- Diagrama de clases: Representa las clases que serán utilizadas dentro del sistema y las relaciones que existen entre ellas. Los diagramas de clases por definición son estáticos, esto es, representan que partes interactúan entre sí, no lo que ocurre cuando.
- Utilización de Módulo UML para NetBeans 6.5.1 para la elaboración de los diagramas
- Utilización de herramientas ofimáticas para hacer las descripciones

- ✓ Construcción
  - Lenguaje de programación Java: Es un lenguaje orientado a objetos, de una plataforma independiente. Permite elaborar y ejecutar aplicaciones independientes.
  - NetBeans 6.5.1: Se refiere a una plataforma para el desarrollo de aplicaciones de escritorio usando Java y a un entorno de desarrollo integrado (IDE).
  
- ✓ Pruebas
  - Pruebas unitarias: Es una forma de probar el correcto funcionamiento de un módulo de código. Esto sirve para asegurar que cada uno de los módulos funcione correctamente por separado.
  - Pruebas integrales: Permite asegurar el correcto funcionamiento del sistema o subsistema en cuestión.
  
- ✓ Documentación
  - Herramientas ofimáticas: La ofimática es el conjunto de técnicas, aplicaciones y herramientas informáticas que se utilizan en funciones de oficina para optimizar, automatizar y mejorar los procedimientos o tareas relacionados.

Las herramientas ofimáticas permiten idear, crear, manipular, transmitir y almacenar información necesaria en una oficina o lugar de trabajo.

#### **5.2.2.2. Equipo**

El equipo que se utilizará para el desarrollo de la herramienta de software consiste en:

- ✓ Dos computadoras portátiles
- ✓ Dos computadora de escritorio
- ✓ Muebles de computadora, sillas y mesas.
- ✓ Un router inalámbrico
- ✓ Cables UTP (Patch Core)
- ✓ Red local LAN
- ✓ Conexión a Internet
- ✓ Un impresor
- ✓ Papelería y materiales de oficina (Lapiceros, agendas, resmas de páginas de papel, engrapadoras, tintas, folders, fastener)
- ✓ Teléfono fijo y teléfonos celulares

### 5.2.3. Recursos

#### 5.2.3.1. Recursos humanos

Para llevar a cabo el desarrollo de esta metodología trabajaran cuatro personas, quienes se encargaran de desarrollar cada una de las fases que expone el modelo de ciclo de vida en espiral. La disposición de tiempo es de 20 horas semanales por cada persona, trabajando por lo menos 4 horas diarias, es decir, el grupo de trabajo invertirá aproximadamente un total de 80 horas semanales a lo largo del desarrollo del proyecto.

#### 5.2.3.2. Otros Recursos

Para efectos de comunicación y elaboración de documentos también se harán uso de elementos como:

- ✓ Comunicación en línea (Messenger, videoconferencias, correo electrónico)
- ✓ Conexión remota a equipo de trabajo mediante LogMeIn

### 5.3. Definición de la situación problemática

Comprende la identificación del problema u oportunidad de desarrollo. Tiene por objetivo determinar si la realización del proyecto es factible.

#### 5.3.1. Identificación del problema

A continuación se muestra una tabla comparativa, que permitió al equipo de trabajo seleccionar la herramienta de resolución de problemas más adecuada. Esto se hizo considerando, en primera instancia las siguientes herramientas metodológicas.

- Variación en el tiempo
- Diagrama Causa-Efecto
- Diagrama de Pareto
- Tormenta de ideas

Debido a que las herramienta a seleccionar será utilizada para conocer las raíces del problema a resolver, determinando si el problema sería solucionado con una herramienta de apoyo, el equipo de trabajo escogió el Diagrama de Causa – Efecto; considerándolo el más adecuado para los fines buscados y beneficiándose de las muchas ventajas que ofrece, tal como lo muestra la tabla 5.1.

Herramienta	Permite identificar el problema	Permite la identificación y análisis de las causas del problema	Facilita la identificación de la solución del problema	Permite una visualización rápida y fácil	Fomenta la participación grupal	Experiencia del equipo de trabajo	Puntaje Total
Variación en el tiempo	✓	✓	✗	✗	✗	✗	✓✓

Diagrama Causa – Efecto							
Diagrama de Pareto							
Tormenta de ideas							

Tabla 5.1 – Criterios de selección de la herramienta de resolución de problemas.

Véase la aplicación técnica de la herramienta Diagrama Causa – Efecto en el Capítulo II: “Análisis del Sistema Informático - Tema 2.1 – Identificación del problema”.

### 5.3.2. Planteamiento del problema

Para llevar a cabo el planteamiento del problema se hizo uso del método de formulación de problemas de la “caja negra”.<sup>17</sup> En teoría de sistemas, se denomina caja negra a aquel elemento que es estudiado desde el punto de vista de las entradas que recibe y las salidas o respuestas que produce, sin tener en cuenta su funcionamiento interno.

En el método de la “caja negra” el comportamiento y/o la organización del sistema considerado no se conocen por completo y tienen que ser determinados. Asimismo, las cantidades externas del sistema sean entradas o salidas, se suponen observables o medibles al nivel de resolución respectivo y pueden obtenerse por experimentación, es decir lo que se trata de determinar el comportamiento, proceso y/o estructura del sistema.

## 5.4. Gestión de proyectos

Dentro de la gestión de proyectos, se encuentran los siguientes métodos:

- ADM
- Ruta Crítica
- Diagramas de Hitos
- Diagramas de Gantt
- Diagrama PERT

Como grupo se ha seleccionado el Diagrama Gantt como herramienta para la gestión de proyectos, con la finalidad de representar las diferentes fases, tareas y actividades programadas durante todo el proyecto, a continuación se muestra una breve información sobre esta técnica.

<sup>17</sup> Teoría general de los sistemas: Fundamentos, desarrollo, aplicaciones. Bertalanffy, Ludwing Von 1992.

- El diagrama de Gantt, gráfica de Gantt o carta Gantt es una herramienta gráfica cuyo objetivo es mostrar el tiempo de dedicación previsto para diferentes tareas o actividades a lo largo de un tiempo total determinado. A pesar de que, en principio, el diagrama de Gantt no indica las relaciones existentes entre actividades, la posición de cada tarea a lo largo del tiempo hace que se puedan identificar dichas relaciones e interdependencias.
- La ventaja principal de la gráfica de Gantt es su simplicidad. El analista de sistemas encontrara que esta técnica no solamente es fácil de usar, sino que también lleva por si misma a una comunicación valiosa con los usuarios finales. Otra ventaja del uso de una gráfica de Gantt es que las barras que representan actividades o tareas son trazadas a escala, esto es, el tamaño de la barra indica la longitud relativa del tiempo que llevara a terminar la tarea.

## 5.5. Análisis y Diseño

Para poder realizar la fase de análisis del sistema se utilizará el análisis orientado a objetos, a continuación se muestra información referente a esta técnica.

### 5.5.1.1. Fundamentos del Enfoque Orientado a Objetos

#### Objetos y Clases

Una *clase* es una categoría o grupo de cosas que tienen atributos y acciones similares. Un *objeto* es una instancia de una clase. Un objeto, también, es una unidad que encapsula estado y comportamiento. Tanto los objetos como las clases pueden caracterizar una entidad física (alumno, libro, etc.) o abstracta (fórmula matemática, juego, etc.).

#### UML y El Modelado

**UML** es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra una gran cantidad de software, desde una perspectiva OO.

El Lenguaje Unificado de Modelado (UML) es, tal como su nombre lo indica, un lenguaje de modelado y no un método o un proceso. El UML está compuesto por una notación muy específica y por las reglas semánticas relacionadas para la construcción de sistemas de software. El UML en sí mismo no prescribe ni aconseja cómo usar esta notación en el proceso de desarrollo o como parte de una metodología de diseño orientada a objetos.

El UML soporta un conjunto rico en elementos de notación gráfica. Describe la notación para clases, componentes, nodos, actividades, flujos de trabajo, casos de uso, objetos, estados y cómo modelar la relación entre esos elementos. El UML también soporta la idea de extensiones personalizadas a través de elementos estereotipados.

El UML provee beneficios significativos para los ingenieros de software y las organizaciones al ayudarles a construir modelos rigurosos, trazables y a los que se pueda dar mantenimiento; además, que soporten el ciclo de vida de desarrollo de software completo.

### **Conceptos de Modelo y Diagrama**

Un modelo captura una vista de un sistema del mundo real. Es una abstracción de dicho sistema, considerando un cierto propósito. Así, el modelo describe completamente aquellos aspectos del sistema que son relevantes al propósito del modelo, y a un apropiado nivel de detalle.

Un diagrama es una representación gráfica de una colección de elementos de modelado.

#### ***Diagrama de Casos de Uso***

El modelo de casos de uso describe la funcionalidad propuesta del sistema a desarrollar. Un Caso de Uso representa una unidad discreta de interacción entre un usuario (humano o máquina) y el sistema. Un Caso de Uso es una unidad de trabajo significativo; por ejemplo crear una solicitud y modificar una solicitud son Casos de Uso.

Cada Caso de Uso tiene una descripción que especifica la funcionalidad que se incorporará al sistema propuesto. Un Caso de Uso puede 'incluir' la funcionalidad de otro Caso de Uso o puede 'extender' otro Caso de Uso con su propio comportamiento.

#### ***Diagrama de clases***

Los diagramas de clases muestran las diferentes clases que componen un sistema y cómo se relacionan unas con otras. Se dice que los diagramas de clases son diagramas estáticos porque muestran las clases, junto con sus métodos y atributos, así como las relaciones estáticas entre ellas: qué clases conocen y a qué otras clases o qué clases son parte de otras clases, pero no muestran los métodos mediante los que se invocan entre ellas.

#### ***Diagramas de secuencia***

Los diagramas de secuencia muestran el intercambio de mensajes (es decir la forma en que se invocan) en un momento dado. Los diagramas de secuencia ponen especial énfasis en el orden y el momento en que se envían los mensajes a los objetos.

En los diagramas de secuencia, los objetos están representados por líneas intermitentes verticales, con el nombre del objeto en la parte más alta. El eje de tiempo también es vertical, incrementándose hacia abajo, de forma que los mensajes son enviados de un objeto a otro en forma de flechas con los nombres de la operación y los parámetros.

## **5.6. Codificación**

Para la fase de desarrollo de la aplicación se hará uso de las siguientes herramientas.

- Máquina virtual de Java

- NetBeans 6.5.1

### **5.7. Pruebas**

Se realizan las pruebas necesarias, centrándose en la lógica interna de la herramienta, de tal forma que se asegure que las entradas definidas producen los resultados esperados.

- Pruebas de componentes
- Pruebas de integración

El proceso de implementación de pruebas se llevará a cabo en la fase de la programación de la herramienta.

### **5.8. Mantenimiento**

La herramienta sufrirá cambios después de ser liberado, ya sea por el surgimiento de errores, adaptarse a cambios externos o que el usuario requiera mejoras funcionales o de rendimiento. Sin embargo, esta etapa de desarrollo ya no será supervisada por el equipo de trabajo ya que el alcance del proyecto culmina con la elaboración del plan de implementación.

## CAPITULO II: ANÁLISIS DE LA HERRAMIENTA

### 1. Situación problemática

#### 1.1. Identificación del Problema

Los principales problemas que se identificaron y que justifican la necesidad del desarrollo de la herramienta de desarrollo asistido de diagramas de flujo son los siguientes:

- ✓ Tiempo que invierten en el desarrollo del proceso solucionador de problemas.
- ✓ Desinterés de los estudiantes para aprender y débiles bases desde educación media
- ✓ Falta de una herramienta que brinde soporte a todo el proceso solucionador de problemas para estandarizar la información.
- ✓ Inscripción masiva de estudiantes en la asignatura de Introducción a la Informática, por ende existen grupos teóricos demasiado numerosos.
- ✓ Las herramientas de libre distribución no cuentan con todas las funcionalidades necesarias para realizar todo el proceso solucionador de problemas.
- ✓ Recarga de trabajo a la planta docente de la EISI y carga administrativa para la Facultad de Ingeniería y Arquitectura.
- ✓ Lento desarrollo de guías y ejercicios prácticos durante sesiones de clase y de laboratorio.
- ✓ El fondo presupuestario para el desarrollo tecnológico de la Facultad de Ingeniería y Arquitectura no es suficiente para cubrir las necesidades de infraestructura, equipo y tecnología para la demanda de estudiantes.
- ✓ Falta de actualización y modernización de equipo informático para impartir clases y desarrollo de laboratorios.

Algunos de los problemas expuestos afectan de manera directa o indirecta el buen desarrollo académico para los estudiantes que cursan Introducción a la Informática en la Facultad de Ingeniería y Arquitectura. Para exponer mejor la problemática se utilizó un Diagrama de Causa – Efecto o Diagrama Ishikawa en el cual se exponen algunas de las principales dificultades y razones por las cuales hace falta una herramienta de apoyo para el proceso solucionador de problemas el cual beneficiaría tanto a los estudiantes como a los docentes, facilitándoles las actividades de enseñanza aprendizaje de una manera sencilla, centralizada y estandarizada. Se seleccionó el método de “**6M**”, este es el método de construcción más común y consiste en agrupar las causas potenciales en seis ramas o espinas principales: métodos de trabajo, mano de obra, materiales, maquinaria, medición y medio ambiente. Estos seis elementos definen de manera global todo el proceso, y cada uno aporta parte de la variabilidad (y de la calidad) final del producto o servicio. Por lo que es natural enfocar los esfuerzos de mejora hacia cada uno de estos elementos de un proceso. De estos seis elementos se descartó “Medición” ya que no aplican al problema en estudio. Por tanto las “**4M**” consideradas son las siguientes: Mano de obra, Métodos

de trabajo, Materiales (Recursos) y Medio Ambiente. La figura 1.1 representa el diagrama mencionado.

### **1.1.1. Descripción de causas principales del problema**

#### **Estudiantes**

##### **Invertir tiempos prolongados en la construcción de los diagramas de flujo.**

Actualmente el tiempo promedio que un estudiante se tarda en elaborar un diagrama de flujo completo son 3 a 5 horas de acuerdo a la complejidad de la actividad evaluada ex aula. Los ejercicios realizados en clases muchas veces no se completan por el tiempo que toma ir dibujando cada figura que debe utilizarse para construir el diagrama de flujo. Los ejercicios realizados en las prácticas de laboratorio tardan casi la hora completa si se desarrolla todo el proceso solucionador de problemas, por ende, se considera que es necesario contar con una herramienta que permita facilitar el desarrollo y disminuir los tiempos de elaboración del proceso solucionador de problemas completo y por consiguiente el desarrollo de diagramas de flujo n cualquier momento.

##### **Dificultad para comprender el pseudocódigo y Dificultad de comprender el código fuente**

Los estudiantes afirmaron que tienen dificultad en la comprensión del pseudo-código, básicamente la manera de construirlo a partir del diagrama de flujo creado. La utilización de palabras para describir las acciones y los procesos que quieren explicar.

##### **Poco conocimiento de herramientas de libre distribución**

La mayor parte de los estudiantes para desarrollar los diagramas de flujo se ayudan por lo general de Microsoft Visio para disminuir el tiempo de construcción de los mismos, sin embargo desconoce que existen otras herramientas de libre distribución que podrían colaborar a desarrollar los diagramas de flujo de manera rápida y eficiente. Otra razón por la cual los estudiantes no utilizan herramientas es debido a que gran parte de las herramientas que se promocionan en la Web incurren en costos muchas veces demasiado altos para las actividades que se utilizaran y el tiempo que se utilizarán. Además a esto se le suma que estas herramientas no brindan soporte completo al proceso solucionador de problemas sino solo a la construcción de los diagramas de flujo y posiblemente la generación de pseudo-código y/o código fuente en el mejor de los casos.

## **Desinterés para aprender**

El desinterés de los estudiantes para aprender la metodología del proceso solucionador de problemas, es una de las causas principales de considerar complejo el desarrollo del mismo, sin embargo el desarrollo del proceso como tal es sencillo a pesar que el tiempo que debe invertirse para hacerlo de manera correcta.

## **Docentes**

### **Carga académica excesiva**

Cada año la cantidad de estudiantes por cada grupo teórico es superior a la capacidad de la planta docente, por ende cada docente tiene a cargo grandes cantidades de estudiantes dificultando impartir las clases. Además de esto, se dificulta la calificación de actividades evaluadas por la cantidad de estudiantes que debe evaluar cada docente. Por otro lado las prácticas de laboratorio se dificultan por el esfuerzo adicional de los docentes para dar a los estudiantes explicaciones completas del proceso solucionador de problemas.

### **Falta de herramienta de apoyo a la asignatura**

Los docentes actualmente no cuentan con una herramienta que les brinde soporte a todo el proceso solucionador de problemas, sino que deben de utilizar diversas herramientas para completar el proceso e incluso intercambiar entre herramientas para desarrollar una y otra etapa.

## **Recursos**

### **Falta de herramientas con las funcionalidades necesarias**

En el mercado no hay herramientas que brinden soporte completo a todo el proceso solucionador de problemas, sino solamente a la construcción de flujogramas o bien editores de texto para realizar las actividades del planteamiento del problema, el análisis de problemas (Definición de variables) y completar los elementos para presentar un documento completo como actividad evaluada.

### **Falta de recursos económicos**

La falta de recursos económicos debido a que dificulta la adquisición de herramientas de colaboración para desarrollar de manera práctica la metodología del proceso solucionador de problemas.

### **Falta de recursos didácticos**

La escasez de los recursos didácticos con los que se cuentan para desarrollar las clases teóricas, que van desde infraestructura hasta micrófonos, computadoras portátiles, proyectores entre otros elementos de primera necesidad para desarrollar las actividades académicas.

### **Facultad de Ingeniería y Arquitectura**

#### **Falta de modernización de equipo de laboratorio**

Hasta la fecha es todavía necesario contar con mejor y mucho más equipo informático para brindar a los estudiantes una excelente calidad de educación y formación profesional, principalmente en el área de laboratorios.

#### **Falta de actualización de curricular**

De manera directa también resulta necesario realizar una actualización de la curricular, permitiendo a los estudiantes contar con opciones de especialización, permitiendo así formar profesionales a la vanguardia de la tecnología y acorde con las exigencias y necesidades del mercado laboral.

#### **Inscripción masiva de estudiantes**

Teniendo el conocimiento de las capacidades de infraestructura, planta docente, material didáctico entre otros, se continúan inscribiendo cantidades de estudiantes superiores a las que pueden ser atendidas, tomando en cuenta dichos elementos. Ya que anualmente se inscribe en promedio 1110 estudiantes.

#### **Poco presupuesto asignado al desarrollo tecnológico**

El presupuesto asignado a la Facultad de Ingeniería es insuficiente ya que las necesidades son muchas y debido a la prioridad de ellas se deja de lado el desarrollo tecnológico de la carrera y por tanto, de los estudiantes.

## **Universidad de El Salvador**

### **Baja asignación de presupuesto para la Facultad de Ingeniería y Arquitectura**

El presupuesto asignado a la Facultad de Ingeniería resulta ser mínimo en base a las necesidades y la cantidad de estudiantes de anualmente recibe.

### **Desconocimiento del problema**

El desconocimiento de la escasez de equipos informáticos, infraestructura, docentes para dar cobertura a la cantidad de estudiantes que ingresan anualmente a la Facultad y la concentración de estudiantes en la carrera de Ingeniería de Sistemas Informáticos.

### **Lenta gestión de proyectos para adquisición de equipos informáticos**

La lentitud del proceso de gestión para la adquisición y renovación de equipos y materiales didácticos, ante la rapidez en el crecimiento de la ciudad estudiantil.

DIAGRAMA ISHIKAWA O DIAGRAMA CAUSA – EFECTO

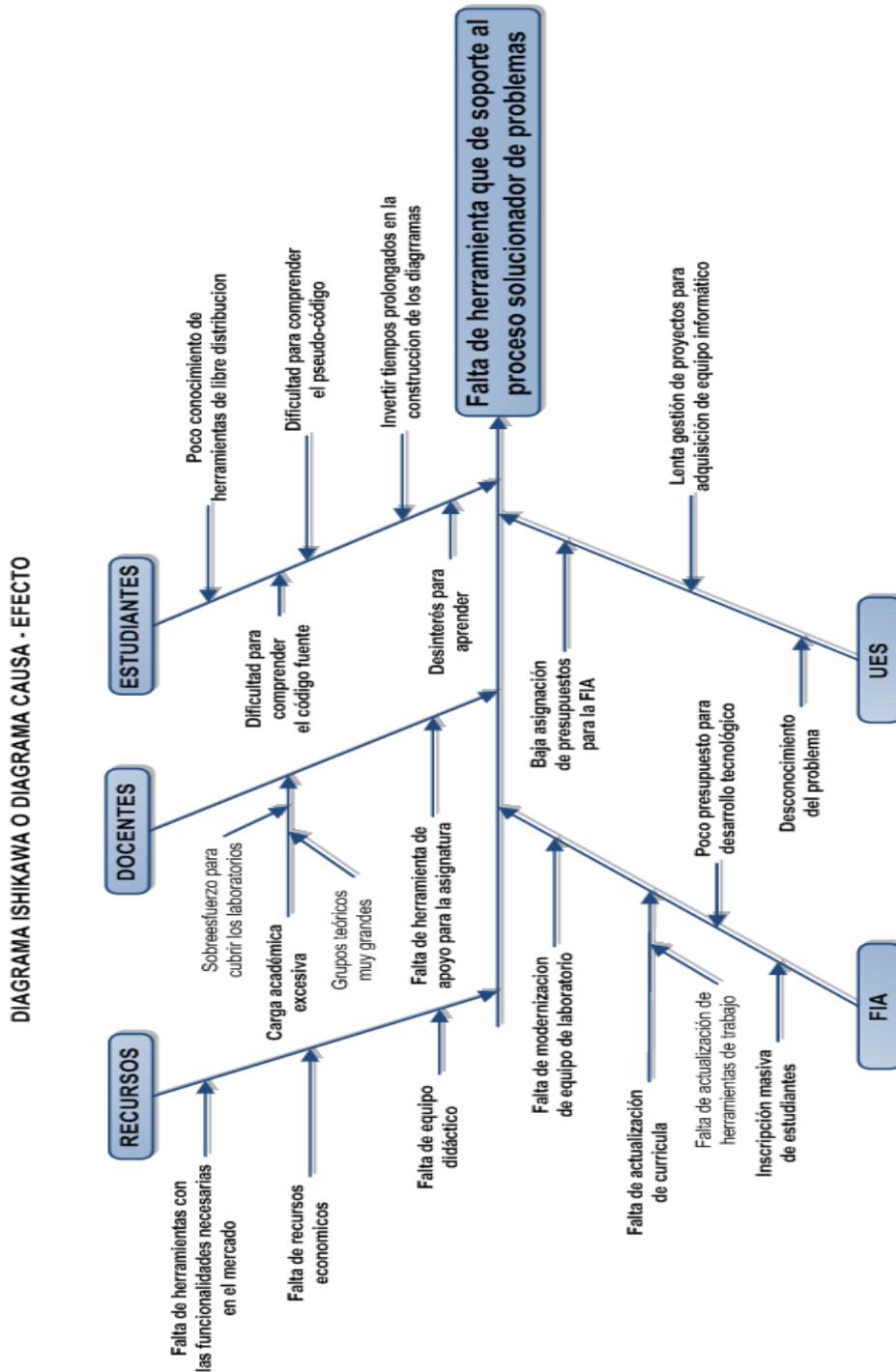


Figura 1.1 – Diagrama de Causa – Efecto o Diagrama Ishikawa

## 1.2. Definición del problema

Luego de haber identificado el problema a resolver se plantea la siguiente definición:

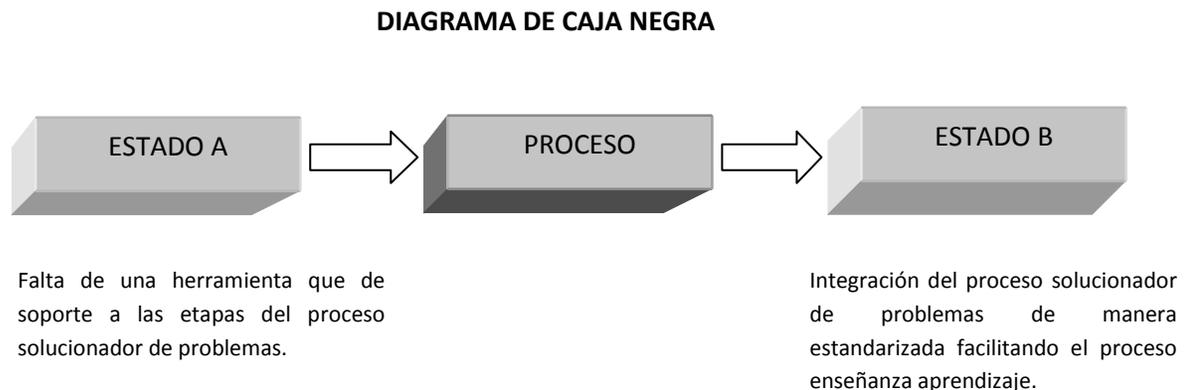
**Falta de una herramienta que de soporte al proceso solucionador de problemas y facilite la enseñanza – aprendizaje de manera estandarizada.**

## 1.3. Planteamiento del problema

Actualmente los docentes de la EISI de la Facultad de Ingeniería y Arquitectura, no cuentan con una herramienta que brinde soporte completo a la metodología del proceso solucionador de problemas. Las etapas de dicha metodología se pueden realizar con diferentes tipos de herramientas que se encuentran en el mercado, pero la utilización de dichas herramientas puede generar algún costo por el licenciamiento, pero también existen herramientas de uso gratuito.

Para plantear el problema, se hace uso de la técnica denominada *Método de la Caja Negra*<sup>18</sup>; la cual consiste en plantear un problema mediante un grafico (Ver figura 2.6), donde se representan los dos estados de una situación real, el estado inicial (Estado A), es la descripción de la situación actual a la cual se le dará solución; el estado final (Estado B) es el resultado del análisis del problema en estudio y solución.

De manera esquemática se expone con el método de la caja negra el análisis de la situación problemática en la figura 1.2



*Figura 1.2 – Diagrama de Caja Negra para el desarrollo de la herramienta de desarrollo asistido de diagramas de flujo.*

<sup>18</sup> Krick, E. (1998). Introducción a la Ingeniería y al Diseño en Ingeniería. México, D.F.: Noriega Editores.

## **1.4. Enfoque de Sistemas**

### **1.4.1. Componentes del Enfoque de sistemas<sup>19</sup>**

#### **Definiciones**

Para describir la estructura, utilizaremos el Enfoque de Sistemas, el cual brinda la capacidad de exponer con amplitud la problemática, los procesos de cambio, recursos y procesos que participan de esta, para comprender con mayor claridad y profundidad el problema, sus múltiples causas y consecuencias:

#### **Medio ambiente**

Es todo lo que está fuera de la Frontera del sistema y que interactúa con él, proveyendo de los insumos y al cual se le presentan los resultados que obtiene el sistema.

#### **Entradas**

Es cualquier elemento que se toma del ambiente y que ingresa al sistema para ser empleado o transformado en salidas de dicho sistema.

#### **Procesos**

Es el momento en el cual todos los componentes que interactúan en el sistema, se coordinan para transformar los insumos y generar productos o resultados previamente determinados. La interacción, armonía, integración, coordinación, interdependencia, etc., son fundamentales en cada etapa o fase del proceso.

#### **Salidas**

Son los productos o los resultados que egresan del sistema después de haber sido procesados.

#### **Control**

Está formada por datos y por acciones correctivas y la aplica la dirección del sistema para reorientar las acciones, o corregir los errores, en caso de que los productos o resultados (salida) no coincidan con los objetivos o metas establecidas para dicho sistema.

#### **Frontera**

Es el límite hasta donde abarca el sistema, fuera de ella está el Medio Ambiente.

---

<sup>19</sup> Gero Levaggi, Teoría general de los sistemas, UGERMAN Editor, Ciencia y Técnica.

### 1.4.2. Esquema del Enfoque de sistemas de la situación actual

La siguiente figura muestra el enfoque de sistemas de la situación actual:

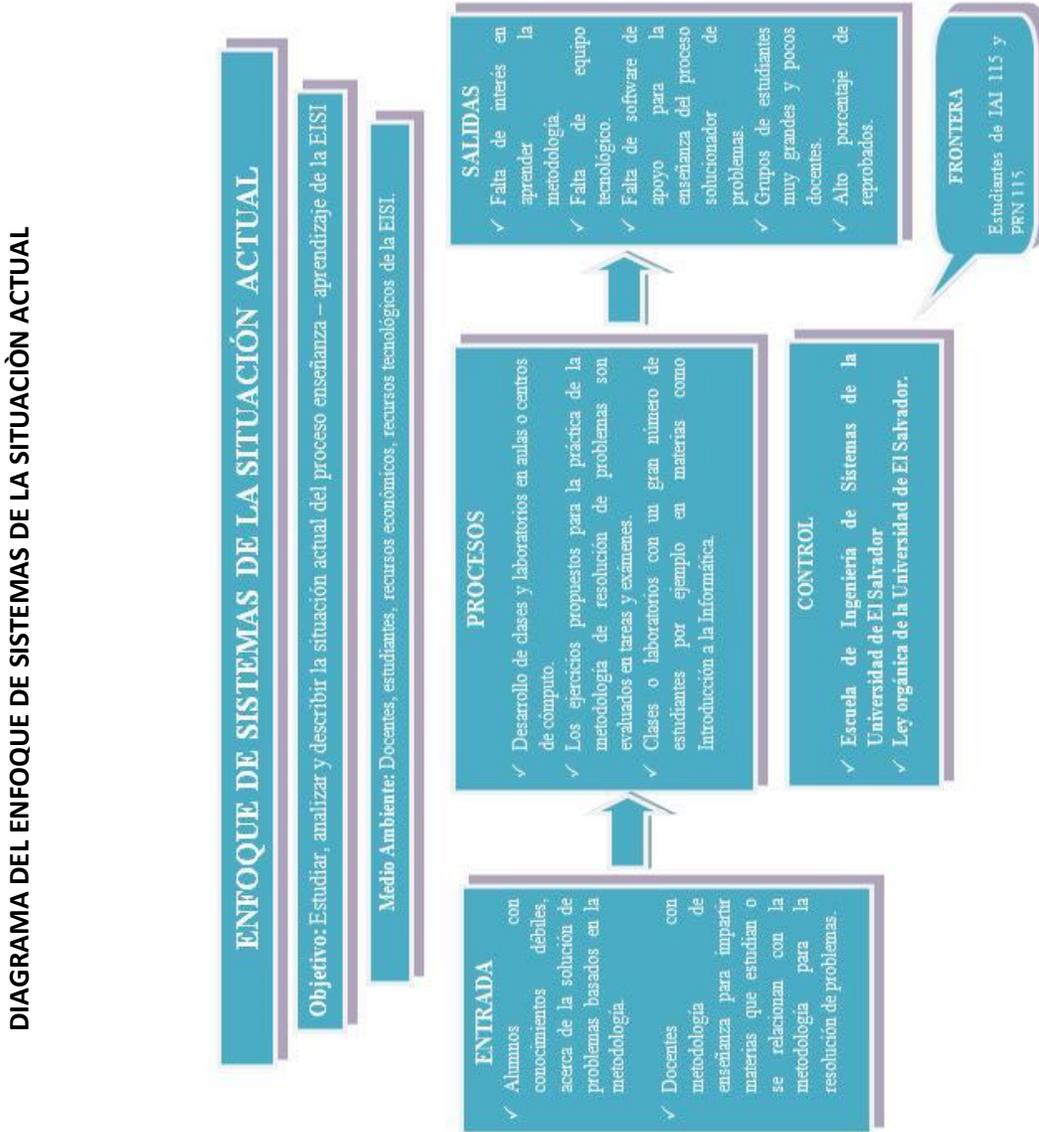


Figura 1.3 – Esquema del Enfoque de sistemas de la Situación Actual

### 1.4.3. Descripción del Enfoque de Sistemas de la Situación Actual

#### Objetivo

Estudiar, analizar y describir la situación actual del proceso enseñanza-aprendizaje de la EISI.

#### Medio\_Ambiente

- ✓ *Docentes*: Son los encargados de la orientación y enseñanza de los estudiantes de las diferentes asignaturas.
- ✓ *Estudiantes*: Grupo de personas pertenecientes a algunas asignaturas impartidas por la Escuela de Ingeniería de Sistemas Informáticos<sup>20</sup>.
- ✓ *Recursos económicos*: Es el recurso económico disponible para la adquisición de nuevos equipos informáticos, construcción de infraestructura y contratación de personal docente necesarias para cubrir la demanda de la población estudiantil de las asignaturas.
- ✓ *Recursos Tecnológicos*: Recursos necesarios para contar con nuevos y mejores centros de cómputo, que ayuden a los estudiantes a aplicar los conocimientos adquiridos en las clases teóricas.

#### Entrada

- ✓ Alumnos con conocimiento débil desde la educación básica, acerca de la solución de problemas basados en la metodología.
- ✓ Docentes con metodología de enseñanza para impartir materias que estudian o se relacionan con la metodología para la resolución de problemas.

#### Proceso

- ✓ Desarrollo de clases y laboratorios en aulas o centros de cómputo: para la mayoría de las clases expositivas hay un apoyo de cañón y laptop. Durante los laboratorios se usa únicamente el compilador de Turbo C 3.0 para desarrollar la codificación de las soluciones, el resto del proceso se realiza durante las clases expositivas o como actividad ex - aula.
- ✓ Los ejercicios propuestos para la práctica de la metodología de resolución de problemas son evaluados en tareas y exámenes: La mayor parte de las actividades de desarrollo académico que utilizan esta metodología del proceso solucionador de problemas son evaluadas.

---

<sup>20</sup> A lo largo del documento se abreviara como EISI.

- ✓ Clases o laboratorios con gran número de estudiantes por ejemplo en materias como Introducción a la Informática: Existe una gran demanda de estudiantes, principalmente para la carrera de Ingeniería de Sistemas Informáticos, y esta por ser una materia de primeros años, la cantidad de estudiantes que se necesita atender es grande, tanto para las clases teóricas como para los laboratorios.

### **Salidas**

- ✓ *Falta de interés en aprender la metodología:* Es el elemento menos controlable de las entradas, pues depende únicamente de la importancia que se le dé al aprender la metodología.
- ✓ *Falta de equipo tecnológico:* La falta del equipo informático para utilizar herramientas que permitan facilitar el proceso enseñanza aprendizaje que brinde la cobertura de la población estudiantil de las asignaturas impartidas en la EISI.
- ✓ *Falta de software de apoyo para la enseñanza del proceso solucionador de problemas:* Una de las razones principales por las cuales no se utiliza una herramienta para el desarrollo del proceso solucionador de problemas es que no hay ninguna en el mercado que brinde todas las funcionalidades necesarias para dar soporte a la metodología del proceso solucionador de problemas.
- ✓ *Grupos de estudiantes muy grandes y pocos docentes:* Debido a la gran demanda de la carrera de Ingeniería en Sistemas Informáticos y por tratarse de una asignatura de primer año, la cantidad de estudiantes que se inscriben es excesiva en cuanto a las capacidades de la planta docente para atenderla, por ende, los grupos teóricos son más grandes, más difíciles de controlar.
- ✓ *Alto porcentaje de reprobados:* En base a datos estadísticos de los últimos dos años, se conoce que el porcentaje de reprobados supera el porcentaje de aprobados en un promedio del 15%.

### **Control**

Escuela de Ingeniería de Sistemas Informáticos de la Universidad de El Salvador y Ley Orgánica de la Universidad de El Salvador.

### **Frontera**

Estudiantes de las Materias de IAI115, PRN115.

## 2. Definición de Requerimientos

Como aspecto fundamental para poder resolver cualquier problema, se definen los requerimientos a través de los cuales se determinan las necesidades de los usuarios, las cuales sirven como marco de referencia para poder desarrollar las siguientes etapas y así poder brindar una solución al problema que se está tratando.

A continuación se detallan los requerimientos, los cuales han sido divididos en 2 áreas:

- ✓ Requerimientos Funcionales: Los cuales describen los aspectos fundamentales que debe contener la aplicación.
- ✓ Requerimientos No Funcionales: Son los requerimientos que están relacionados con aspectos fuera de las funcionalidades principales de la herramienta a desarrollar.
- ✓ Requerimientos de Seguridad: Son los requerimientos que están relacionados con las funciones de auditoría para evitar la réplica de soluciones que se construyan en la herramienta EISI-FlowChart.
- ✓ Requerimientos Legales: son los que corresponden a la autorización de brindar los derechos de autor a la Universidad de El Salvador como propietario del proyecto.
- ✓ Requerimientos de Desarrollo: Son los requerimientos que tienen que ver con el grupo de trabajo, los cuales están asociados a las necesidades de hardware y software mínimos para poder construir la herramienta.
- ✓ Requerimientos de Implementación: Son los requerimientos mínimos que el usuario debe poseer tanto en Hardware como en Software para poder utilizar de forma óptima la herramienta.



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

Ciudad Universitaria, 20 de Octubre de 2009.

A quien corresponda:

Le saludo cordialmente, deseándole éxitos en sus labores profesionales.

El motivo de la presente es hacer de su conocimiento que Yo, Ing. César González, estoy participando como contraparte del proyecto de Trabajo de Graduación “*Herramienta para el desarrollo asistido de diagramas de flujo*” actualmente dirigido por el docente director Ing. William Castellanos y desarrollado por los bachilleres: Jeannette Mellany Aguilar Zepeda, Christian Alexander Martínez Arteaga, Elvis Mauricio Ramírez y Miguel Isaac Sánchez Ramos.

Por lo cual, y para los efectos que se consideren convenientes, apruebo y certifico los requerimientos para el desarrollo de la segunda etapa del proyecto y que son presentados a continuación.

**F.**

---

*Ing. César González*  
Jefe del Departamento de Programación y Manejo de Datos

## 2.1.Requerimientos Funcionales

A continuación se detallan los requerimientos funcionales de la Herramienta para el desarrollo asistido de diagramas de flujo, los cuales fueron recabados a través de encuestas realizadas a los docentes involucrados en el proceso enseñanza – aprendizaje del proceso solucionador de problemas de la Escuela de Ingeniería de Sistemas Informáticos<sup>21</sup>, los cuales fueron formulados de forma comprensible, cabe mencionar que algunos de los requerimientos fueron formulados en base a una entrevista realizada con algunos docentes.

- **Requerimiento 1:** Se debe crear un solo archivo que almacene todo lo trabajado en la aplicación EISI – FlowChart. La extensión que debe utilizar el archivo es “.prsp”
- **Requerimiento 2:** La herramienta debe manejar dos tipos de proyectos uno que permita seguir el proceso solucionador de problemas y otro que no obligue a utilizarlo.
- **Requerimiento 3:** Al iniciar la herramienta, debe mostrarse un asistente que permita seleccionar el tipo de proyecto sobre con el cual se trabajará.
- **Requerimiento 4:** La herramienta debe consultar si se utilizará modularidad o no en el proyecto.
- **Requerimiento 5:** Mostrar las opciones de menú por medio de una barra de herramientas de fácil acceso.
- **Requerimiento 6:** Las opciones de menú deben estar sincronizadas, mostrando activas las opciones disponibles de acuerdo a la actividad o tarea que se esté realizando en el área de trabajo.
- **Requerimiento 7:** Las barras de herramientas, barra de menú y barra de estado debe facilitar su ocultación o visualización.
- **Requerimiento 8:** La ventana de controles para la creación de flujogramas debe facilitar su ocultación o visualización.
- **Requerimiento 9:** La herramienta debe permitir guardar las configuraciones que el usuario establezca para:
  - Definir color de controles para creación de diagramas de flujo y color de interfaz.

---

<sup>21</sup> Ver encuesta en Anexo 5 – Encuesta para docentes universitarios

- **Requerimiento 10:** Permitir la ampliación/reducción de la vista del entorno de trabajo en cualquiera de los pasos del proceso solucionador de problemas
- **Requerimiento 11:** Cada vez que se crea un proyecto nuevo, debe abrirse una pestaña diferente.
- **Requerimiento 12:** Mostrar cada etapa del proceso solucionador de problemas utilizando pestañas para cada una.
- **Requerimiento 13:** El tipo de proyecto que utilice el proceso solucionador de problemas, únicamente permitirá pasar a la siguiente etapa (pestaña) cuando se haya definido como mínimo un elemento de cada una de las siguientes etapas o pasos del proceso solucionador de problemas:
  - En el planteamiento del problema se deberá describir al menos una frase como entrada y una frase como salida.
  - En el análisis de problema se debe definir como mínimo una variable de salida.
  - En el diagrama de flujo se deberá crear por lo menos el inicio, una figura de entrada, una figura de salida y el fin del diagrama con los flujos correspondientes.
- **Requerimiento 13:** La aplicación debe permitir utilizar modularidad. Los módulos pueden desarrollarse de manera independiente como proyectos diferentes, aplicando para cada uno el proceso solucionador de problemas. Solamente los proyectos principales podrán llamar a submódulos, es decir, existirá un único nivel de anidamiento para submódulos.
- **Requerimiento 14:** Cada vez que se abra un proyecto debe abrirse con las modificaciones guardadas la última vez que estas se modificaron.
- **Requerimiento 15:** La aplicación solamente soportará la programación estructurada.
- **Requerimiento 16:** La herramienta debe permitir copiar, pegar y cortar elementos en el entorno de trabajo.
- **Requerimiento 17:** La herramienta debe permitir la conexión de impresoras.
- **Requerimiento 18:** Se creará un archivo de configuración, en el cual se agregarán los valores predeterminados para tamaño de fuente, tipo de fuente, color de fuente, tamaño mínimo y máximo de figuras, tiempo de guardado automático entre otros.
- **Requerimiento 19:** Crear un documento de texto con extensión RTF, para que pueda ser leído a partir de cualquier plataforma.

### Planteamiento del problema

- **Requerimiento 1:** En el planteamiento del problema, se definirán las entradas y salidas utilizando líneas, tipo tabla, para definir cada una de ellas con frases cortas, las cuales tendrán un mínimo de 10 caracteres y un máximo de 255 caracteres.
- **Requerimiento 2:** En el planteamiento del problema, deben existir breves descripciones que indiquen el significado de las entradas y salidas en el diagrama de Caja Negra cuando el cursor pase sobre las etiquetas Entradas y Salidas.

### Análisis del problema

- **Requerimiento 1:** En el análisis del problema, se debe definir cada variable de manera secuencial, en una especie de tabla en la cual se definan los atributos de cada una.
- **Requerimiento 2:** En el análisis del problema, de acuerdo al tipo de proyecto seleccionado (con modularidad o sin modularidad) se deben definir para cada variable: Nombre, Tipo de Dato, Longitud, Dimensionamiento, Filas, Columnas y Descripción.
- **Requerimiento 3:** En el análisis del problema, para definir las restricciones es posible establecer condiciones e incluso expresiones lógicas complejas.
- **Requerimiento 4:** En el análisis del problema, es posible asignar valores a los procesos los valores e incluso permitir evaluar expresiones lógicas y matemáticas.
- **Requerimiento 5:** En el análisis del problema, es posible asignar los tipos de datos siguientes: Carácter, Real (simple y doble precisión), Entero (simple y doble precisión), Cadena de caracteres y Boolean como mínimo.
- **Requerimiento 6:** En el análisis del problema se dará soporte a modularidad, permitiendo agregar y quitar módulos a un proyecto principal. Existirá un único nivel de anidamiento para módulos.
- **Requerimiento 7:** En el análisis del problema, los módulos que se creen de manera independiente pueden agregarse o quitarse dentro de otros proyectos.
- **Requerimiento 8:** En el análisis del problema, para aplicar modularidad, se debe crear primero un proyecto principal para agregar los módulos que se necesiten.

- **Requerimiento 9:** En el análisis de variables, al utilizar modularidad, el usuario debe indicar si cada variable de entrada será un parámetro de entrada o una variable local.
- **Requerimiento 10:** En el análisis de variables, al utilizar modularidad, el usuario debe indicar si se tendrán valores de retorno para cada módulo, permitiendo seleccionar si el módulo es una función o un procedimiento.

### Construcción de flujograma

- **Requerimiento 1:** El entorno de desarrollo de flujogramas debe mostrar páginas secuenciales.
- **Requerimiento 2:** El entorno de desarrollo de flujogramas mostrará líneas guías, estas líneas pueden visualizarse o no según lo requiera el usuario.
- **Requerimiento 3:** Al iniciar la construcción del diagrama de flujo, se mostrará por defecto la figura de inicio y fin del flujograma.
- **Requerimiento 4:** En la construcción del diagrama de flujo, para agregar una figura, se debe permitir arrastrar la figura y ubicarla en alguna posición específica del área de trabajo.
- **Requerimiento 5:** En la construcción del diagrama de flujo, cuando se requiera agregar una figura a un flujograma finalizado, será necesario reconectar la nueva figura en la ubicación que se considere adecuada.
- **Requerimiento 6:** En la construcción del diagrama de flujo, se permitirá cambiar figuras seleccionándola y utilizando el menú contextual con la opción *cambiar*, mostrando todas las figuras disponibles.
- **Requerimiento 7:** En la construcción del diagrama de flujo para eliminar cualquier figura dentro del área de trabajo se debe presionar la tecla Suprimir (Delete).
- **Requerimiento 8:** En la construcción del diagrama de flujo, cuando se elimine una figura que ya tenga conexiones con otras se eliminarán automáticamente la figura y las conexiones que tenga asociadas.
- **Requerimiento 9:** En la construcción del diagrama de flujo, cuando se elimine una figura que no tenga conexiones, solo será necesario seleccionarla y eliminarla del área de trabajo.
- **Requerimiento 10:** En la construcción del diagrama de flujo, la ubicación de la primera figura, será la línea guía para las subsiguientes figuras.

- **Requerimiento 11:** En la construcción del diagrama de flujo, cada figura contará con cuatro puntos de anclaje, ubicando uno en cada lado de la figura.
- **Requerimiento 12:** Los puntos de anclaje de cada figura serán visibles cuando el cursor se acerquen a dichos puntos.
- **Requerimiento 13:** En la construcción del diagrama de flujo, se utilizará como punto de anclaje por defecto el que está ubicado en la parte superior de cada figura.
- **Requerimiento 14:** En la construcción del diagrama de flujo, al trazar las líneas de conexión únicamente se dibujaran en manera rectangular (trazando líneas horizontales y verticales), independientemente de la forma de la traza.
- **Requerimiento 15:** En la construcción del diagrama de flujo, se permitirá la utilización de los ciclos estructurados Hasta que y Mientras. No se permitirá la utilización del ciclo no estructurado Hacer - Mientras.
- **Requerimiento 16:** En la construcción del diagrama de flujo, las figuras de los ciclos repetitivos Hasta que y Mientras se permitirán arrastrar al área de trabajo como estructuras indicando las conexiones que debe tener cada bucle.
- **Requerimiento 17:** En la construcción del diagrama de flujo, los flujos podrán dibujarse hacia abajo, hacia la derecha y hacia la izquierda. Se permitirán trazas hacia arriba, cuando se utilicen los ciclos Hasta que y Mientras.
- **Requerimiento 18:** En la construcción del diagrama de flujo, al momento de colocar una figura, se debe mostrar el listado de variables, restricciones, procesos, condiciones o módulos que se pueden asociar a cada figura que el usuario utilice.
- **Requerimiento 19:** En la construcción del diagrama de flujo, solamente se podrán utilizar las variables previamente definidas en el análisis de variables.
- **Requerimiento 20:** En la construcción del diagrama de flujo, se utilizarán las figuras especificadas en el anexo 5, tabla 11.1.
- **Requerimiento 21:** En la construcción del diagrama de flujo, las figuras tendrán un tamaño normal y mínimo de 70 píxeles de alto por 70 píxeles de ancho. Las figuras tendrán un tamaño máximo de 470 píxeles de alto por 225 de ancho.
- **Requerimiento 22:** En la construcción del diagrama de flujo, se permitirán cuatro niveles de anidamiento de estructuras, ya sean estructuras repetitivas o selectivas.
- **Requerimiento 23:** En la construcción del diagrama de flujo, para las figuras de decisión (selección doble o rombo), la parte verdadera se ubicará siempre al lado derecho y la parte falsa al lado izquierdo de la figura.

- **Requerimiento 24:** En la construcción del diagrama de flujo, para las figuras de ciclos repetitivos, la parte verdadera (entrada al ciclo) se ubicará siempre en la parte de abajo, el retorno del ciclo se ubicará al lado izquierdo y la parte falsa (salida del ciclo) al lado derecho del ciclo.
- **Requerimiento 25:** En la construcción del diagrama de flujo, para las figuras selectivas múltiples, se agregarán procesos al lado derecho de la línea trazada hacia abajo y las salidas serán siempre en el lado derecho de la última figura del proceso definido.
- **Requerimiento 26:** En la construcción del diagrama de flujo, se permitirá configurar la fuente del texto dentro de cada figura (tipo, tamaño, color). Los cambios realizados a la configuración de la fuente se aplicarán a todas las figuras del diagrama de flujo en el área de trabajo.
- **Requerimiento 27:** En la construcción del diagrama de flujo, se utilizará de manera predeterminada el tipo de letra Times New Roman con 12 puntos de tamaño.
- **Requerimiento 28:** En la construcción del diagrama de flujo, se utilizará como tamaño máximo 14 puntos, como mínimo 8 puntos y como tamaño predeterminado 12 puntos.
- **Requerimiento 29:** En la construcción del diagrama de flujo, no debe permitirse editar texto dentro de las figuras, sino solamente aplicar las expresiones previamente establecidas en la fase de análisis del problema.
- **Requerimiento 30:** En la construcción del diagrama de flujo debe permitirse el redimensionamiento de las figuras a utilizar.
- **Requerimiento 31:** En la construcción del diagrama de flujo, cada figura contará con ocho puntos de expansión.
- **Requerimiento 32:** Los márgenes para cada página no deben ser mayores de 2.5 centímetros.
- **Requerimiento 33:** En la construcción del diagrama de flujo, cuando el espacio para colocar una figura toque algún margen se lanzará un mensaje de advertencia indicando que es necesario colocar un conector de página o conector de hoja.
- **Requerimiento 34:** En la construcción del diagrama de flujo, se debe crear de manera automática dos conectores dentro de la misma página o dos conectores en diferentes páginas.
- **Requerimiento 35:** La numeración de los conectores se realizará de manera automática.

- **Requerimiento 36:** En la construcción del diagrama de flujo, para aplicar modularidad se agregará la figura y seleccionando el nombre del módulo correspondiente. Se utilizarán parámetros pasados por valor cuando se utilice modularidad.
- **Requerimiento 37:** En la construcción del diagrama de flujo, al agregar un módulo, se ubicarán los nombres y tipos de datos correspondientes a los parámetros establecidos en el análisis del problema. Además se mostrará el prefijo correspondiente para distinguirlos entre funciones y procedimientos.
- **Requerimiento 38:** En la construcción del diagrama flujo, se contará con una ventana de Exploración de Sub-módulos para facilitar la navegación sobre los módulo.
- **Requerimiento 39:** En la construcción del diagrama de flujo, para visualizar el diagrama de flujo de un módulo, es necesario buscar en la ventana del *Explorador de submódulos* el correspondiente diagrama de flujo.
- **Requerimiento 40:** En la construcción del diagrama de flujo, se permitirá compilar el diagrama de flujo.
- **Requerimiento 41:** En la construcción del diagrama de flujo, al compilar, únicamente se verificará que existan al menos una figura de inicio, una figura de entrada, una de proceso, una de salida y un fin con sus flujos correspondientes.
- **Requerimiento 42:** Los ciclos repetitivos se deben considerar como una estructura completa, es decir, debe contar con las figuras, los flujos de entrada y los flujos de cierres correspondientes a los ciclos Mientras y Hasta que.

### Ejecución del flujograma

- **Requerimiento 1:** Se debe solicitar al usuario el ingreso de los valores para las variables de entrada (definidas en el análisis del problema) a medida se avance en la ejecución del diagrama de flujo.
- **Requerimiento 2:** El ingreso de variables se debe realizar utilizando un asistente que solicite cada variable.
- **Requerimiento 3:** Si se solicitan múltiples variables en una misma figura de lectura, se debe activar una ventana cada vez que permita ingresar el valor de cada variable.
- **Requerimiento 4:** Se debe validar que los valores ingresados para las variables coincidan con los tipos de datos definidos en el análisis de variables.

- **Requerimiento 5:** Se debe advertir al usuario cuando el ciclo repetitivo Mientras no cuente con una condición de salida definida en la estructura correspondiente.
- **Requerimiento 6:** Se debe advertir al usuario cuando el ciclo repetitivo Hasta que, tenga definidos valores de cambio o valores de salida incongruentes.
- **Requerimiento 7:** Cuando se ejecuten módulos, se solicitará al usuario el ingreso de los valores de las variables locales correspondientes a cada módulo.
- **Requerimiento 8:** Mostrar en una ventana los resultados de los valores de las variables y sus cambios durante la ejecución.
- **Requerimiento 9:** Mostrar en una ventana las excepciones ocurridas durante la ejecución de un flujograma. Es decir, errores no producidos por datos de entrada. Las excepciones pueden ser divisiones por cero, ciclos infinitos.
- **Requerimiento 10:** La herramienta no debe validar la lógica que el estudiante utilice para la creación del diagrama de flujo, sino únicamente que exista secuencia lógica en las figuras y en los cierres de las mismas.
- **Requerimiento 11:** Se debe cancelar la ejecución de un flujograma cuando se tarde más de treinta segundos en finalizar, no obstante, debe permitir al usuario configurar este tiempo.

### Generación de pseudocódigo

- **Requerimiento 1:** Debe existir como mínimo en el diagrama de flujo: un inicio, una instrucción de entrada, una instrucción de salida y un fin.
- **Requerimiento 2:** No deben existir errores en la ejecución del diagrama de flujo.
- **Requerimiento 3:** No se permite editar el pseudocódigo dentro de la herramienta.
- **Requerimiento 4:** El pseudocódigo se debe mostrar como salida en pantalla y como salida en archivo de texto plano (archivo con extensión .txt).
- **Requerimiento 5:** Cada línea del pseudocódigo debe tener un número que la identifique.
- **Requerimiento 6:** Las palabras reservadas se deben utilizar en idioma Español.
- **Requerimiento 7:** Las palabras reservadas deben resaltar dentro del pseudocódigo que se construya marcándolas con color azul.
- **Requerimiento 8:** Las palabras reservadas deben utilizar inicial mayúscula como parte de su formato.

- **Requerimiento 9:** Indicar el nombre del pseudocódigo al inicio de este.
- **Requerimiento 10:** El nombre del pseudocódigo no debe estar compuesto por palabras separadas por espacios en blanco.
- **Requerimiento 11:** El nombre del pseudocódigo debe ser alfanumérico y representativo de la funcionalidad.
- **Requerimiento 12:** Las palabras reservadas no deben distinguirse entre mayúsculas y minúsculas.
- **Requerimiento 13:** Cada instrucción debe ir separada por un salto de línea.
- **Requerimiento 14:** La interpretación debe realizarse sobre todo el pseudocódigo cargado sin hacer excepciones. En caso de existir errores, estos se deben mostrar en pantalla.
- **Requerimiento 15:** Los bloques de instrucciones de pseudocódigo deben estar delimitados con palabras reservadas que indiquen el inicio y final del bloque.
- **Requerimiento 16:** Todo texto que no sea una palabra reservada debe colocarse entre comillas dobles.
- **Requerimiento 17:** La ejecución del pseudocódigo se trabajará en modo consola.
- **Requerimiento 18:** Se desplegara como resultado un archivo de texto que muestre el pseudocódigo del programa principal.

### Generación de código fuente

- **Requerimiento 1:** No deben existir errores durante en la ejecución del diagrama de flujo.
- **Requerimiento 2:** Se permite generar el código fuente en un orden independiente, es decir, antes o después de generar el pseudocódigo.
- **Requerimiento 3:** Se debe crear un archivo con la extensión correspondiente a cada tipo de lenguaje de programación (.cpp o .java).
- **Requerimiento 4:** Se debe permitir guardar el archivo en la ubicación que el usuario indique, sin embargo, de manera predeterminada se debe ubicar en la carpeta Proyectos de PrSP.
- **Requerimiento 5:** Se debe permitir renombrar el archivo generado con el código fuente.
- **Requerimiento 6:** Se debe mostrar el código fuente en una pantalla adicional.

- **Requerimiento 7:** El código que se genere se debe ejecutar en modo consola desde los compiladores correspondientes a cada tipo de lenguaje de programación, Turbo C o Java, es decir, no se debe permitir la ejecución en ambiente gráfico.
- **Requerimiento 8:** La instrucción break se debe utilizar como parte de un selector de caso.
- **Requerimiento 9:** Se debe respetar la sintaxis y reglas de programación de cada lenguaje de programación, turbo C y Java.
- **Requerimiento 10:** La codificación de los módulos utilizados se mostrara al inicio del archivo, es decir, antes del cuerpo del programa principal.

### Generación de versiones de la herramienta

- **Requerimiento 1:** Se debe utilizar un asistente que facilite la generación del paquete de distribución que se desea construir.
- **Requerimiento 2:** El asistente debe contar con 3 pasos para generar el paquete de distribución.
- **Requerimiento 3:** Se deben tener disponibles dos tipos de versiones:
  - *Versión básica:* el cual no podrá generar versiones y además no podrá validar ningún tipo de regla en el proceso solucionador de problemas. Las opciones disponibles para esta versión son:
    - Validación de definición de variables de entrada y variables de salida en el análisis del problema
    - Ejecución de flujograma
    - Interpretación de pseudocódigo
  - *Versión completa* que contenga todos los elementos del proceso solucionador de problemas y el modulo de generación de versiones.
- **Requerimiento 4:** Se debe permitir guardar el paquete de distribución en la ubicación que el usuario decida, sin embargo, de manera predeterminada se guardará en el mismo directorio de carpetas de los proyectos.

### Generar flujograma a partir de pseudocódigo

- **Requerimiento 1:** Se debe cargar a la herramienta un documento de texto plano sin formato para realizar la conversión.
- **Requerimiento 2:** No se debe digitar el pseudocódigo dentro de la herramienta.

- **Requerimiento 3:** Se permite validar el pseudocódigo cargado en la herramienta.
- **Requerimiento 4:** Es necesario que la compilación sea exitosa, caso contrario no se generará el flujograma.
- **Requerimiento 5:** La compilación del pseudocódigo se realizará en modo gráfico.
- **Requerimiento 6:** Se enviarán mensajes a pantalla en caso que existan errores en la compilación del pseudocódigo.

## 2.2.Requerimientos No Funcionales

A continuación se detallan los requerimientos no funcionales de la herramienta para el desarrollo asistido de diagramas de flujo, los cuales fueron recabados a través de una encuesta (Ver detalle en Anexo 1) realizada a los docentes de la cátedra de Introducción a la Informática de la Escuela de Ingeniería de Sistemas Informáticos, los cuales fueron formulados de forma comprensible.

- **Requerimiento 1:** La barra de herramientas debe estar ubicada en la parte superior de la pantalla.
- **Requerimiento 2:** Se debe contar con barra de estado, barra de herramientas.
- **Requerimiento 3:** Se debe contar con ventanas para Exploración de proyecto, Debugg.
- **Requerimiento 4:** Se debe almacenar la información de cada etapa en un mismo archivo.
- **Requerimiento 5:** Para el planteamiento del problema, la definición de entradas y salidas son descriptivos.
- **Requerimiento 6:** La interfaz gráfica debe tener un color fijo predeterminado.
- **Requerimiento 7:** Los colores de los controles que se utilizaran para crear los diagramas de flujo deben tener un color fijo predeterminado.
- **Requerimiento 8:** Se contará con una herramienta que permita manejar el zoom o ampliación/reducción del contenido en el área de trabajo.
- **Requerimiento 9:** En el planteamiento del problema, no debe existir cantidad máxima de variables a definir.
- **Requerimiento 10:** En el análisis de variables, en la definición de variables, no debe existir cantidad máxima de variables a definir.

- **Requerimiento 11:** En el análisis de variables, en las Restricciones, no debe existir límite de expresiones.
- **Requerimiento 12:** La ventana de ejecución debe permanecer visible únicamente cuando se trabaje en la etapa de diseño de la solución (construcción de flujograma).
- **Requerimiento 13:** Para guardar el proyecto se debe utilizar el formato de archivo XML.

### 2.3.Requerimientos de seguridad

A continuación se detallan los requerimientos de seguridad con los que contará EISI – FlowChart para controlar la réplica de ejercicios o actividades evaluadas entre los estudiantes.

- **Requerimiento 1:** Para todos los proyectos, se debe crear un archivo de seguridad no editable, en el cuál se incluya:
  - Nombre de usuario
  - Nombre de equipo
  - Fecha de creación de archivo
  - Fecha de última modificación.
  - Tipo de versión en la que se trabajo el proyecto.
- **Requerimiento 2:** El archivo con extensión .prsp generado por esta herramienta, debe estar encriptado, impidiendo así la lectura y escritura del mismo en otras aplicaciones. Para todos los proyectos, se debe crear un archivo de seguridad no editable, en el cuál se incluya:
  1. Nombre de usuario
  2. Nombre de equipo
  3. Fecha de creación de archivo
  4. Fecha de última modificación.
  5. Tipo de versión en la que se trabajo el proyecto.

## 2.4.Requerimientos legales.

La herramienta desarrollada fue realizada con el objetivo de que los integrantes del equipo de trabajo obtengan el título de Ingeniero de Sistemas Informáticos. Por esta razón su desarrollo se encuentra regulado por algunas leyes y reglamentos aplicables. A continuación se describe el impacto de estas regulaciones en el proyecto.

### **Ley de fomento y protección a la propiedad intelectual**

EISI-FlowChart estará en base a los Derechos de Propiedad Intelectual, provenientes de la legislación de El Salvador, bajo la Ley de Fomento y Protección a la Propiedad Intelectual (Decreto No 604)<sup>22</sup>.

### **Reglamento general de procesos de trabajos de graduación de la Universidad de El Salvador.**

La Universidad de El Salvador como propietario legal del EISI-FlowChart será el ente responsable de en entregar a solicitud del interesado la licencia de uso, por medio de Facultad de Ingeniería y Arquitectura, ya que de acuerdo al artículo 29 del capítulo V del mencionado reglamento se indica que los derechos de autor para los trabajos de graduación, serán de propiedad exclusiva de la Universidad de El Salvador, pudiendo disponer de los mismos, de conformidad a su marco jurídico interno y legislación aplicable.

## 2.5.Requerimientos de Desarrollo

Para definir los requerimientos de desarrollo se han dividido en requerimientos de software y requerimientos de hardware, para las cuales se han realizado sus respectivas evaluaciones a fin de seleccionar el más adecuado, a continuación se detallan cada uno de estos elementos.

La selección del lenguaje de programación se realizó en base a los siguientes criterios de evaluación, resultando Java como el lenguaje más conveniente para el desarrollo del proyecto.

### **2.5.1. Software de desarrollo**

A continuación se explican los criterios y porcentajes de evaluación para la elección del lenguaje de programación a utilizar para el desarrollo del software.

- ✓ Multiplataforma: El fuerte del software a construir es que se pueda ejecutar en diferentes Sistemas Operativos y esto se puede conseguir con un lenguaje que posea esta característica.

---

<sup>22</sup>Información obtenida a través del sitio

[http://www.sice.oas.org/int\\_prop/nat\\_leg/el\\_salvador/D604as.asp#C2set](http://www.sice.oas.org/int_prop/nat_leg/el_salvador/D604as.asp#C2set)

- ✓ Orientado a objetos: Esto permitirá un mejor diseño debido a las ventajas que proporciona este estilo de programación.
- ✓ Licencia Pública General: Esta característica es importante ya que permite que el incremento en el costo total del proyecto no se incremente por la adquisición de licencias para la utilización del lenguaje de programación.
- ✓ Interfaz amigable de desarrollo: esto permitirá que el desarrollo se agilice, ya que facilita el diseño en la parte visual del software a construir.
- ✓ Soporte en línea: Para el desarrollo del software se debe de contar con cierta documentación de soporte sobre la arquitectura del lenguaje de programación, esto es de gran importancia ya que facilita la búsqueda de temas específicos a utilizar.
- ✓ Experiencia: Considerando los 3 meses estimados para el desarrollo software a construir, el aprendizaje de un lenguaje de programación no es viable, ya que implica cierto tiempo para conocer la arquitectura y sintaxis de dicho lenguaje y esto requiere la necesidad de tiempo con el cual se acortaría el tiempo de desarrollo del software. Debido a lo anterior la experiencia es necesaria en el lenguaje a utilizar.

En la Tabla 2.1 se explica el significado de los pesos definidos para la evaluación de cada uno de los criterios con los que se compararan los lenguajes de programación elegidos.

Atributo	Valor nominal	Descripción
Malo	0	El lenguaje de programación no cumple con el criterio a evaluar.
Regular	1	El criterio a evaluar es deficiente en el lenguaje de programación.
Bueno	2	El lenguaje de programación cumple con los requerimientos mínimos en el criterio a evaluar.
Muy Bueno	3	El lenguaje de programación cumple satisfactoriamente con el criterio a evaluar.

*Tabla 2.1 – Definición de pesos para la evaluación de los lenguajes de programación.*

El resultado de las evaluaciones de los criterios de comparación de los lenguajes programación se reflejan en la tabla 2.2

		Lenguajes de programación				
		Python	Java	Ruby	C++	VB.Net
te	Multiplataforma	3	3	3	3	0
	Orientado a Objetos	3	3	3	3	3

Licencia Pública General	3	3	3	3	0
Interfaz amigable de desarrollo	2	3	3	1	3
Soporte en línea	3	3	3	3	3
Experiencia del grupo	0	2	0	2	3
<b>Total</b>	14	17	15	15	12

Tabla 2.2 – Criterios de selección del Lenguaje de Programación.

En base a la evaluación de los lenguajes y la puntuación obtenida, según los datos de la tabla 5.2, el lenguaje Java es la herramienta que obtiene la mayor puntuación y debido a esto, es el lenguaje a utilizar para el desarrollo del software.

### 2.5.2. Selección del entorno de desarrollo.

Las comparaciones se realizaron con 5 de herramientas de desarrollo utilizadas por la comunidad de Java, haciendo uso de los siguientes criterios:

- ✓ Licencia Pública General: Con este criterio se pretende conseguir que el costo total del proyecto desarrollo no se incremente por la adquisición de licencia para la utilización de la herramienta de desarrollo a utilizar.
- ✓ Multiplataforma: Debido a que el software que se desarrollará por modelos, 2 maquinas del equipo informático del grupo de trabajo estarán trabajando bajo el sistema operativo Linux, y por lo tanto la herramienta de desarrollo debe tener soporte para poder trabajar ya sea en el sistema operativo Windows o Linux.
- ✓ Experiencia: La experiencia en la utilización de la herramienta de desarrollo es necesaria para optimizar el tiempo de desarrollo y no invertir tiempo en el aprendizaje de una nueva herramienta

En la Tabla 2.3 se explica el significado de los pesos definidos para la evaluación de cada uno de los criterios con los que se compararan los entornos de desarrollo integrados.

Atributo	Valor nominal	Descripción
Malo	0	La herramienta de desarrollo no cumple con el criterio a evaluar.
Regular	1	El criterio a evaluar es deficiente en la herramienta de desarrollo.
Bueno	2	La herramienta de desarrollo cumple con los requerimientos mínimos en el criterio a evaluar.
Muy bueno	3	La herramienta de desarrollo cumple satisfactoriamente con el criterio a evaluar.

Tabla 2.3 - Definición de pesos para la evaluación de los entornos de desarrollo integrado.

En la tabla 2.4 se detalla los resultados de la evaluación de las 5 herramientas de desarrollo.

		Entorno de Desarrollo Integrado				
		NetBeans	Eclipse	JDeveloper	JBuilder	IntelliJ IDEA
Elemento	Licencia Pública General	3	3	2	0	0
	Multiplataforma	3	3	3	3	3
	Experiencia	2	1	2	0	0
Total		8	7	7	3	3

Tabla 2.4 - Criterios de selección del Entorno de Desarrollo Integrado.

En base a los resultados de la tabla 2.4 el entorno de desarrollo integrado a utilizar es NetBeans ya que fue el que obtuvo la mayor puntuación en base a los criterios de evaluación.

En los equipos de desarrollo se instalará el paquete Java SE y NetBeans (JDK 5.0 Update 18 y NetBeans 6.5.1) el cual requiere los mismos recursos de hardware para la operatividad de los dos software.

### 2.5.3. Requerimientos mínimos para la instalación del paquete Java SE y NetBeans

En la tabla 2.5 se detallan los requerimientos mínimos para la instalación del paquete Java SE y NetBeans (JDK 5.0 Update 18 y NetBeans 6.5.1).

		Requerimiento mínimo por Sistema Operativo (32 bits)	
		Windows XP	Linux/Solaris
Elemento	RAM	512MB	512MB
	Espacio en Disco duro	750MB	650MB
	Velocidad del procesador	800MHz	450Mhz

Tabla 2.5 - Requerimiento mínimo de hardware para la instalación del paquete Java y NetBeans (JDK 5.0 Update 18 y NetBeans 6.5.1)<sup>23</sup>

<sup>23</sup> Java. Download J2SE5.0, Recuperado el 26 de mayo de 2009, de [http://java.sun.com/j2se/1.5.0/install\\_jdk1.5\\_18-nb6.5.1.html](http://java.sun.com/j2se/1.5.0/install_jdk1.5_18-nb6.5.1.html)

**2.5.3.1. Hardware de desarrollo**

La información que contiene la tabla 2.6 proporciona las características generales de los 4 equipos con los que se cuenta para el desarrollo del proyecto.

		Equipos			
		Equipo 1	Equipo 2	Equipo 3	Equipo 4
<b>Elemento</b>	Tipo y Modelo	Laptop, Toshiba Tecra A8 S8513	Laptop, HP Pavilion zv6000	Desktop, Clon	Desktop, Clon
	Procesador	Intel Core 2 Duo T5600	AMD Athlon 64	Celeron Dual Core	Intel Dual Core
	Velocidad de procesador	1.83GHz	3200+ (2GHz)	2.0GHz	2.4GHz
	Memoria física	1GB	1GB	1GB	1GB
	Disco duro	100GB, 5400rpm	80GB, 5400rpm	80GB, 7200rpm	80GB, 7200rpm
	Unidad de disco	CD/CD-RW/DVD-RW	CD/CD-RW/DVD-RW	CD/CD-RW/DVD-RW	CD/CD-RW/DVD-RW
	Sistema operativo	Windows XP, profesional SP2	Windows XP, profesional SP2	Windows XP, profesional SP2	Windows XP profesional SP2

*Tabla 2.6 - Descripción del Hardware*

Estos equipos están conectados por medio de un Wireless Router TP-Link TL-WR542G7 con 4 puertos el cual provee de conexión a Internet de forma alámbrica e inalámbrica.

Además se cuenta con un impresor Canon IP 1800, el cual servirá para imprimir la documentación generada del proyecto.

**2.6.Requerimientos de Implementación**

También para determinar los requerimientos de implementación se hizo una evaluación de software y hardware de implementación, los cuales se detallan a continuación:

**2.6.1. Software de implementación**

La maquina virtual de Java (JVM por sus siglas en ingles) proporciona la facilidad de ejecución del software a desarrollar en diferentes Sistemas Operativos, lo que permitirá fácilmente la distribución de la herramienta a construir. Con esto, cualquier estudiante que reciba el software no se deberá de preocupar por el Sistema Operativo que posea su equipo de trabajo.

En la tabla 2.7 se detallan los requerimientos mínimos para la instalación de la Maquina virtual de Java.

Elemento	Requerimiento mínimo			
	Windows 2000, XP, Vista, Windows 2003	Linux	Mac OS X	Solaris
RAM	64MB	64MB	Utilizar la función de <b>actualización de software</b> (disponible en el menú de Apple).	64MB
Disco Duro	98MB	58MB		53MB
Versión	6.0	6.0		5.0

Tabla 2.7- Requerimiento mínimo de hardware para la instalación de la JVM<sup>24</sup>

### 2.6.2. Hardware de implementación

Para determinar el hardware requerido para la implementación del software, se considero los requerimientos mínimos para el sistema operativo y la maquina virtual de java, ambos serán el software mínimo para el buen funcionamiento de la herramienta a construir.

La tabla 2.8 contiene la información de los requerimientos de hardware de los sistemas operativos con soporte de Java.

		Requerimientos mínimos por Sistema Operativo			
		Windows XP Pro <sup>25</sup>	Linux, Distribución Ubuntu 8.04 <sup>26</sup>	Mac OS X <sup>27</sup>	Solaris 10 <sup>28</sup>
Elemento	Microprocesador	Pentium 233MHz	300MHz	867MHz	Sistemas de 32 y 64 bits basados en CPU de AMD, Intel y VIA x86
	RAM	64MB	64MB	512MB	512MB
	Disco Duro	1.5GB	4GB	9GB	2GB

Tabla 2.8 – Requerimientos mínimos de hardware por Sistema Operativo

En la sección 2.6.1 Software de Implementación, se detalla los requerimientos mínimos de hardware para la instalación de la maquina virtual de Java.

### 2.6.3. Equipo informático actual de la EISI

La tabla 2.9 contiene información referente a las descripciones técnicas de las computadoras con las que cuenta actualmente la Escuela de Ingeniería en Sistemas informáticos.

<sup>24</sup> Java. *Maquina Virtual de Java*, Recuperado el 1 de mayo de 2009, de <http://www.java.com/es/download/manual.jsp>

<sup>25</sup> Microsoft, Ayuda y Soporte, Recuperado el 29 de mayo de 2009, de <http://support.microsoft.com/kb/314865>

<sup>26</sup> Ubuntu, Documentación, Recuperado el 29 de mayo de 2009, de <https://help.ubuntu.com/community/Installation/SystemRequirements>

<sup>27</sup> Apple, Mac OS X Leopard, Recuperado el 29 de mayo de 2009, de <http://www.apple.com/macosx/techspecs/>

<sup>28</sup> Sun, System Requirements, Recuperado el 29 de mayo de 2009, de <http://es.sun.com/practice/software/solaris/specs.jsp>

Procesador	Memoria	Disco Duro	Sistema Operativo	Cantidad
AMD Athlon k7	512MB	60GB	Windows XP Pro	10
Pentium 4 de 1.8GHz	512MB	40GB	Windows XP Pro	20
Pentium 4 de 1.8GHz	256MB	40GB	Windows XP Pro	20
IBM Pentium 4 de 1.8GHz	256MB	40GB	Windows XP Pro	5
Pentium I, II y otras de baja frecuencia.	64MB	50MB	MS-DOS	19

*Tabla 2.9 – Descripción técnica de las computadoras con las que cuenta la EISI.*

En base a la información anterior, se concluye que:

Para la utilización del software a construir, la EISI debe actualizar al menos 19 computadoras que actualmente cuentan con poca capacidad de hardware.

Actualmente existe un plan de renovación de los equipos informáticos de los centros de cómputo de la EISI, con lo cual se espera recibir el próximo año 111 equipos con las características que se presentan en la tabla 2.10

Elemento	Descripción
Procesador	Core 2 Duo de 3GHz
RAM	2GB
Disco Duro	200GB

*Tabla 2.10 – Descripción de equipo informático a renovar en la EISI<sup>29</sup>.*

### 3. Diagramas de casos de uso

#### 3.1. Nomenclatura

##### 3.1.1 Descripción del Personal Involucrado

###### Usuarios

###### Administrador del sistema (EISI)

Será el ente encargado de administrar, distribuir las diferentes versiones de la aplicación, así como también de ser el ente encargado de instalar la misma en los centros de cómputo donde será necesario hacer uso de la herramienta para utilizarla en las prácticas de laboratorio.

###### Usuario (docentes y alumnos)

<sup>29</sup> La información anterior fue proporcionada por la Jefatura de la Comunicación y Ciencias de la Computación de la EISI.

Serán las personas que utilizaran la herramienta, los docentes para utilizarla como apoyo para el proceso enseñanza – aprendizaje del proceso solucionador de problemas; Los alumnos utilizaran la herramienta para desarrollar los ejercicios de forma práctica y dinámica.

### 3.1.2 Objetivos de Nivel de Usuario

Los usuarios necesitan un software para satisfacer sus objetivos:

#### Administrador del sistema

Administrar las diferentes versiones de la herramienta, distribuir las diferentes versiones de la herramienta.

#### (Usuarios) Docentes y alumnos

Docentes: Utilizarlo para la enseñanza – aprendizaje del proceso solucionador de problemas.

Alumnos: Utilizarla para el desarrollo de los ejercicios sobre la temática del proceso solucionador de problemas.

### 3.1.3 Perspectiva del sistema

La herramienta para el desarrollo asistido de diagramas de flujo funcionará en condiciones normales en los centros de cómputo de la EISI proporcionando diferentes funcionalidades dependiendo de la materia en la cual se esté utilizando.

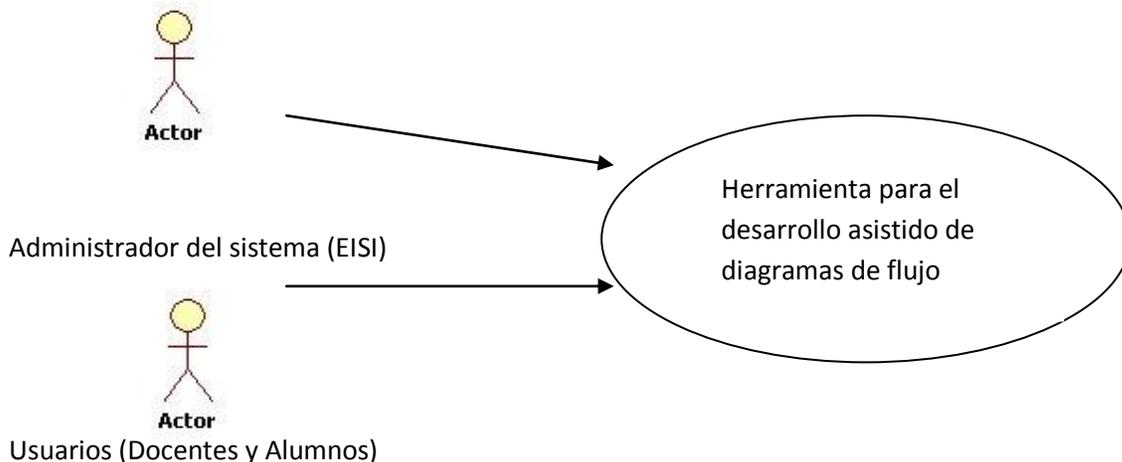


Figura 3.2 Caso de uso principal

### 3.1.4 Resumen de beneficios

Características soportada	Beneficio del personal involucrado
Soporte a todo el proceso solucionador de problemas.	Ser utilizada para el proceso enseñanza – aprendizaje de la metodología para resolver problemas.
Multiplataforma	Poder utilizar la herramienta en otros sistemas operativos diferente a Windows.
Ejecución de los flujogramas.	Posibilidad de verificar los resultados de los flujogramas.
Capacidad de exportación de Flujograma a Pseudocódigo y viceversa; y de flujograma a dos lenguajes de programación.	Poder verificar el flujograma diseñado en lenguaje narrado o pseudocódigo, así como en dos lenguajes de programación.

*Tabla 3.2 Resumen de beneficios*

### 3.1.5 Resumen de las características del sistema

- Capacidad de dar soporte al proceso enseñanza - aprendizaje del proceso solucionador de problemas.
- Capacidad de ejecutar el flujograma para validar los resultados obtenidos.
- Capacidad de generar pseudocódigo a partir del flujograma creado.
- Capacidad de generar código fuente a partir del flujograma creado.

Ver en DVD – Documentación / Análisis y Diseño del Sistema /

### 3.1.6 Modelo de caso de Uso

#### Diagrama de Caso de uso de Interfaz Gráfica

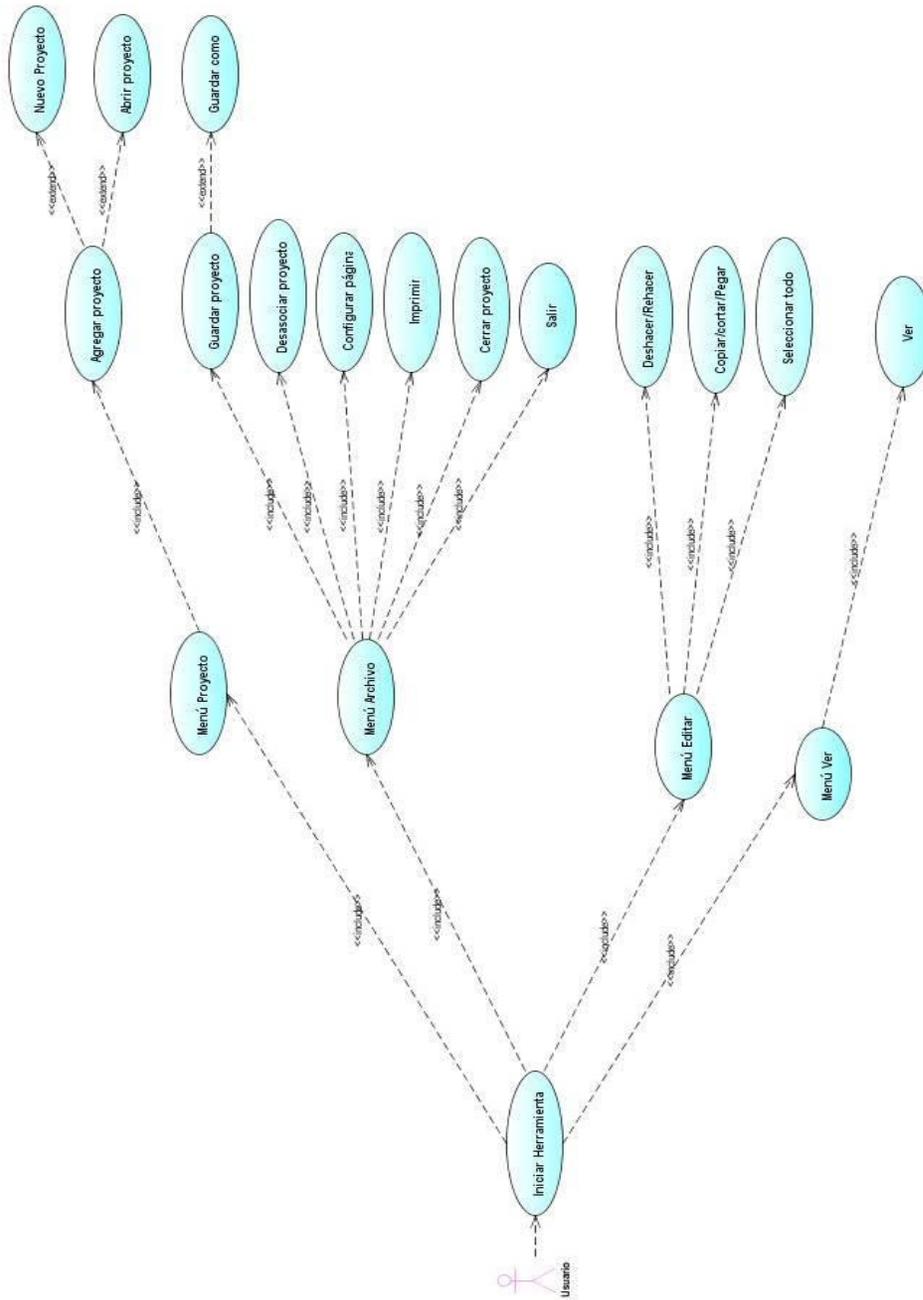


Figura 3.3 Diagrama de Caso de Uso para Interfaz Gráfica

**Diagrama de Caso de Uso de Funcionalidades Principales**

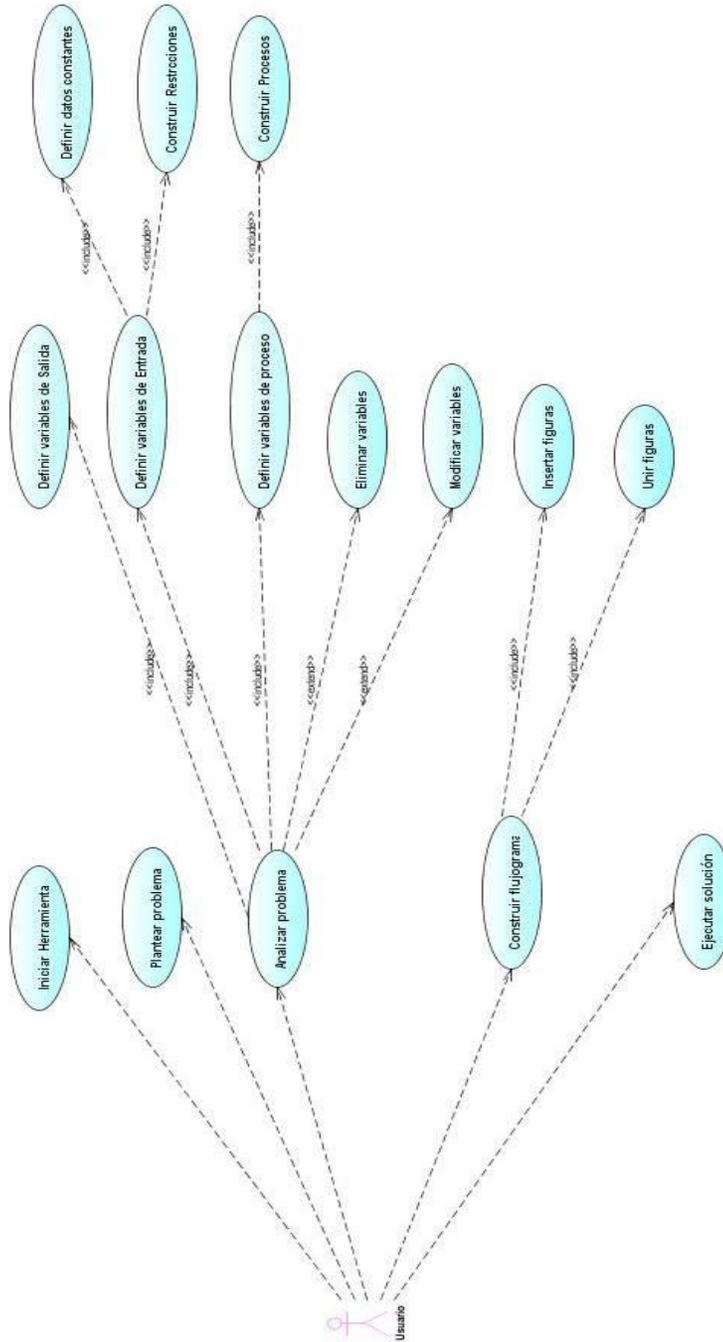


Figura 3.4 Diagrama de Caso de Uso para Funcionalidades Principales

## CAPÍTULO III: DISEÑO DE LA HERRAMIENTA

### 1. Diseño de estándares

En tecnología y otros campos, un **estándar** es una especificación que regula la realización de ciertos procesos o la fabricación de componentes para garantizar la interoperabilidad.

Los objetivos del diseño de sistemas son muy amplios y afectan aspectos tanto de la aplicación como de la organización en la que será utilizado el sistema. Por consecuencia, no debe sorprender el hecho de que los grupos de desarrollo de aplicaciones mejor manejados, también mantengan estándares para el desarrollo de aplicaciones. Las especificaciones de diseño se establecen dentro del marco fijado por los estándares.

#### 1.1 Ventajas de la estandarización

- Ayuda al entrenamiento del nuevo personal dentro y fuera de la organización de sistemas.
- Es útil para cualquiera que tenga la responsabilidad del mantenimiento de los sistemas
- Ayuda a los analistas y diseñadores de sistemas en el trabajo de integración de sistemas.
- Asegura que el sistema opere correctamente.
- Se utilizan eficientemente los recursos que se dispongan.

A través del uso e implementación de estándares, se han definido las normativas a seguir para la creación de nombres de los diferentes elementos que intervienen en el sistema, cuya clasificación se ha establecido de la siguiente manera:

- Estándares de programación
- Estándares de interfaz
- Estándares de documentación

Durante el desarrollo de un sistema informático, es necesario definir y establecer patrones de diseño, programación, documentación, etc., con el objetivo de que el sistema pueda comprender y administrar posteriormente por parte de informáticos ajenos al desarrollo inicial, por lo que a continuación se describen los aspectos que serán estandarizados al momento de diseñar los diferentes elementos del sistema.

## 1.2 Estándares de la aplicación

### 1.2.1 Estándares para Generación de pseudocódigo

#### 1.2.1.1 Palabras reservadas

Palabra	Significado	Nomenclatura
PseudoCodigo	Servirá para comenzar un pseudocódigo	PseudoCodigo NombreSolucion
Inicio	Servirá para delimitar un lote de instrucciones. Se usara para iniciar un pseudocódigo y todas aquellas estructuras que puedan tener dentro de ellas más de una línea. Si No, Selección múltiple Bucle Hacer Desde - Hasta, mientras	Inicio <i>instrucciones</i> fin
Fin	Servirá para finalizar un bloque delimitado. Es obligatorio cuando se ha utilizado Inicio y no se ha cerrado.	Inicio <i>instrucciones</i> Fin
Declarar	Servirá para poder declarar las variables que se utilizaran en el pseudocódigo. Si se declaran más de una variable se pueden separar por una coma y en la misma línea todas serán del mismo tipo.	Declarar NombreVariable como Tipo de dato.
Leer	Servirá para representar la captura de datos desde el teclado. En el caso de que sean muchas variables entonces se separaran por coma.	Leer NombreVariable
Asignar	Servirá para poder colocar valores a las variables.	Asignar NombreVariable = Valor
Escribir	Servirá para poder imprimir. Se pueden imprimir el contenido, variables y/o constantes.	Escribir "Contenido a imprimir"
Llamar	Servirá para hacer invocaciones a los módulos	Llamar NombreModulo

	que se definan. Estos pueden ser funciones o procedimientos.	
--	--	--

Tabla 1.1 Palabras reservadas

**1.2.1.2 Estructuras**

Palabra	Significado	Nomenclatura
Entonces	Palabra que servirá para darle entrada a las instrucciones de la estructura Si – No. Esta palabra reservada no se puede utilizar en otra parte.	
Si – No	Esta estructura servirá para poder evaluar una condición en la cual puede tener una parte verdadera y una parte falsa	Si (Condición Lógica) Entonces Inicio ... Instrucciones Fin [No] Inicio ... Instrucciones Fin
Seleccionar	Esta estructura se usara para evaluar una variable en caso de que esta tenga múltiples valores.	Seleccionar NombreVariable Inicio En Caso que <condicion1>: ...Instrucciones Fin del caso En Caso que <condicion2>: ...Instrucciones Fin del caso Cualquier Caso: ...Instrucciones Fin del caso Fin
Repetir	Servirá para ejecutar un número finito de veces un conjunto de instrucciones.	Repetir (desde VariableContador= Valor inicial, hasta VariableContador ,incrementar/decrementar en #)
Mientras	Servirá para poder repetir instrucciones mediante una	Mientras (CondicionLogica) Hacer

	condición lógica.	Inicio ...Instrucciones Fin
--	-------------------	-----------------------------------

*Tabla 1.2 Estándar de estructuras para el pseudocódigo*

### 1.2.1.3 Operadores matemáticos

Símbolo	Significado	Nomenclatura
+	Servirá para indicar que se realizará una operación de adición entre dos valores o expresiones.	Operando1 + Operando2
-	Servirá para indicar que se realizará una sustracción entre dos valores o expresiones.	Operando1 – Operando2
*	Servirá para indicar un producto o multiplicación entre dos valores o expresiones.	Operando1 * Operando2
/	Servirá para realizar una multiplicación entre dos valores o expresiones incluyendo valores decimales.	Operando1 / Operando2
=	Servirá para realizar una asignación de una expresión o valor.	NombreVariable = Valor1 NombreVariable = Expresion1

*Tabla 1.3 Operadores aritméticos*

### 1.2.1.4 Operadores relacionales o comparativos

Símbolo	Significado	Nomenclatura
>	Mayor que	Expresion1 > Expresion2
<	Menor que	Operando1 – Operando2
>=	Mayor o igual	Operando1 * Operando2
<=	Menor o igual	Operando1 / Operando2
==	Igual (comparación)	Expresion1 == Expresion2
!=	Distinto a	Expresion1 != Expresion2

*Tabla 1.4 Operadores relacionales*

### 1.2.1.5 Operadores lógicos

Símbolo	Significado	Nomenclatura
Y	Indica que se realizará la comparación entre dos expresiones lógicas y cuyo resultado devolverá un resultado verdadero o falso	Expresion1 Y Expresion2
O	Indica que se realizará una comparación entre dos expresiones	Expresion1 O Expresion2
NO	NO	!Expresion1

Tabla 1.5 Operadores lógicos

### 1.2.1.6 Funciones

Símbolo	Significado	Nomenclatura
Ceil(par)	Devuelve el valor entero más alto con respecto al número que se ha enviado como argumento o parámetro a la función	Ceil(x)
Cos(par)	Devuelve el valor del coseno del argumento	Cos(x)
Exp(par)	Eleva "e" (2.71828.....) a la potencia que es el argumento de la función.	Exp(x)
Abs(par)	Devuelve el valor absoluto de un número	Abs(x)
Floor(par)	Devuelve el mayor número entero menor o igual que el argumento.	Floor(x)
Log(par)	Función logarítmica de base a, es la función inversa de la exponencial de base a de un parámetro o argumento enviado.	Log(x)
Pow(par1,par2)	Calcula x elevado a la potencia de y.	Pow(x,y)
Sin(par)	Devuelve el seno de la expresión	Sin(x)
Sqrt(par)	Restituye la raíz cuadrada del argumento	Sqrt(x)

Tabla 1.6 Operadores lógicos

### 1.2.1.7 Consideraciones del ambiente

- ✓ En el editor de pseudocódigo se pondrá un número que identifique la línea donde se está trabajando.
- ✓ Se resaltará en color azul aquellas palabras que sean reservadas.
- ✓ Todo texto que no sea una palabra reservada deberá ir entre comillas dobles.
- ✓ Todo texto que esté entre comillas se distinguirán con color rojo.

Ver el estándar de las figuras para la construcción de diagramas de flujo en el Anexo 3.

## 1.3 Estándares de programación

### Métodos

Los nombres de los métodos que se utilizarán en el código fuente de la aplicación son:

XXX \_ XXXXX  
 { } { }

Nombre del Método → **Prefijo Identificador**

Donde **Prefijo** será “sub” e **Identificador** una palabra cualquiera que preferiblemente hará referencia al propósito del método que representa. Sin embargo, la firma completa para la definición de los métodos está formado por el alcance (cuyos valores pueden ser Public, Private, Protected), el tipo de dato que retornará (cuyos valores pueden ser Byte, Short, Int, Long, Char, Float, Double o Boolean), Nombre del método y los parámetros, cada uno con sus respectivos tipos de datos.

Por ejemplo: Método para calcular un total de variables X1 y X2 sería:

Alcance
Tipo de Dato de retorno
Nombre del método
Parámetros

### Clases

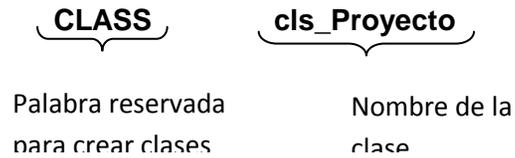
Las diferentes clases que se implementarán en el código fuente utilizarán el siguiente formato:

XXX \_ XXXXX  
 { } { }

Nombre de la clase → **Prefijo Identificador**

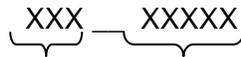
Donde el Prefijo será “cls” y el **Identificador** una palabra que hace referencia al propósito de la clase.

Por ejemplo: Clase Proyecto:



### Interfaces

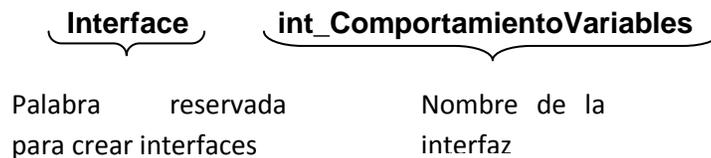
Los nombres de los métodos que se utilizarán en el código fuente de la aplicación son:



Nombre de la Interfaz → **Prefijo Identificador**

Donde **Prefijo** será “int” e **Identificador** una palabra cualquiera que preferiblemente hará referencia al propósito de la interfaz que representa, utilizando el estándar Camel Case para colocar el nombre, el cual consiste en iniciar cada palabra con letra mayúscula.

Por ejemplo: Interfaz para comportamiento de variables sería:



## 1.4 Estándares de documentación

### 1.4.1 Documentación Interna

En cada uno de los archivos que formaran parte de la aplicación, la documentación interna estará compuesta de la siguiente manera:

Inicio de cada archivo

/\*\*\*\*\*

Nombre del archivo:

Descripción:

Fecha de creación: 99/99/9999

Creado por: MACMERMS

Fecha de última modificación: 99/99/9999

Modificado por: MACMERMS

\*\*\*\*\*/

{Código fuente contenido en el archivo}

Antes de cada clase

/\*\*\*\*\*

Nombre de la clase:

Descripción:

Fecha de creación: 99/99/9999

Creado por: MACMERMS

Fecha de última modificación: 99/99/9999

Modificado por: MACMERMS

\*\*\*\*\*/

{Código fuente de la clase}

Comentarios internos

De igual forma, para realizar comentarios específicos dentro de la codificación de cada archivo se hará uso de la doble pleca para aquellos comentarios de una sola línea y la convención pleca asterisco al inicio y asterisco pleca al final de los que son de más de una línea. Por ejemplo: (en código Java)

```
var1 = X + Y; // Se asigna el resultado de la suma de las variables X y Y a var1.
```

var1=(X + Y) \* W; /\* Se suman las variables X y Y, luego se multiplica ese resultado por el valor de la variable W, esto se asigna a var1 \*/

### 1.4.1.1 Convención de Nombres de Elementos de Formularios

Todos los formularios cuentan con elementos que se utilizan para el envío de información, entre estos podemos mencionar: cajas de texto (jTextBox), botones de envío (jButton), panel (JPanel) entre otros. El nombramiento de todos estos elementos se hará en base a la utilización de prefijos de 3 letras seguidos de palabras que intenten identificar el contenido del elemento tal y como se muestra a continuación:

Elemento	Formato del nombre	Ejemplo
Formulario (JPANEL)	jPnlNombre	jPnlAnálisis
Botón de envío (JBUTTON)	jBtnNombre	jBtnSiguiente
Área de texto (JTEXTAREA)	jTxaNombre	jTxaEntrada
Etiqueta (JLABEL)	jLblNombre	jLblTipoProyecto
Tabla (JTABLE)	jTblNombre	jTblModulos
Vista de árbol(JTREE)	jTreNombre	jTreVariable

Tabla 1. 7 Convención de nombres elementos de formulario

### 1.4.1.2 Convención para Nombres de Archivos de la Aplicación

La asignación de nombres de los archivos que conformen la aplicación que se desarrollará se hará en base a su contenido y al tipo del archivo específicamente, tal y como se muestra a continuación:

Tipo de archivo	Contenido	Descripción
.PrSP	Código fuente Java	Es la extensión del archivo que la aplicación creará para los proyectos.
.xml	Código fuente XML	Incluye código fuente que puede ser interpretado por muchos lenguajes de programación.
.jar	Librerías de Java	Librería creada en lenguaje de programación Java.

Tabla 1.8 Convención de nombres de archivos de aplicación.

La convención para representar los datos en los formularios es la siguiente:

- **XX--99--XX:** Se utiliza para representar caracteres alfanuméricos.  
Ejemplo: **XX--40--XX** Indica que el campo del panel estaría formada por un máximo de 40 caracteres alfanuméricos.
- **9:** Se utiliza para representar datos numéricos.  
Ejemplo: **99.9** Indica que el campo estaría formado por 2 enteros y un decimal.
- **99/99/9999:** Se utiliza para representar fechas en el siguiente formato: día/mes/año.
- **0 y 1:** Se utiliza para representar datos booleanos, 0: cierto y 1: falso
- Está diseñado para ser leído de izquierda a derecha y de arriba hacia abajo.

### 1.4.1.3 Estándares para Elaboración de Manuales

Formato de documentos:

Atributo	Valor
Tamaño de papel	Carta (216 x 279 mm.)
Márgenes Superior - Izquierdo	2cm-2.5cm

<b>Inferior - Derecho</b>	2cm-2cm
<b>Tipo de fuente</b>	Arial
<b>Tamaño de fuente tipo título</b>	16 Pto.
<b>Estilo de fuente tipo título</b>	Negrita y con borde color gris al 15%
<b>Color de fuente tipo título</b>	Negro
<b>Tamaño de fuente tipo párrafo</b>	11 Pto.
<b>Estilo de fuente tipo párrafo</b>	Normal
<b>Color de fuente tipo párrafo</b>	Negro
<b>Espaciado</b>	Posterior a 6 puntos
<b>Interlineado</b>	Sencillo
<b>Alineación</b>	Justificado

*Tabla 1.9 Formato de documentos para manuales*

El contenido de cada uno de los manuales es el siguiente:

Manual de Instalación/desinstalación.

- Portada
- Índice
- Introducción
- Requerimientos de instalación
- Instrucciones de instalación
- Instrucciones de desinstalación
- Preguntas y respuestas.

Manual de Usuario.

- Portada
- Índice
- Introducción
- Funcionamiento de la aplicación
- Preguntas y respuestas

Manual Técnico.

- Portada
- Índice
- Introducción a la aplicación
- Descripción de estándares de programación
- Descripción de elementos UML
  - Diagramas de caso de uso
  - Diagramas de secuencia
- Diagrama de clases

## 2. Arquitectura de la herramienta

### 2.1. Diagrama Jerárquico de Procesos

Una de las principales razones de la utilización de este diagrama es la facilidad de dividir un problema complejo en sub-problemas más sencillos y de ser necesario, dividir estos en otros sub-módulos con el fin de hacer más sencilla la tarea de desarrollar o construir cada uno de esos sub-módulos.

A continuación se muestra el esquema del Diagrama de Descomposición Funcional para la “Herramienta para el desarrollo asistido de diagramas de flujo”

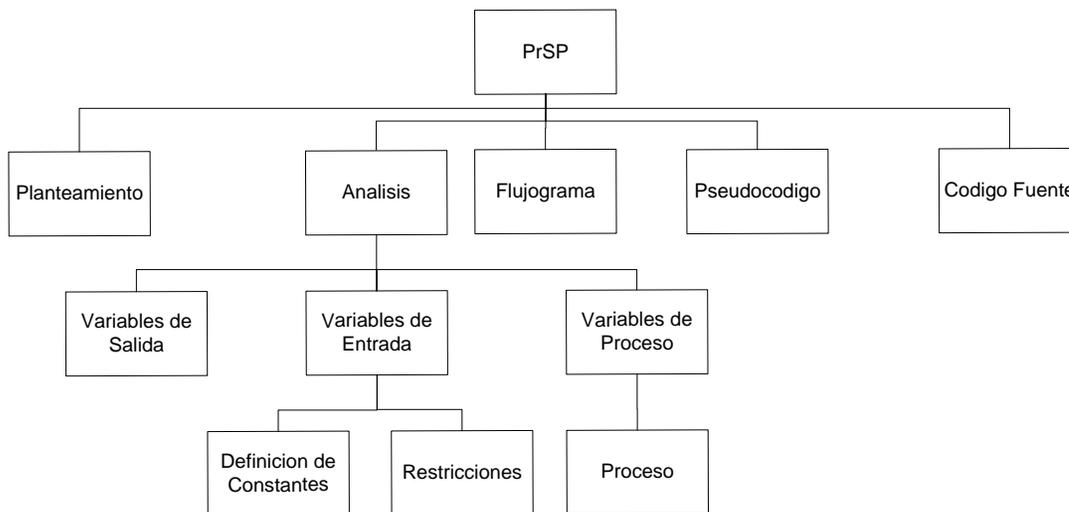


Figura 2.1 - Diagrama Jerárquico de Procesos

#### Descripción de módulos

A continuación se describen de forma breve el propósito de los módulos y sub-módulos mostrados en el esquema anterior:

<b>Módulo</b>	0. EISI-FlowChart
<b>Descripción</b>	Es el módulo principal que brindará soporte en la enseñanza – aprendizaje del proceso solucionador de problemas.

<b>Módulo</b>	1. Planteamiento
<b>Descripción</b>	Módulo en el cual se realiza la captura de los elementos del estado A y el estado B del problema, es decir, los elementos de entrada y el/los elementos de salida del problema.

<b>Módulo</b>	2. Análisis
<b>Descripción</b>	Módulo que permitirá realizar a detalle cada una de los elementos que componen el análisis del problema.

<b>Módulo</b>	2.1 Variables de salida
<b>Descripción</b>	Permitirá la definición de todas las variables de salida identificadas para el problema en estudio.

<b>Módulo</b>	2.2 Variables de Entrada
<b>Descripción</b>	Permitirá la definición de todas las variables de entrada identificadas para el problema en estudio.

<b>Módulo</b>	2.2.1 Definición de Constantes
<b>Descripción</b>	Permitirá definir los elementos los cuales se considera que su valor no cambiará a los largo de todo el problema en estudio.

<b>Módulo</b>	2.2.2 Restricciones
<b>Descripción</b>	Permitirá definir cada una de las condiciones que acompañan a las variables de entrada definidas.

<b>Módulo</b>	2.3 Variables de Proceso
<b>Descripción</b>	Permitirá definir cada una de las variables identificadas como elementos de proceso los cuales son utilizados a lo largo del problema.

<b>Módulo</b>	2.3.1 Procesos
<b>Descripción</b>	Permitirá definir cada una de las operaciones o cálculos a realizar para poder proporcionar los resultados esperados.

<b>Módulo</b>	3. Flujograma
<b>Descripción</b>	Módulo a través del cual se podrá crear de forma gráfica la propuesta de solución al problema en estudio.

<b>Módulo</b>	4. Pseudocódigo
<b>Descripción</b>	Módulo que permitirá obtener el algoritmo narrado de la solución propuesta y así poder comprenderlo de forma más fácil y sencilla.

<b>Módulo</b>	5 Código Fuente
<b>Descripción</b>	Módulo que permitirá obtener el código fuente en Turbo C o Java a partir del flujograma creado

### 3. Diagrama de Clases

#### 3.1. Diagrama de Clases

El propósito de este diagrama es el de representar los objetos fundamentales de la aplicación, es decir, los que percibe el usuario y con los que espera tratar para completar su tarea.

La notación utilizada para la construcción del diagrama de clases es la siguiente:

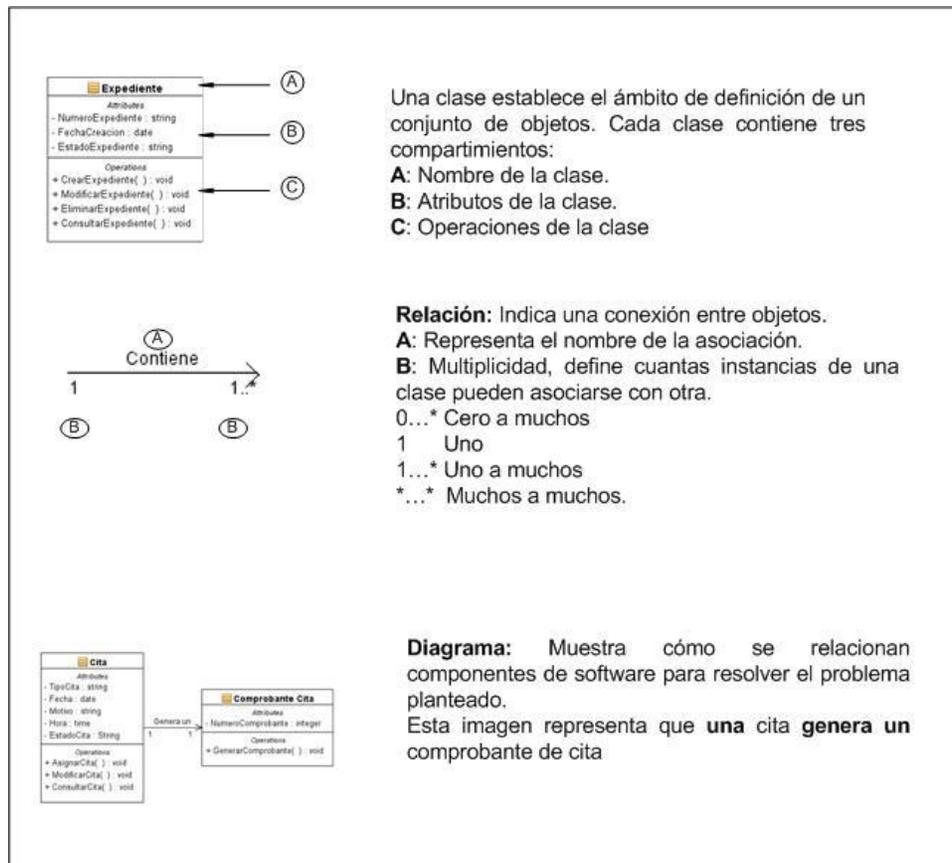


Figura 3.1 Descripción de los elementos del Diagrama de Clases

El siguiente diagrama de clases es un extracto de las clases y relaciones entre ellas diseñadas para el desarrollo de la aplicación, para tener mayor detalle del diagrama de

clases con sus atributos y métodos, referirse al DVD – Documentación / Análisis y Diseño/.

**DIAGRAMA DE CLASES Y SUS RELACIONES**

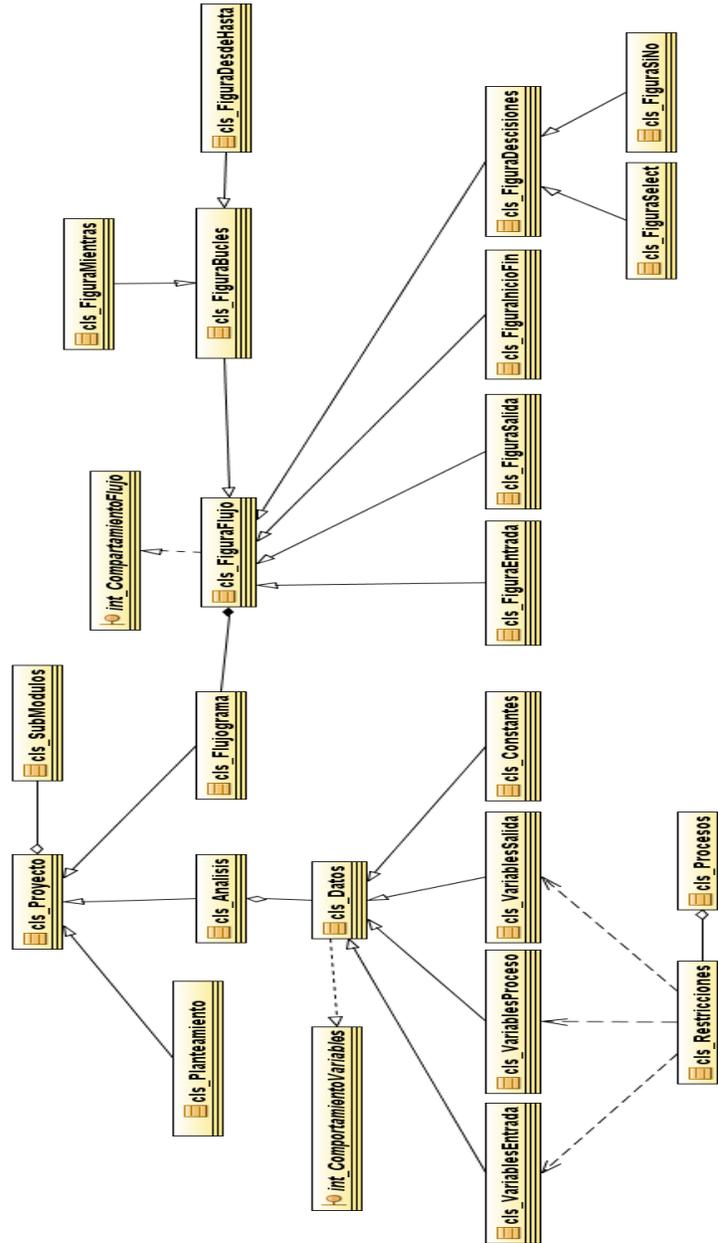


Figura 3.2 Consolidación del Diagrama de Clases y sus relaciones

## **4. Diseño de interfaces**

El diseño de interfaces describe cómo se comunica el software consigo mismo, con los sistemas que operan con él y con los usuarios que lo emplean. Para detallar estas interacciones a continuación se define la interfaz externa de la herramienta EISI – FlowChart.

### **4.1. Interfaz externa**

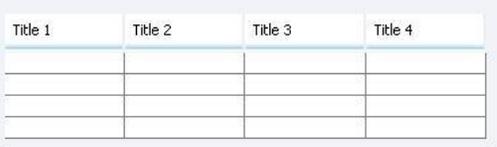
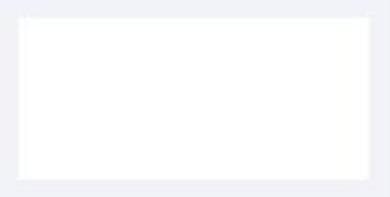
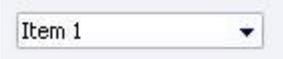
El presente apartado comprende todos los aspectos relacionados al diseño de la interfaz gráfica de usuario de la herramienta, en el cual se plasma el patrón a seguir para representar los elementos utilizados para establecer la comunicación entre el usuario y la aplicación.

A continuación se describen los elementos que se deberán seguir para el respectivo diseño de interfaz:

#### **4.1.1. Aspectos a considerar en cuanto a la interfaz de usuario**

- Las interfaces deberán contener un orden lógico en cuanto a la manera de realizar y representar los elementos en ellas contenidos. Es decir, para las interfaces de entrada, los elementos de datos u otros objetos deberán guiar al usuario en el curso de las acciones a realizar en una forma ordenada, de arriba hacia abajo y de izquierda a derecha.
- Los títulos de la interfaz deben ser significativos, de manera que permitan al usuario distinguir inequívocamente el uso que representan.
- Las interfaces deberán contener los mecanismos necesarios para notificar y guiar a los usuarios de una manera sencilla en el caso de generarse algún error, ya sea de usuario o de la aplicación.

#### 4.1.2. Elementos gráficos a utilizar para el diseño de interfaces

Elemento gráfico	Nombre Objeto	Descripción
	Botones de opciones	Muestra múltiples opciones de las cuales el usuario solo puede escoger una.
	Botón de comando	Es utilizado para ejecutar una operación.
	Cuadro de verificación	Permite indicar si un criterio es falso o verdadero o puede utilizarse para seleccionar una o más opciones de una serie de alternativas.
	Tabla	Muestra un conjunto de datos organizados en filas y columnas.
	Cuadro de edición	Permite la captura amplia de datos y una mejor visualización de la misma.
	Cuadro de lista	Muestra una lista de datos de forma secuencial.
	Lista desplegable	Es una lista desplegable de opciones en la que el usuario puede seleccionar un elemento de la lista de alternativas que se muestra.

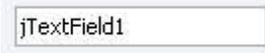
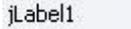
Elemento gráfico	Nombre Objeto	Descripción
	Cuadro de texto	Permite la captura de datos desde teclado.
	Etiqueta	Es utilizado para describir o nombrar un objeto.
	Vista de árbol	Permite mostrar elementos en una cadena de directorios, haciendo a cada carpeta un contenedor de otros elementos.
	Contenedor de pestañas	Permite mostrar mediante pestañas diferentes elementos similar a los componentes de un menú.

Tabla 4.1 Elementos gráficos a utilizar para el Diseño de Interfaces

#### 4.1.3. Diseño de Menú

En la aplicación se utilizarán dos diseños para presentar los menús, el primero que se utilizará como lista desplegable, el cual es comúnmente utilizado por diversas herramientas y el otro utilizando figuras como acceso directo a las opciones de la aplicación. Además se utilizarán menús contextuales utilizando los siguientes diseños:

Elemento gráfico	Nombre Objeto	Descripción
	Barra de menú	Presenta las múltiples opciones que tiene el usuario para la aplicación, el usuario puede navegar a través de ellas para seleccionar la opción que desee.

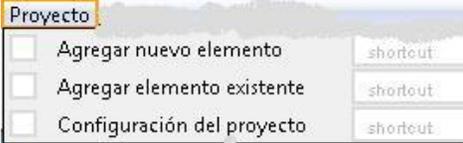
Elemento gráfico	Nombre Objeto	Descripción
	Menú desplegable	Menú utilizado para mostrar las opciones hijas disponibles para construir la solución.
	Menú contextual	Activado mediante el evento de clic derecho sobre un elemento del Explorador de módulos, el cual facilita la realización de actividades comunes.
	Menú utilizando iconos de fácil acceso	Utilización de iconos para facilitar el acceso a las opciones de uso común de los usuarios.

Tabla 4.2 Elementos a utilizar para el diseño de Menús

#### 4.1.4. Diseño de Mensajes

Cada vez que la herramienta deba informar o alertar al usuario, se hará uso de mensajes emergentes. A continuación se muestran los mensajes generales, las opciones de respuesta y su descripción.

##### *Mensajes de operación*

Mensaje	Respuesta	Descripción
Proceso completado exitosamente	Aceptar	Informa al usuario que ha finalizado correctamente el proceso solucionador.
Cerrar proyecto	Aceptar / Cancelar	Consulta al usuario si desea cerrar el proyecto.
Guardar proyecto	Aceptar / Cancelar	Consulta al usuario si desea guardar

		el proyecto.
Datos guardados exitosamente	Aceptar	Informa al usuario que los datos se han guardado exitosamente.
Eliminar datos	Aceptar / Cancelar	Consulta al usuario si desea eliminar el o los datos seleccionados para borrarse.
Datos eliminados exitosamente	Aceptar	Informa al usuario que los datos se han eliminado exitosamente.

*Tabla 4.3 –Diseño de mensajes de operación*

***Mensajes de advertencia***

<b>Mensaje</b>	<b>Respuesta</b>	<b>Descripción</b>
Definición de variables incompletas	Aceptar	Informa al usuario que no ha definido las variables de manera correcta.
Flujograma incompleto	Aceptar	Informa al usuario que el flujograma está incompleto para poder ejecutarlo.
Variable no definida	Aceptar	Informa al usuario que la variable que desea utilizar no está definida en el Análisis de Variables.
Modificación de variables	Aceptar	Informa al usuario que la variable que se está modificando está siendo utilizada en otro paso y que se verá afectada.
Eliminación de variables	Aceptar	Informa al usuario que no puede ser eliminada la variable por que está siendo utilizada en otro proceso.
Espacio insuficiente para almacenar el proyecto	Aceptar	Informa al usuario que en la ubicación en que se desea guardar no hay suficiente espacio disponible.

*Tabla 4.4 –Diseño de mensajes de advertencia*

## **CAPITULO IV: PLAN DE IMPLEMENTACIÓN**

### **1. Planeación**

Considerando que la etapa de planeación es una de las más importantes para el proyecto de implementación de un software, se ha desarrollado la presente guía que contiene una propuesta de elementos a considerar, la cual consiste en identificar las actividades y tareas que se deben llevar a cabo y los recursos requeridos (financieros, humanos, tecnológicos y tiempo) para realizar una implementación adecuada. Además del plan detallado se producirán documentos tales como cronograma general, estructura organizativa del proyecto, roles y responsabilidades, entre otros.

La guía se presenta como una propuesta metodológica que les permitirá a las autoridades correspondientes de la Escuela de Ingeniería de Sistemas Informáticos, desarrollar la etapa de planeación, para implementar la aplicación EISI - FlowChart a través de un esquema lógico y secuencial.

#### **1.1 Consideraciones Previas a la Etapa de Planeación**

Antes de realizar la etapa de Planeación es conveniente comunicar a los docentes involucrados de la Escuela de Ingeniería de Sistemas Informáticos (EISI), el programa que se va a desarrollar para implementar la aplicación EISI - FlowChart, explicar las fases, plazos y responsables. Así mismo se debe designar a las personas responsables que se encargaran de difundir dicha información, quienes deberán tener conocimiento acerca de la EISI y de la aplicación.

##### **1.1.1 ¿Qué es la Etapa de Planeación?**

Es la etapa en la que se define la estructura organizativa del proyecto, el cronograma general y el plan de comunicaciones, así mismo se determinan las actividades a efectuar para llevar a cabo la implementación de la aplicación EISI - FlowChart. El producto resultante de la misma es un plan detallado de actividades que describe responsables, tiempos de ejecución y plazos.

##### **1.1.2 ¿Por qué es Conveniente Realizar la Etapa de Planeación?**

Es conveniente realizar esta fase, porque permite determinar las actividades a desarrollar, los recursos, los plazos de ejecución y los responsables. La planeación establece el

horizonte para dar cumplimiento a los objetivos de un proyecto, evitando pérdida de tiempo y de recursos.

### 1.1.3 ¿Cómo se Realiza la Etapa de Planeación?

La Figura 5.1 presenta la etapa de planeación con un enfoque de procesos (entradas, actividades y salidas) de tal manera que facilite su aplicación.

FASE DE PLANEACIÓN		
ENTRADAS	ACTIVIDADES	SALIDAS
<ul style="list-style-type: none"> <li>Compromiso formal de la Alta Dirección</li> </ul>	<pre> graph TD     A[Planificar la realización de la Fase de Planeación] --&gt; B[Ejecutar la Fase de Planeación]     B --&gt; C[Aprobar los resultados de la Fase de Planeación]             </pre>	<ul style="list-style-type: none"> <li>Plan detallado de actividades necesarias para llevar a cabo el proyecto</li> </ul>
<ul style="list-style-type: none"> <li>Personal involucrado de la EISI con conocimientos del proyecto</li> </ul>		<ul style="list-style-type: none"> <li>Cronograma general y Plan de comunicaciones para el proyecto</li> </ul>
<ul style="list-style-type: none"> <li>Estructura organizativa del proyecto</li> </ul>		<ul style="list-style-type: none"> <li>Estructura organizativa del proyecto: Listado de personas que conformarán los grupos de trabajo (con % de disponibilidad de tiempo)</li> </ul>
<ul style="list-style-type: none"> <li>Otros recursos necesarios para la fase de planeación tales como: Infraestructura física, equipos y software, entre otros</li> </ul>		<ul style="list-style-type: none"> <li>Recursos necesarios para el proyecto</li> </ul>

Figura 1.1 - Esquema de procesos aplicado a la etapa de planeación

## 1.2 Método de Implementación de la Aplicación EISI - FlowChart

En esta sección se presenta el método seleccionado para realizar la implementación de la aplicación EISI - FlowChart, la selección se hizo en base al estudio y evaluación de las ventajas y desventajas que poseen los métodos de conversión más comunes, lo cual puede consultarse en el Anexo 4.

Los resultados de la evaluación realizada se observan en la Tabla 1.1, los cuales indican que el método más conveniente que debe ser utilizado es el **Paralelo**.

Tipo de Método	Mejor adaptación de los usuarios a la herramienta	Permite la validación completa de la herramienta	Menor Costo	Se dispone del método manual, en caso de fallas en la herramienta	La implementación se realiza de manera rápida	Método soportado por la EISI	Puntaje Total
Paralelo	✓	✓	✗	✓	✗	✓	✓✓✓
Directo	✗	✗	✓	✗	✓	✗	✓✓
Piloto	✓	✓	✗	✓	✗	✗	✓✓✓
Por Etapas	✗	✓	✗	✓	✗	✗	✓✓

Tabla 1.1 – Comparación de Métodos de Conversión de Sistemas.

### 1.3 Listado de Actividades

La estrategia a seguir para el correcto cumplimiento de la propuesta de implementación de la aplicación, está compuesta por las siguientes actividades principales:

- Preparación e Instalación
- Capacitación de personal
- Verificación y pruebas
- Puesta en marcha

A continuación se detallan los objetivos y las tareas de cada una de las actividades principales definidas por el equipo de trabajo.

#### 1.3.1 Preparación

##### Objetivo:

- Dar a conocer el plan de implementación para la herramienta de software EISI – FlowChart y definir plataforma de trabajo.

##### Lista de tareas:

1. Presentar el plan de implementación a las autoridades de la EISI.

Esta actividad tiene como finalidad presentar el proyecto de plan de implementación a las autoridades correspondientes, con la finalidad de verificar el contenido, validarlo y actualizarlo si fuese necesario.

2. Seleccionar la plataforma en que se habilitará la aplicación.

### ***1.3.1.1 Capacitación de Personal***

#### **Objetivo:**

- Proporcionar las capacitaciones requeridas al personal involucrado con el manejo de la aplicación EISI - FlowChart, para prepararlos en el uso y manipulación del mismo.

#### **Lista de tareas:**

1. Preparar el contenido de la capacitación.
2. Definir los recursos que se requieren para realizar la capacitación.
  - Identificar y definir al personal que será capacitado.
  - Conformar grupos.
  - Definir horarios.
  - Definir el local en donde se impartirá la capacitación.
  - Especificar el equipo tecnológico y el material didáctico a utilizar.
3. Informar formalmente al personal que recibirá la capacitación, sobre el plan a seguir (grupos, lugar, fechas, horarios, equipo y material a utilizar).
4. Realizar la capacitación.
5. Evaluar los conocimientos adquiridos del personal capacitado.

### ***1.3.1.2 Instalación***

#### **Objetivo:**

- Realizar los pasos necesarios para que la instalación de la aplicación EISI – FlowChart se lleve a cabo de manera satisfactoria.

#### **Lista de tareas:**

1. Instalar ó habilitar el sistema operativo requerido en cada uno de los equipos del laboratorio de cómputo.

Si la plataforma elegida es Windows, se puede omitir este paso debido a que los equipos informáticos ya se encuentran habilitados con esta plataforma. Sin embargo, en caso de cambiar de sistema operativo, no hay restricciones para instalar la aplicación en ninguna plataforma específica.

2. Instalar la Máquina Virtual de Java (JVM versión 6 Update 18, última actualización)
3. Instalar y configurar la aplicación EISI – FlowChart.

### ***1.3.1.3 Verificación y Pruebas***

#### **Objetivo:**

- Llevar a cabo las tareas de verificación de resultados y pruebas de los módulos que conforman a la aplicación EISI - FlowChart, que garanticen el correcto funcionamiento de ésta.

#### **Lista de tareas:**

1. Preparar casos de prueba para verificar el funcionamiento de la aplicación.
2. Aplicar pruebas.
3. Revisar los resultados de las pruebas.
4. Realizar correcciones y ajustes a la herramienta de ser necesario.
5. Documentar las modificaciones ejecutadas en la aplicación en caso de ser necesario.
6. Repetir los pasos.

### ***1.3.1.4 Puesta en Marcha***

#### **Objetivo:**

- Llevar a cabo los pasos necesarios para poner en funcionamiento la aplicación EISI - FlowChart.

#### **Lista de tareas:**

1. Utilizar la aplicación EISI – FlowChart para el desarrollo de la asignatura Introducción a la Informática.
2. Utilizar la aplicación EISI – FlowChart para evaluar las actividades desarrolladas por los estudiantes.
3. Sustituir totalmente el método de desarrollo manual por el uso la aplicación EISI - FlowChart.

### **1.3.2 Recursos a Utilizar**

Los recursos que se requieren para llevar a cabo el plan de implementación se han dividido de la siguiente forma:

- Recurso Humano.
- Tiempo - Cronograma de Actividades.
- Recursos Financieros.
  - Costo de Recurso Humano en base al tiempo disponible.
  - Costo de Materiales a utilizar.
  - Costo de Recursos tecnológicos a utilizar.

A continuación se describe cada uno de estos elementos.

### **1.3.2.1 Recurso Humano**

Se considera que el personal requerido para llevar a cabo el plan de implementación debe estar conformado por un equipo de 14 personas, incluyendo a los participantes en la capacitación, tal como se muestra en la Tabla 1.2.

Cargo	Cantidad
Director de Proyecto	1
Encargado de Ejecución de Actividades	1
Docentes Universitarios	8
Encargados de Mantenimiento y Soporte de Computadoras	3
Instructor de Capacitaciones	1
<b>Total de Responsables</b>	<b>14</b>

*Tabla 1.2 - Recurso humano requerido para realizar el plan de implementación*

### 1.3.2.2 Tiempo - Cronograma de Actividades

La realización de todas las actividades y tareas identificadas es, a la vez, requisito necesario y suficiente para lograr el resultado final que el proyecto persigue.

El Cronograma corresponde a un documento que presenta de manera resumida las fases del proyecto y los tiempos de ejecución. Se basa en el plan detallado del proyecto y sirve como instrumento de socialización y a su vez como mecanismo de control.

Para elaborar el cronograma requerido, en primera instancia se tomaron en cuenta las actividades y sub-actividades previamente establecidas para poder llevar el plan de implementación, luego se hizo uso del diagrama de Gantt, éste diagrama es la forma habitual de presentar el plan de ejecución de un proyecto, recogiendo en las filas la relación de actividades a realizar y en las columnas la escala de tiempos que estamos manejando, mientras la duración y situación en el tiempo de cada actividad se representa mediante una línea dibujada en el lugar correspondiente.

La ventaja de un gráfico de este tipo es la facilidad de construcción y comprensión, y el mantenimiento de la información global del proyecto.

A través de la Tabla 1.3 puede observarse que se ha asignado una abreviatura al cargo de cada responsable de la implementación de la herramienta de software, esto se ha realizado con el objetivo de identificarlos de manera más fácil en el cronograma de actividades.

Cargos	Cantidad	Abreviatura
<b>DIR</b> ector de Proyecto	1	DIR
<b>ENC</b> argado de Ejecución de Actividades	1	ENC
<b>DOC</b> entes de la Cátedra	8	DOC
Encargado de Soporte y <b>MAN</b> tenimiento de Computadoras	3	MAN
<b>INS</b> tructor de Capacitaciones	1	INS
<b>Total de Responsables</b>	<b>14</b>	

Tabla 1.3 - Definición de abreviatura para el cargo de cada responsable.

La Tabla 1.4 muestra el listado de actividades, tareas y sub-tareas del proyecto de implementación, asimismo se presentan las precedencias, estimación de tiempo y responsables. Luego la Figura 1.2 muestra el diagrama de Gantt correspondiente. Los

tiempos estimados se han especificado en base al conocimiento y experiencia laboral del equipo de trabajo.

	Nombre de tarea	Duración	Comienzo	Fin	Pred	Nombres de los recursos
1	<b>CRONOGRAMA DE ACTIVIDADES PARA PROPUESTA DE PLAN DE IMPLEMENTACION</b>	<b>7.38 días</b>	<b>vie 15-01-10</b>	<b>mar 26-01-10</b>		
2	<b>PREPARACIÓN</b>	<b>0.5 días</b>	<b>vie 15-01-10</b>	<b>vie 15-01-10</b>		
3	Presentar el plan de implementación a las autoridades de la EISI.	3 horas	vie 15-01-10	vie 15-01-10		DIR,ENC
4	Seleccionar la plataforma en que se habilitará el sistema.	1 hora	vie 15-01-10	vie 15-01-10	3	DIR,ENC
5	<b>CAPACITACION DE PERSONAL</b>	<b>4.06 días</b>	<b>vie 15-01-10</b>	<b>jue 21-01-10</b>		
6	Preparar el contenido de la capacitación.	8 horas	vie 15-01-10	vie 15-01-10		INS
7	Definir los recursos requeridos para realizar la capacitación.	1 hora	lun 18-01-10	lun 18-01-10	6	INS
8	Identificar y definir al personal que será capacitado.	1 hora	lun 18-01-10	lun 18-01-10	7	DIR,ENC
9	Conformar grupos.	0.5 horas	lun 18-01-10	lun 18-01-10	8	DIR,ENC
10	Definir fechas y horarios.	0.5 horas	lun 18-01-10	lun 18-01-10	9	DIR,ENC
11	Definir el local en donde se impartirá la capacitación.	0.5 horas	lun 18-01-10	lun 18-01-10	10	DIR,ENC
12	Especificar el equipo tecnológico y el material didáctico a utilizar.	0.5 horas	lun 18-01-10	lun 18-01-10	11	DIR,ENC
13	Informar al personal involucrado sobre el plan a seguir.	0.5 horas	lun 18-01-10	lun 18-01-10	12	DIR,ENC,INS
14	Realizar la capacitación.	19 horas	lun 18-01-10	mié 20-01-10	13	INS,DIR,DOC,ENC,MAN
15	Evaluar los conocimientos adquiridos del personal capacitado.	1 hora	mié 20-01-10	jue 21-01-10	14	INS
16	<b>INSTALACIÓN</b>	<b>0.63 días</b>	<b>jue 21-01-10</b>	<b>jue 21-01-10</b>	<b>5</b>	
17	Instalar ó habilitar el sistema operativo requerido en el equipo informático.	1 hora	jue 21-01-10	jue 21-01-10	14	MAN
18	Instalar la Máquina Virtual de Java	2 horas	jue 21-01-10	jue 21-01-10	17	MAN
19	Instalar aplicación EISI - FlowChart	2 horas	jue 21-01-10	jue 21-01-10	18	MAN
20	<b>VERIFICACIÓN Y PRUEBAS</b>	<b>2.5 días</b>	<b>jue 21-01-10</b>	<b>mar 26-01-10</b>	<b>16</b>	
21	Preparar casos de prueba para validar el funcionamiento del sistema.	8 horas	jue 21-01-10	vie 22-01-10		DIR
22	Aplicar pruebas.	2 horas	vie 22-01-10	vie 22-01-10	21	DIR,ENC
23	Monitorear los resultados de las pruebas.	2 horas	vie 22-01-10	lun 25-01-10	22	DIR,ENC
24	Realizar correcciones de ser necesario.	4 horas	lun 25-01-10	lun 25-01-10	23	DIR,ENC
25	Documentar las modificaciones ejecutadas en la aplicación en caso de ser necesario.	4 horas	lun 25-01-10	mar 26-01-10	24	DIR,ENC
26	<b>PUESTA EN MARCHA</b>	<b>0.19 días</b>	<b>mar 26-01-10</b>	<b>mar 26-01-10</b>	<b>20</b>	
27	Ejecutar la aplicación EISI - FlowChart.	0.5 horas	mar 26-01-10	mar 26-01-10		DIR,ENC,MAN
28	Monitorear resultados.	1 hora	mar 26-01-10	mar 26-01-10	27	DIR,ENC

Tabla 1.4 – Tabla de Programación

**DIAGRAMA DE GANTT PARA EL PLAN DE IMPLEMENTACIÓN**

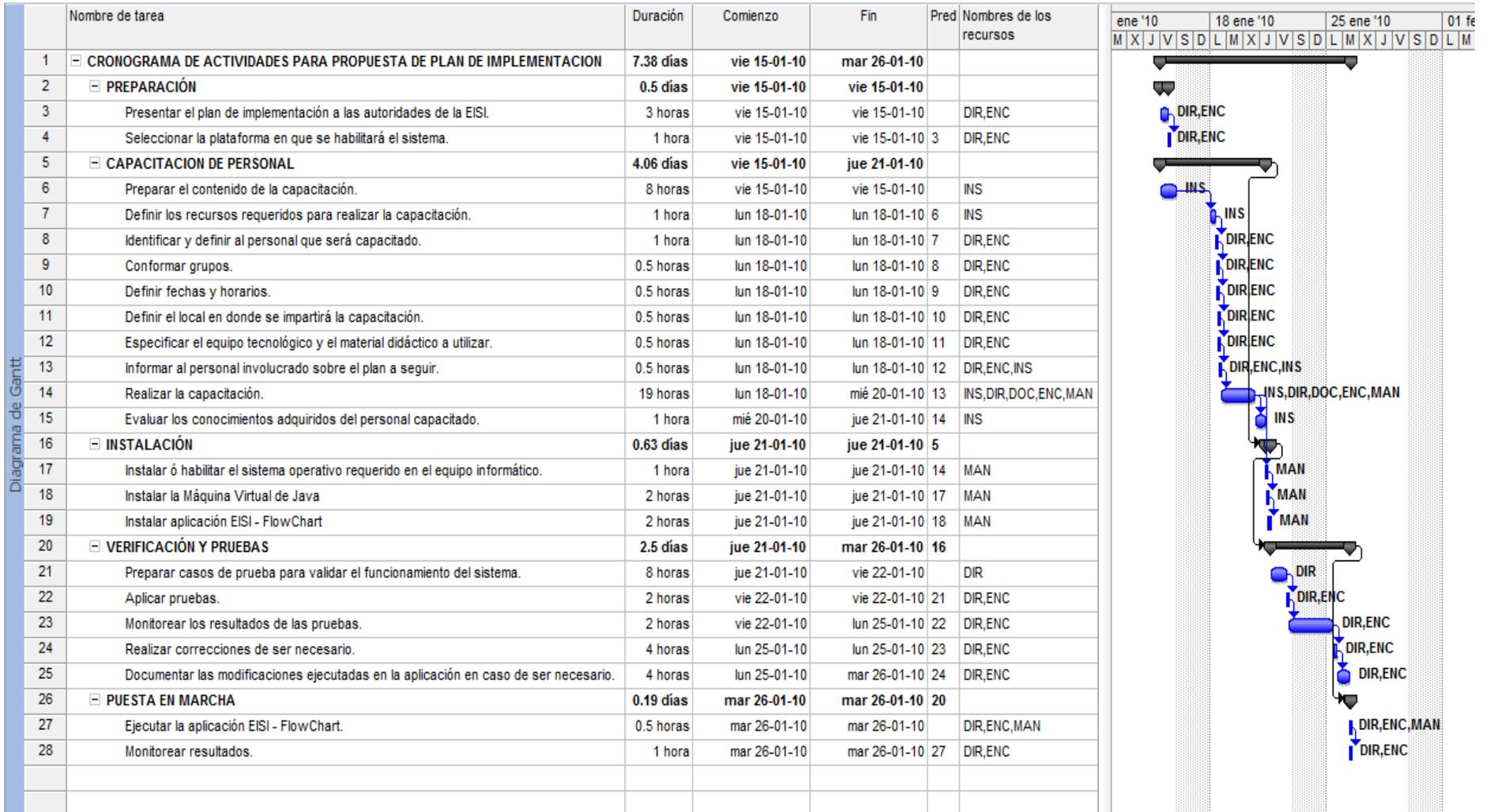


Figura 1.2 – Diagrama de Gantt para el Plan de Implementación

Actividades Principales	Duración (Horas)	Duración (Días)
Preparación	4.00	0.50
Capacitación de Personal	32.48	4.06
Instalación	5.00	0.63
Verificación y Pruebas	16.00	2.00
Puesta en Marcha	1.52	0.19
<b>Total de Tiempo</b>	<b>59 Horas</b>	<b>7.38 días</b>

Tabla 1.5 – Tiempo total de duración de la implementación de la herramienta de software

### 1.3.2.3 Recursos Financieros

#### 1.3.2.3.1 Costo de Recurso Humano

Para realizar el cálculo del costo de recurso humano, se hicieron las siguientes consideraciones:

- En base al cronograma de actividades el Director del Proyecto y el Encargado de Ejecución de Actividades estarán presentes durante la mayor parte del tiempo en que se ha previsto realizar la implementación de la herramienta de software, lo cual corresponde a 6 días hábiles, por la naturaleza propia de sus responsabilidades. Respecto al Instructor de Capacitaciones, se ha previsto que se requieren 0.5 días para que capacite a los Encargado Soporte de y Mantenimiento de Computadoras acerca del proceso de instalación de la herramienta de software. Además se necesitan 4.06 días para ejecutar la capacitación a docentes, incluyendo el tiempo de preparación de la misma.
- Se ha considerado que tanto el Director del Proyecto como el Encargado de Ejecución de Actividades, gozan del sueldo promedio de un Docente Universitario.
- Se obtuvieron los datos referentes al sueldo mensual que recibe un Docente Universitario<sup>30</sup>, un Instructor de Capacitaciones y un Encargado de Soporte y Mantenimiento de Computadoras, tanto en empresas privadas, como en instituciones públicas, luego se procedió a extraer el sueldo promedio mensual de cada uno, tal como se muestra en las Tablas 1.6, 1.7 y 1.8.

<sup>30</sup>[http://www.mh.gob.sv/portal/page/portal/MH\\_Finanzas/MH\\_PRESUPUESTO/PRESUPUESTOS\\_ESTADO/LPGE09F INAL/Salarios/LS3101-09.pdf](http://www.mh.gob.sv/portal/page/portal/MH_Finanzas/MH_PRESUPUESTO/PRESUPUESTOS_ESTADO/LPGE09F%20INAL/Salarios/LS3101-09.pdf)

Docentes	Sueldo Mensual de un Docente (\$)
Salario mínimo	\$ 1,100.00
Salario máximo	\$ 2,400.00
<b>Sueldo Promedio Mensual<sup>31</sup></b>	<b>\$ 1,750.00</b>

Tabla 1.6 - Cálculo del Sueldo Promedio Mensual de un Docente Universitario.

Empresa	Sueldo Mensual de un Instructor (\$)
Ministerio de Educación - MINED	\$ 600.00
Ministerio de Hacienda - MH	\$ 650.00
SALNET	\$ 400.00
CONSISA.	\$ 500.00
TACA	\$ 600.00
<b>Sueldo Promedio Mensual</b>	<b>\$ 550.00</b>

Tabla 1.7 - Cálculo del Sueldo Promedio Mensual de un Instructor de Capacitaciones.

Empresa	Sueldo Mensual de un Técnico de Mantenimiento de Sistemas (\$)
Ministerio de Educación - MINED	\$ 400.00
Ministerio de Hacienda - MH	\$ 400.00

<sup>31</sup> Los valores de los salarios mínimo y máximo se obtuvieron de los datos publicados por el Ministerio de Hacienda de la República en la siguiente dirección:

[http://www.mh.gob.sv/portal/page/portal/MH\\_Finanzas/MH\\_PRESUPUESTO/PRESUPUESTOS\\_ESTADO/LPGE09FINAL/Salarios/LS3101-09.pdf](http://www.mh.gob.sv/portal/page/portal/MH_Finanzas/MH_PRESUPUESTO/PRESUPUESTOS_ESTADO/LPGE09FINAL/Salarios/LS3101-09.pdf)

SALNET	\$ 250.00
CONSISA.	\$ 300.00
TACA	\$ 350.00
<b>Sueldo Promedio Mensual</b>	<b>\$ 340.00</b>

*Tabla 1.8 - Cálculo del Sueldo Promedio Mensual de un Encargado de Soporte y Mantenimiento de Computadoras.*

- Luego se procedió a definir la cantidad promedio de días laborales por semana, tal como se muestra en la Tabla 1.9.

Horas por Mes	
Días por semana (Empresa Privada)	6
Días por semana (Instituciones Públicas)	5
Días por semana (Promedio)	<b>5.5</b>

*Tabla 1.9 - Cantidad promedio de días por semana*

- Considerando los datos expuestos anteriormente, a continuación se muestra el Cálculo del Sueldo promedio por día, de un Docente Universitario, un Instructor de Capacitaciones y un Encargado de Soporte y Mantenimiento de Computadoras, a través de las Tablas 1.10, 1.11 y 1.12.

Sueldo Promedio por Día	
Sueldo promedio por mes.	\$1,750.00
Sueldo promedio por semana (4 semanas).	\$437.50
Sueldo promedio por día (5.5 días)	<b>\$79.55</b>

*Tabla 1.10 - Cálculo del Sueldo Promedio por día (para un Docente Universitario)*

Sueldo Promedio por Día	
Sueldo promedio por mes.	\$550.00
Sueldo promedio por semana (4 semanas).	\$138
Sueldo promedio por día (5.5 días)	<b>\$25.00</b>

Tabla 1.11 - Cálculo del Sueldo Promedio por día (para un Instructor de Capacitaciones)

Sueldo Promedio por Día	
Sueldo promedio por mes.	\$340.00
Sueldo promedio por semana (4 semanas).	\$85
Sueldo promedio por día (5.5 días)	<b>\$15.45</b>

Tabla 1.12 - Cálculo del Sueldo Promedio por día (para un Encargado de Soporte y Mantenimiento de Sistemas)

Finalmente se realizó el cálculo respectivo para obtener el costo total del recurso humano, lo cual se presenta en la Tabla 1.13.

Recurso Humano	Cantidad	Duración (días)	Costo/día	Costo Total
Director de proyecto	1	6	\$79.55	\$477.30
Encargado de Ejecución de Actividades	1	5	\$79.55	\$397.75
Docentes Universitarios	8	2.38	\$79.55	\$188.93
Instructor de Capacitaciones	1	3.56	\$25.00	\$89.06
Encargado de Soporte y Mantenimiento de Computadoras	1	3.06	\$15.45	\$47.32
<b>Total</b>				<b>\$1200.36</b>

Tabla 1.13 - Costo de Total del Recurso Humano

El costo total del recurso humano considerado en la implementación del de la herramienta de software EISI - FlowChart es de **\$1200.36**.

### 1.3.2.3.2 Costo de Materiales a Utilizar

Las consideraciones tomadas en cuenta para el cálculo del costo de los materiales requeridos en la implementación de la herramienta de software se presentan en las Tablas 1.14 y 1.15, tomando en cuenta previamente lo siguiente:

- ✓ Los materiales que se han tomado en cuenta, son aquellos que se requieren para llevar a cabo las capacitaciones correspondientes.
- ✓ El cálculo del costo de reproducción de los manuales que se utilizarán en la capacitación, se basó en la cantidad de páginas contenidas por cada uno de ellos y en el costo de una fotocopia \$ 0.02.

Recurso	Unidad de Medida	Cantidad	Precio Unitario (\$)	Costo Total (\$)
Manual del Usuario	Página	40	\$0.02	\$0.80
Manual Técnico	Página	30	\$0.02	\$0.60
Manual de Instalación	Página	20	\$0.02	\$0.40

Tabla 1.14 - Cálculo de Costo por Manual

En la Tabla 5.15 se presenta el costo total de recursos materiales requeridos para realizar la implementación de la herramienta de software.

Recurso	Cantidad <sup>32</sup>	Costo por unidad(\$) <sup>33</sup>	Costo Total (\$)
Manual del usuario	11	\$0.80	\$8.80
Manual Técnico	3	\$0.60	\$1.80

<sup>32</sup> Cantidad obtenida en base a la cantidad de usuarios en la EISI y la categorización de usuarios.

<sup>33</sup> Datos obtenidos de varias librerías y fotocopiadoras del centro de San Salvador

Manual de Instalación	3	\$0.40	\$1.20
Caja de 12 bolígrafos	1	\$1.55	\$3.10
Fólderes	24	\$0.15	\$3.60
Fastener	24	\$0.15	\$3.60
<b>TOTAL</b>			<b>\$22.10</b>

Tabla 1.15 – Cálculo del Costo Total de los Materiales a Utilizar.

### 1.3.2.3.3 Costo de Recursos Tecnológicos a Utilizar

La descripción de los requerimientos tecnológicos para poner en marcha la herramienta de software EISI - FlowChart, se encuentra en el Anexo 2.

El recurso tecnológico a utilizar se compone principalmente de los siguientes elementos:

- ✓ Software
- ✓ Hardware

A través del estudio realizado se verificó que la EISI dispone del hardware, software e infraestructura de red necesarios para poner en marcha la herramienta de software EISI - FlowChart, por lo tanto no se requiere de nuevas adquisiciones o compras, por lo consiguiente se ha estimado que no existen costos adicionales en recursos tecnológicos.

### 1.3.2.3.4 Costo Total

Para determinar el costo total, se consideró lo siguiente:

- ✓ Según el Ministerio de Economía de El Salvador, el porcentaje de imprevistos que se asigna en la elaboración de presupuestos para el desarrollo de proyectos es del 5%<sup>34</sup> sobre el costo total del proyecto.

La Tabla 1.16, muestra el cálculo del costo total de la implementación de la herramienta de software:

<sup>34</sup> Dato extraído del sitio: <http://www.minec.gob.sv/>

Categoría	Subtotal	TOTAL
Total de Costos del Recurso Humano		\$1,200.36
Total de Costos de Materiales a Utilizar		\$ 22.10
Total de Costos de Recursos Tecnológicos a Utilizar		\$ 0.00
<b>Subtotal</b>	<b>\$ 1,222.46</b>	
Imprevistos (5%)	\$ 61.12	
<b>COSTO TOTAL</b>		<b>\$1,283.58</b>

*Tabla 1.16 – Costo Total del Proyecto*

**COSTO TOTAL = \$1,283.58**

### 1.3.3 Aprobar los Resultados de la Planeación

Corresponde a la socialización del plan detallado del proyecto, la estructura organizativa, los roles y responsabilidades, el cronograma general y el plan de comunicaciones ante el grupo de responsables de las actividades descritas en dichos documentos, con el fin de precisar detalles o aclarar inquietudes.

Se realizan los cambios y ajustes pertinentes de los documentos y se hace oficial a las autoridades de la EISI.

#### 1.3.3.1 Recomendaciones Generales para el Desarrollo de esta Etapa

Es importante asegurar la disponibilidad de tiempo de dedicación al proyecto del personal involucrado. Se recomienda oficializar mediante acto administrativo, la delegación dada a estas personas y su tiempo de dedicación al proyecto.

El plan detallado se podrá ajustar durante la ejecución del mismo, pero se recomienda mantener la fecha de finalización como una referencia, para evaluar la eficiencia y eficacia del proyecto.

Es importante difundir toda la documentación de esta etapa a los miembros que conforman la estructura del proyecto. Ello con el fin de facilitar el control de las agendas y evitar que las actividades críticas de proyecto caigan en retraso o se incumpla con lo planeado.

## 2. Organización

Para que la implementación de la herramienta de software EISI – FlowChart se realice de manera adecuada, es necesario garantizar que las distintas actividades se ejecuten de acuerdo a la programación establecida, para ello se ha definido la estructura organizativa que se muestra en la Figura 2.1.



Figura 2.1 - Estructura Organizativa del equipo de trabajo

Una vez establecida la estructura organizativa, se procede a describir las funciones que tendrán asignadas cada uno de los responsables.

### 2.1 Descripción de Funciones y Perfiles de Puestos

A continuación se presentan las funciones y el perfil para cada uno de los puestos involucrados en la estructura organizativa del equipo encargado de la implementación del sistema informático. Dichos manuales de puestos han sido definidos y elaborados utilizando técnicas de sistemas y procedimientos<sup>35</sup>.

FUNCIONES Y PERFIL - Director de Proyecto	
<b>Nombre del puesto:</b>	Director del Proyecto
<b>Nivel superior:</b>	Jefe de Departamento de Programación y Manejo de Datos
<b>Subordinados:</b>	Encargado de Ejecución de Actividades
	Encargado de Soporte y Mantenimiento de Computadoras
	Instructor de Capacitaciones
<b>Área de adscripción:</b>	Departamento de Programación y Manejo de Datos.
<b>Función Genérica:</b>	

<sup>35</sup> Ingeniería de proyectos informáticos: Actividades y procedimientos, José Salvador Sánchez Garreta, 2003

<p>Coordinación y supervisión de la implementación de la herramienta EISI - FlowChart, garantizando que el desarrollo del mismo se realice acorde al plan definido, con los recursos requeridos (recurso humano, tiempo, materiales y dinero).</p>
<p><b>Funciones Específicas:</b></p> <ol style="list-style-type: none"> <li>1. Presentar el plan de implementación a las autoridades de la Escuela de Ingeniería de Sistemas Informáticos.</li> <li>2. Efectuar la selección del recurso humano capacitado para la implementación.</li> <li>3. Presentar informes a superiores y usuarios acerca del seguimiento y cumplimiento de los objetivos del proyecto.</li> <li>4. Identificar y planificar las actividades a realizar, los recursos requeridos, los plazos y los costes previstos.</li> <li>5. Informar y asignar las tareas a ejecutar a los miembros del proyecto.</li> <li>6. Apoyar en las actividades requeridas, a los responsables de las diferentes etapas.</li> <li>7. Resolver los problemas que se presenten.</li> <li>8. Monitorear y controlar el avance de las diversas actividades de implementación del proyecto.</li> <li>9. Tomar decisiones para llevar a cabo acciones correctivas ante las desviaciones que se presenten en el proyecto.</li> </ol>
<p><b>Especificación del puesto:</b></p> <p>Educación:</p> <ul style="list-style-type: none"> <li>• Graduado de Ingeniería de Sistemas Informáticos o carreras afines.</li> </ul> <p>Experiencia:</p> <ul style="list-style-type: none"> <li>• Mínima de un año en puestos afines.</li> </ul> <p>Habilidades:</p> <ul style="list-style-type: none"> <li>• Habilidad para hablar en público y comunicarse.</li> <li>• Capacidad de liderazgo.</li> <li>• Desarrollo de software.</li> </ul> <p>Requisitos adicionales al puesto:</p> <ul style="list-style-type: none"> <li>• Experiencia en planificación, organización y control de proyectos informáticos.</li> <li>• Experiencia en análisis, desarrollo e implementación de proyectos informáticos.</li> <li>• Buenas relaciones interpersonales.</li> </ul>

*Tabla 2.1 – Funciones y perfil del Director del Proyecto*

FUNCIONES Y PERFIL - Encargado de Ejecución de Actividades	
<b>Nombre del puesto:</b>	Encargado de Ejecución de Actividades
<b>Nivel superior:</b>	Director de Proyecto
<b>Subordinados:</b>	Sin subordinados
<b>Área de adscripción:</b>	Departamento de Programación y Manejo de Datos.
<b>Función Genérica:</b>	

Encargado de la supervisión y ejecución de las actividades requeridas para llevar a cabo el plan de implementación del sistema, en el periodo establecido y con los recursos asignados, así también es el responsable de los aspectos técnicos involucrados en el desarrollo del mismo.

**Funciones Específicas:**

1. Etapa de capacitación de personal:
  1. Asegurar que todas las actividades del plan de capacitación se lleven a cabo y registrar el avance y seguimiento de éste.
  
2. Etapa de Verificación y Pruebas:
  1. Controlar el correcto funcionamiento de la herramienta de software EISI - FlowChart.
  2. Informar al Jefe del Departamento de Programación y Manejo de Datos sobre las dificultades presentadas.
  3. Monitorear la realización de pruebas sobre la herramienta.
  4. Verificar el correcto funcionamiento de la herramienta
  5. Aplicar correcciones a la herramienta de software EISI - FlowChart, en caso de que surja algún error.
  6. Presentar informe final de resultados obtenidos al Jefe del Departamento de Programación y Manejo de Datos.
  
3. Etapa de puesta en marcha:
  1. Utilización de la herramienta para el desarrollo de las asignaturas Introducción a la Informática y Programación I en los Laboratorios de Cómputo indicados.
  2. Sustituir totalmente el sistema manual por el sistema de información.
  3. Presentar informe final de resultados obtenidos al Jefe del Departamento de Programación y Manejo de Datos.
  4. Dar visto bueno para la puesta en marcha de la herramienta EISI - FlowChart.

**Especificación del puesto:**

Educación:

- Graduado de Ingeniería de Sistemas Informáticos o carreras afines.

Experiencia:

- Mínimo un año en puestos afines.
- Instalación y configuración de sistemas operativos en plataformas Linux, Solaris, Mac OS y Windows.
- Conocimientos intermedios en Lenguaje de Programación Java.
- Experiencia en implementación de sistemas.

Habilidades:

- Capacidad de análisis y síntesis.
- Habilidad para trabajar en equipo.

Requisitos adicionales al puesto:

- Experiencia en análisis, desarrollo e implementación de proyectos informáticos.
- Buenas relaciones interpersonales.
- Conocimientos básicos de inglés técnico.

*Tabla 2.2 – Funciones y perfil del Encargado de Ejecución de Actividades*

FUNCIONES Y PERFIL – Docentes Universitarios	
<b>Nombre del puesto:</b>	Docente Universitario
<b>Nivel superior:</b>	Director de Proyecto
<b>Subordinados:</b>	Sin subordinados
<b>Área de adscripción:</b>	Departamento de Programación y Manejo de Datos.
<b>Función Genérica:</b>	
Encargados de impartir clases teóricas y prácticas a los estudiantes universitarios de los primeros años de la carrera de Ingeniería de Sistemas Informáticos. Participantes de la capacitación para la utilización de la herramienta en aras de utilizar la herramienta de software EISI – FlowChart para mecanizar el proceso solucionador de problemas y facilitar su aplicación.	
<b>Funciones Específicas:</b>	
<ol style="list-style-type: none"> <li>1. Capacitación de personal: Participar en la capacitación para la correcta utilización de la herramienta EISI – FlowChart para aplicarla en el desarrollo de la metodología del proceso solucionador de problemas.</li> </ol>	
<b>Especificación del puesto:</b>	
Educación:	
<ul style="list-style-type: none"> <li>•Graduado de Ingeniería de Sistemas Informáticos o carreras afines.</li> </ul>	
Experiencia:	
<ul style="list-style-type: none"> <li>•Mínimo un año en puestos afines.</li> </ul>	
Conocimientos:	
<ul style="list-style-type: none"> <li>• Metodología del proceso solucionador de problemas</li> <li>•Conocimientos básicos de la arquitectura de Java. (no indispensable)</li> </ul>	

*Tabla 2.3 – Funciones y perfil del Encargado de Ejecución de Actividades*

FUNCIONES Y PERFIL - Instructor de capacitaciones	
<b>Nombre del puesto:</b>	Instructor de Capacitaciones
<b>Nivel superior:</b>	Encargado de Ejecución de Actividades
<b>Subordinados:</b>	Sin subordinados
<b>Área de adscripción:</b>	Departamento de Programación y Manejo de Datos
<b>Función Genérica:</b>	
Preparar el plan de capacitación requerido para asegurar que los usuarios de la herramienta de software, adquieran los conocimientos necesarios para utilizarlo de manera adecuada.	
<b>Funciones Específicas:</b>	
Etapa de Capacitación:	
<ol style="list-style-type: none"> <li>1. Elaborar el plan de capacitación requerido definiendo aspectos tales como: Personal que será capacitado, grupos, local, fechas, horarios, uso de equipos y material didáctico.</li> <li>2. Gestionar la adquisición de los recursos necesarios para realizar las capacitaciones programadas.</li> <li>3. Asegurarse que todas las actividades del plan de capacitación se lleven a cabo y registrar el avance de éstas.</li> <li>4. Capacitar al personal requerido que utilizará la herramienta de software.</li> <li>5. Evaluar el nivel de aprendizaje de los usuarios.</li> <li>6. Informar al Jefe del Departamento de Programación y Manejo de Datos sobre los avances o dificultades presentadas, así como también de los resultados de la capacitación.</li> </ol>	
<b>Especificación del puesto:</b>	
Educación:	
<ul style="list-style-type: none"> <li>•Estudiante de 5º año o egresado de la carrera Ingeniería de Sistemas Informáticos o carreras afines.</li> </ul>	
Experiencia:	
<ul style="list-style-type: none"> <li>•Mínimo un año en puestos afines.</li> </ul>	
Habilidades:	
<ul style="list-style-type: none"> <li>•Habilidad para hablar en público y comunicarse.</li> <li>•Facilidad para comunicar sus conocimientos a otros.</li> <li>•Aprendizaje rápido de manejo de paquetes de software.</li> <li>•Habilidad para trabajar en equipo.</li> </ul>	
Requisitos adicionales al puesto:	
<ul style="list-style-type: none"> <li>•Conocimiento de técnicas de capacitación.</li> <li>•Responsable y con iniciativa.</li> </ul>	

- Buenas relaciones interpersonales.

*Tabla 2.4 – Funciones y perfil del Instructor de capacitaciones*

FUNCIONES Y PERFIL – Encargado de Soporte y Mantenimiento de Computadoras	
<b>Nombre del puesto:</b>	Encargado de Soporte y Mantenimiento de Computadoras
<b>Nivel superior:</b>	Encargado de Ejecución de Actividades
<b>Subordinados:</b>	Sin subordinados
<b>Área de adscripción:</b>	Departamento de Programación y Manejo de Datos
<b>Función Genérica:</b> Encargado del proceso de instalación en Laboratorios de Cómputo de la EISI.	
<b>Funciones Específicas:</b>  Etapa de Preparación e Instalación: <ol style="list-style-type: none"> <li>1. Instalar ó habilitar el sistema operativo en el equipo informático asignado.</li> <li>2. Instalar la Maquina Virtual de Java.</li> <li>3. Instalar la herramienta de software EISI - FlowChart</li> <li>4. Monitorear avance de las actividades correspondientes a la preparación e instalación</li> <li>5. Presentar informe final de resultados obtenidos al Jefe del Departamento de Programación y Manejo de Datos.</li> </ol>	
<b>Especificación del puesto:</b> Educación: <ul style="list-style-type: none"> <li>• Estudiante de 5º año o egresado de la carrera Ingeniería de Sistemas Informáticos o carreras afines.</li> </ul> Experiencia: <ul style="list-style-type: none"> <li>• Deseable mínima de 1 año en Soporte y Mantenimiento de Computadoras.</li> </ul> Habilidades: <ul style="list-style-type: none"> <li>• Aprendizaje rápido de manejo de paquetes de software.</li> <li>• Habilidad para trabajar en equipo.</li> </ul> Requisitos adicionales al puesto: <ul style="list-style-type: none"> <li>• Responsable y con iniciativa.</li> <li>• Buenas relaciones interpersonales.</li> </ul>	

*Tabla 2.5 – Funciones y perfil del Encargado de Soporte y Mantenimiento de Computadoras*

## 3. Control

### 3.1 Canales de Comunicación

El establecimiento de un plan de comunicaciones es una herramienta necesaria para que la herramienta de software se construya con la participación activa de todos los responsables e involucrados. Este plan debe además servir de plataforma para que, una vez implementada la herramienta de software, se garantice una comunicación interna que contribuya al mantenimiento y mejoramiento de la misma<sup>36</sup>.

Generalmente el plan de comunicaciones se estructura bajo el modelo Mensaje-Emisor-Medio-Frecuencia-Perceptor-Respuesta, que describe la secuencia básica de cualquier proceso de comunicación. Se identifican así sus elementos constitutivos:

1. Mensaje: Aquello que se quiere comunicar.
2. Emisor: Responsable de la comunicación, (quien comunica el mensaje)
3. Medio: Mecanismo a través del cual el emisor comunica el mensaje.
4. Frecuencia: Periodicidad con la que el mensaje debe ser comunicado.
5. Receptor: Aquel a quien se le va a comunicar el mensaje.
6. Respuesta: Mecanismo a través del cual el perceptor retroalimenta al emisor acerca de su entendimiento y análisis del mensaje recibido.

### 3.2 Documentos de Control

Para ejecutar correctamente el plan de implementación, se han definido una serie de documentos de control que tienen como objetivo verificar que se cumplan las actividades programadas, dichos documentos se han estructurado de la siguiente manera:

1. Control de actividades de preparación e instalación
  - Formulario de control de instalación de software
  - Formulario de control de instalación de hardware y/o recursos.
2. Control de actividades generales:
  - Formulario de control de ejecución de actividades.
  - Formulario de control de gastos del proyecto.
  - Formulario de control de asistencia.

---

<sup>36</sup> “Como Elaborar un Plan de Comunicaciones” – Extraído del sitio: <http://www.dafp.gov.co/Documentos/guia%20de%20planeacion.pdf>

- Formulario de control de avance del Proyecto.

### 3. Control de Capacitaciones

- Formulario de control de asistencia.
- Formulario de control de evaluación de capacitaciones.

A continuación se muestran los formularios correspondientes, para cada uno se presenta su estructura de contenido y su esquema.

### 3.2.1 Control de Actividades de Preparación e Instalación

#### 3.2.1.1 Formulario de Control de Instalación de Software

La Tabla 3.1 muestra la estructura del contenido de la información y la Tabla 5.22 el esquema resultante.

<b>Nombre de formulario:</b>	Control de instalación de software.
<b>Objetivo:</b>	Elaborar un avance de las actividades realizadas respecto a la instalación del software necesario para el correcto funcionamiento de la herramienta de software.
<b>Quién lo elabora:</b>	Encargado de Ejecución de Actividades.
<b>Quién lo recibe:</b>	Director del Proyecto.
<b>Datos a considerar</b>	1. Número correlativo del formulario.
	2. Cantidad de páginas que contiene el formulario.
	3. Fecha en que se realiza el control.
	4. Nombre del proyecto.
	5. Fecha de inicio (Desde).
	6. Fecha de finalización (Hasta).
	7. Nombre del software que se instalará.
	8. Fases en la que se encuentra la aplicación por ejemplo: instalada, configurada o en pruebas.
	9. Observaciones.
	10. Nombre y firma del Encargado de Ejecución de Actividades.
<b>Frecuencia:</b>	Semanal durante la realización de la etapa.
<b>Volumen:</b>	2 copias.
<b>Cómo se elabora el formulario:</b>	Al final de la semana el encargado de Ejecución de Actividades anota en el formulario el estado de las diferentes aplicaciones necesarias para dejar en operación la herramienta de software.
	Elabora dos copias del formulario, entrega una copia al Director del Proyecto y la otra copia la archiva para registro interno.

Tabla 3.1 – Estructura del formulario de control de instalación de software.

	<b>ESCUELA DE INGENIERÍA DE SISTEMAS INFORMÁTICOS</b>		No. Correlativo: Página: ____ de ____	
	<b>CONTROL DE INSTALACIÓN DE SOFTWARE</b>		Fecha: ____/____/____ <u>  x  </u>	
Nombre del Proyecto: _____ x				
Desde: ____/____/____ c Hasta: ____/____/____ v				
No.	Nombre del Software	Fase		
		Instalación	Configuración	Pruebas
Observaciones:		Nombre del encargado:		
		Firma:		

Tabla 3.2 – Esquema del formulario de Control de instalación de software.

### 3.2.1.2 Formulario de Control de Instalación de Hardware y/o Recursos

La Tabla 3.3 muestra la estructura del contenido de la información y la Tabla 3.4 el esquema resultante.

<b>Nombre de formulario:</b>	Control de instalación de hardware y/o recursos.
<b>Objetivo:</b>	Elaborar un informe sobre el hardware y/o recursos que se han instalado.
<b>Quién lo elabora:</b>	Encargado de ejecución de actividades.
<b>Quién lo recibe:</b>	Director del proyecto.
<b>Datos a considerar</b>	1. Número correlativo del formulario.
	2. Cantidad de páginas que contiene el formulario.
	3. Fecha en que se realiza el control.
	4. Nombre del proyecto.
	5. Fecha de inicio (Desde).
	6. Fecha de finalización (Hasta).
	7. Nombre del software y/o recursos que se instalarán.
	8. Fases en la que se encuentra la aplicación por ejemplo: instalada, configurada o en pruebas.
	9. Observaciones.
	10. Nombre y firma del Encargado de Ejecución de Actividades.

<b>Frecuencia:</b>	Se realiza una vez terminada la fase de preparación e instalación del plan de implementación.
<b>Volumen:</b>	2 copias.
<b>Cómo se elabora el formulario:</b>	Al finalizar el periodo asignado para realizar todas las actividades relacionadas con la preparación e instalación, el encargado de ejecución de actividades anota en el formulario el avance logrado en cada actividad.
	Elabora dos copias del formulario, entrega una copia al director del proyecto y la otra copia la archiva para registro interno.

Tabla 3.3 – Estructura del formulario de control de instalación de hardware y/o recursos.

	<b>ESCUELA DE INGENIERÍA DE SISTEMAS INFORMÁTICOS</b>	No. Correlativo: Página: ____ de ____ Fecha: __/__/__ X		
	<b>CONTROL DE INSTALACIÓN DE SOFTWARE Y/O RECURSOS</b>			
Nombre del Proyecto: _____ x Desde: __/__/__ c Hasta: __/__/__ v				
No.	Nombre del Software	Fase		
		Instalación	Configuración	Pruebas
Observaciones:		Nombre del encargado:		
		Firma:		

Tabla 3.4 – Esquema del formulario de Resultados de instalación de hardware y otros recursos.

### 3.2.2 Control de Actividades Generales.

#### 3.2.2.1 Formulario de Control de Ejecución de Actividades

La Tabla 3.5 muestra la estructura del contenido de la información y la Tabla 3.6 el esquema resultante.

<b>Nombre de formulario:</b>	Control de Ejecución de Actividades.
<b>Objetivo:</b>	Informar al Director del Proyecto sobre la situación actual de la fase de implementación de la herramienta de software.

<b>Quién lo elabora:</b>	Encargado de Ejecución de Actividades.
<b>Quién lo recibe:</b>	Director del proyecto.
<b>Datos a considerar:</b>	1. Número correlativo del formulario.
	2. Cantidad de páginas que contiene el formulario.
	3. Fecha en que se realiza el control.
	4. Nombre del proyecto.
	5. Fecha de inicio (Desde).
	6. Fecha de finalización (Hasta).
	7. Nombre de la actividad.
	8. Duración de la actividad expresada en días.
	9. Responsable de esa actividad.
	10. Estado en que se encuentra la actividad (por ejemplo: en proceso o finalizada).
	11. Observaciones.
	12. Nombre y firma del Encargado de Ejecución de Actividades.
<b>Frecuencia:</b>	Diario.
<b>Volumen:</b>	2 copias.
<b>Cómo se elabora el formulario:</b>	Al final de la semana el encargado de ejecución de actividades anota en el formulario todas las actividades relacionadas con la implementación de la herramienta de software, realizadas por el mismo y sus subalternos.
	Elabora dos copias del formulario, entrega una copia al Director del Proyecto y la otra copia la archiva para registro interno.

Tabla 3.5 – Estructura del formulario de Control de ejecución de actividades

	<b>ESCUELA DE INGENIERÍA DE SISTEMAS INFORMÁTICOS</b>		No. Correlativo: Página: ____ de ____  Fecha: ___/___/___ x	
	<b>CONTROL DE EJECUCIÓN DE ACTIVIDADES</b>			
Nombre del Proyecto: _____ x				
Desde: ___/___/___ c Hasta: ___/___/___ v				
No.	Actividad	Duración (días)	Responsable	Estado
Observaciones:		Nombre del encargado:		
		Firma:		

Tabla 3.6 – Esquema del formulario de Control de Ejecución de Actividades

### 3.2.2.2 Formulario de Control de Gastos del Proyecto

La Tabla 3.7 muestra la estructura del contenido de la información y la Tabla 3.8 el esquema resultante.

<b>Nombre de formulario:</b>	Control de gastos del proyecto.
<b>Objetivo:</b>	Informar al Director del Proyecto sobre los gastos en los que se ha incurrido durante una semana.
<b>Quién lo elabora:</b>	Encargado de Ejecución de Actividades.
<b>Quién lo recibe:</b>	Director del Proyecto.
<b>Datos a considerar</b>	1. Número correlativo del formulario.
	2. Cantidad de páginas que contiene el formulario.
	3. Fecha en que se realiza el control.
	4. Nombre del proyecto.
	5. Fecha de inicio (Desde).
	6. Fecha de finalización (Hasta).
	7. Nombre de cada actividad o recurso empleado.
	8. Costo de cada actividad o recurso empleado.
	9. Total de costos.
	10. Observaciones.
	11. Nombre y firma del Encargado de Ejecución de Actividades.
<b>Frecuencia:</b>	Diario
<b>Volumen:</b>	2 copias.
<b>Cómo se elabora el formulario:</b>	Al final de la semana el encargado de ejecución de actividades anota en el formulario todos los gastos en los que se ha incurrido durante la semana, en la implementación de la herramienta de software.
	Elabora dos copias del formulario, entrega una copia al Director del Proyecto y la otra copia la archiva para registro interno.

Tabla 3.7 – Estructura del formulario de Control de gastos del proyecto.

	<b>ESCUELA DE INGENIERÍA DE SISTEMAS INFORMÁTICOS</b> <b>GASTOS DEL PROYECTO</b>	No. Correlativo: Página: ____ de ____ Fecha: ____ / ____ / ____ x
	Nombre del Proyecto: _____ x Desde: ____ / ____ / ____ c Hasta: ____ / ____ / ____ v	
No.	Actividad / Recurso	Costo (\$)

	<b>TOTAL DE COSTOS</b>	
Observaciones:	Nombre del encargado:	
	Firma:	

Tabla 3.8 – Esquema del formulario de Control de gastos del proyecto.

### 3.2.2.3 Formulario de Control de Asistencia

La Tabla 3.9 muestra la estructura del contenido de la información y la Tabla 3.10 el esquema resultante.

<b>Nombre de formulario:</b>	Control de asistencia.
<b>Objetivo:</b>	Registrar las entradas y salidas de los miembros del proyecto de implementación del sistema informático.
<b>Quién lo elabora:</b>	Encargado de Ejecución de Actividades.
<b>Quién lo recibe:</b>	Director del Proyecto.
<b>Datos a considerar</b>	1. Número correlativo del formulario.
	2. Cantidad de páginas que contiene el formulario.
	3. Fecha en que se realiza el control.
	4. Nombre del proyecto.
	5. Fecha de inicio (Desde).
	6. Fecha de finalización (Hasta).
	7. Nombre del miembro del proyecto.
	8. Actividad a realizar.
	9. Hora de inicio de labores.
	10. Hora de finalización de labores.
	11. Lugar donde desempeñara sus labores.
	12. Firma del miembro del proyecto.
	13. Observaciones.
	14. Nombre y firma del Encargado de Ejecución de Actividades.
<b>Frecuencia:</b>	Diario.
<b>Volumen:</b>	1 copia.
<b>Cómo se elabora el formulario:</b>	Todos los días, los miembros del proyecto de implementación se anotan en la hoja de asistencia, indicando, la hora de llegada, la actividad a realizar, el lugar y la hora de salida.
	Al final del día, el encargado de ejecución de actividades recoge la hoja de asistencia y la archiva.

Tabla 3.9 – Estructura del formulario de Control de Asistencia.

		<b>ESCUELA DE INGENIERÍA DE SISTEMAS INFORMÁTICOS</b>			No. Correlativo: Página: ____ de ____ Fecha: ____ / ____ / ____ x	
<b>CONTROL DE ASISTENCIA</b>		Nombre del Proyecto: _____ x Desde: ____ / ____ / ____ c Hasta: ____ / ____ / ____ v				
No.	Nombre	Actividad a Realizar	Hora Inicio	Hora Fin	Lugar	Firma
Observaciones:			Nombre del encargado:			
			Firma:			

Tabla 3.10 – Esquema del formulario de Control de Asistencia.

### 3.2.2.4 Formulario de Control de Avance del Proyecto.

La Tabla 3.11 muestra la estructura del contenido de la información y la Tabla 3.12 el esquema resultante.

<b>Nombre de formulario:</b>	Control de avance del proyecto de implantación.
<b>Objetivo:</b>	Elaborar un avance de las actividades realizadas del proyecto de implantación de la herramienta de software.
<b>Quién lo elabora:</b>	Encargado de Ejecución de Actividades.
<b>Quién lo recibe:</b>	Director del Proyecto.
	1. Número correlativo del formulario.
	2. Cantidad de páginas que contiene el formulario.
	3. Fecha en que se realiza el control.
	4. Nombre del proyecto.
<b>Datos a considerar</b>	5. Fecha de inicio (Desde).
	6. Fecha de finalización (Hasta).
	7. Nombre de la actividad.
	8. Porcentaje de avance semanal.
	9. Total de porcentaje de avance.
	10. Observaciones.
	11. Nombre y firma del Encargado de Ejecución de Actividades.
<b>Frecuencia:</b>	Diario.

<b>Volumen:</b>	2 copias.
<b>Cómo se elabora el formulario:</b>	Al final de la semana el encargado de ejecución de actividades anota en el formulario el avance logrado en cada actividad y expresado en porcentajes.
	Elabora dos copias del formulario, entrega una copia al director del proyecto y archiva para registro interno la otra copia.

Tabla 3.11 – Estructura del formulario de Control de avance del proyecto.

	<b>ESCUELA DE INGENIERÍA DE SISTEMAS INFORMÁTICOS</b>  <b>CONTROL DE AVANCE DEL PROYECTO</b>	No. Correlativo: Página: ____ de ____  Fecha: __/__/____							
	Proyecto: _____ x Desde: ____/____/____ c Hasta: ____/____/____ v								
No	Actividad	Porcentaje de Avance							
		Día 1	Día 2	Día 3	Día 4	Día 5	Día 6	Día n	Total de Avance
<b>TOTAL DE AVANCE:</b>									
Observaciones:		Nombre encargado:							
		Firma:							

Tabla 3.12 – Esquema del formulario de Control de avance del proyecto.

### 3.2.3 Control de Capacitaciones

#### 3.2.3.1 Formulario de Control de Asistencia a Capacitaciones

La Tabla 3.13 muestra la estructura del contenido de la información y la Tabla 3.14 el esquema resultante.

<b>Nombre de formulario:</b>	Control de asistencia a capacitaciones.
<b>Objetivo:</b>	Elaborar un informe sobre la asistencia de los usuarios a las capacitaciones relacionadas con la herramienta de software.
<b>Quién lo elabora:</b>	Instructor de Capacitaciones.
<b>Quién lo recibe:</b>	Encargado de Ejecución de Actividades.
<b>Datos a considerar</b>	1. Número correlativo del formulario.
	2. Cantidad de páginas que contiene el formulario.
	3. Fecha en que se realiza el control.
	4. Nombre del proyecto.
	5. Fecha de inicio (Desde).
	6. Fecha de finalización (Hasta).
	7. Nombre del usuario capacitado.
	8. Hora de entrada.
	9. Hora de salida.
	10. Firma del Usuario.
	11. Observaciones.
	12. Nombre y firma del Instructor de Capacitaciones.
<b>Frecuencia:</b>	Diariamente durante la fase de capacitaciones.
<b>Volumen:</b>	1 copia.
<b>Cómo se elabora el formulario:</b>	Al presentarse los administradores y los usuarios de la herramienta de software a sus respectivas capacitaciones, se deben anotar en el formulario “Control de asistencia a capacitaciones”.
	El capacitador recoge la hoja de asistencia de las capacitaciones y se la entrega al Encargado de Ejecución de Actividades.

Tabla 3.13 – Estructura del formulario de Control de asistencia a capacitaciones.

	<b>ESCUELA DE INGENIERÍA DE SISTEMAS INFORMÁTICOS</b>	No. Correlativo: Página: ____ de ____
	<b>CONTROL DE ASISTENCIA A CAPACITACIONES</b>	Fecha: ____ / ____ / ____ x
Nombre del Proyecto: _____ x		
Desde: ____ / ____ / ____ c Hasta: ____ / ____ / ____ v		

No.	Nombre	Hora de Entrada	Hora de Salida	Firma
Observaciones:		Nombre del encargado:		
		Firma:		

Tabla 3.14 – Esquema del formulario de Control de asistencia a capacitaciones.

### 3.2.3.2 Formulario de Control de Evaluación de Capacitaciones

La Tabla 3.15 muestra la estructura del contenido de la información y la Tabla 3.16 el esquema resultante.

<b>Nombre de formulario:</b>	Resultado de las capacitaciones.
<b>Objetivo:</b>	Elaborar un informe de los resultados obtenidos de las evaluaciones en las capacitaciones impartidas, de la fase de implementación de la herramienta de software.
<b>Quién lo elabora:</b>	Instructor de Capacitaciones.
<b>Quién lo recibe:</b>	Encargado de ejecución de actividades.
<b>Datos a considerar</b>	1. Número correlativo del formulario.
	2. Cantidad de páginas que contiene el formulario.
	3. Fecha en que se realiza el control.
	4. Nombre del proyecto.
	5. Fecha de inicio (Desde).
	6. Fecha de finalización (Hasta).
	7. Nombre del usuario capacitado.
	8. Nota correspondiente a la asistencia del usuario a las capacitaciones.
	9. Nota de la prueba realizada al usuario.
	10. Resultado de la nota global de la capacitación.
	11. Observaciones.
	12. Nombre y firma del Instructor de Capacitaciones.
<b>Frecuencia:</b>	Al final de cada curso de capacitación.
<b>Volumen:</b>	1 copia.
<b>Cómo se elabora el formulario:</b>	Al finalizar el periodo asignado para realizar todas las actividades relacionadas con las capacitaciones, el Instructor de Capacitaciones llena el formulario requerido.

	Elaborar una copia del formulario y entregarla al Encargado de Ejecución de Actividades.
--	--

Tabla 3.15 – Estructura del formulario de control de evaluación de capacitaciones.

	<b>ESCUELA DE INGENIERÍA DE SISTEMAS INFORMÁTICOS</b>	No. Correlativo:			
	<b>CONTROL DE EVALUACIÓN DE CAPACITACIONES</b>	Página: ____ de ____			
		Fecha: ____ / ____ / ____ x			
Nombre del Proyecto: _____ x					
Desde: ____ / ____ / ____ c Hasta: ____ / ____ / ____ v					
No	Nombre	Evaluación			Observaciones
		Nota Asistencia	Nota Prueba	Resultado	
Observaciones:		Nombre encargado:			
		Firma:			

Tabla 3.16 – Esquema del formulario de control de evaluación de capacitaciones.

## 4. Plan de Capacitación

### 4.1. Contenido de la Capacitación

La organización y la efectividad de las sesiones de capacitación mejoran considerablemente si la sesión se divide en módulos específicos planeados. Las sesiones de formato libre pueden fácilmente dejar de cubrir los temas a tratar o las actividades grupales requeridas y por lo tanto es posible que no se cumpla con los objetivos de la sesión.

Las sesiones de capacitación se pueden controlar mejor si se dividen en módulos, cada uno de los cuales representa un bloque específico de aprendizaje que los participantes deben adquirir. La organización de las sesiones en módulos ayuda a los capacitadores ya que guarda una estructura para la distribución del tiempo y ofrece objetivos específicos identificables de

capacitación y aprendizaje que sirven de parámetro para evaluar los logros de los participantes y el desempeño del instructor de capacitaciones.

El contenido de la capacitación se impartirá en base a los módulos que conforman la herramienta EISI - FlowChart, y de acuerdo a la categoría cada usuario. Los cursos que se han definido, se listan a continuación:

- ✓ Curso 1: Generalidades y módulos principales
- ✓ Curso 2: Módulos principales
- ✓ Curso 3: Módulos adicionales
- ✓ Curso 4: Módulos adicionales
- ✓ Curso 5: Proceso de instalación

En la Tabla 4.1 se presenta una breve descripción del contenido de los cursos para la capacitación del personal que operará la herramienta de software:

Nombre del Curso	Temas	Descripción
Curso 1: Proceso de instalación	<ol style="list-style-type: none"> <li>1. Requerimientos de hardware en el equipo informático para la instalación.</li> <li>2. Requerimientos de software en el equipo informático para la instalación.</li> <li>3. Aplicaciones a utilizar.</li> <li>4. Forma de instalación.</li> <li>5. Desarrollo de ejemplo de proceso de instalación completo.</li> </ol>	Se deben dar a conocer los requerimientos mínimos de hardware y software en el equipo informático disponible para realizar la instalación de la herramienta. Las aplicaciones necesarias con sus versiones y generalidades de las mismas. Además, se deben describir los pasos para realizar la instalación de la herramienta EISI – FlowChart en equipos informáticos individuales.
Curso 2: Generalidades y Módulos principales	<ol style="list-style-type: none"> <li>1. Generalidades sobre la aplicación</li> <li>2. Estructura lógica de la aplicación</li> <li>3. Ubicación de opciones y funcionalidades</li> </ol>	El curso se divide en dos partes: una parte teórica y una parte práctica. En la parte teórica se presentan las generalidades de la herramienta EISI – FlowChart, es decir, cuales son las opciones, ubicación de las mismas en

Nombre del Curso	Temas	Descripción
	<ol style="list-style-type: none"> <li>4. Planteamiento del problema                             <ol style="list-style-type: none"> <li>4.1. Explicación de funcionamiento</li> <li>4.2. Validaciones</li> <li>4.3. Desarrollo de ejemplo</li> </ol> </li> <li>5. Análisis de variables                             <ol style="list-style-type: none"> <li>5.1. Explicación de funcionamiento                                     <ol style="list-style-type: none"> <li>5.1.1. Definición de variables de salida</li> <li>5.1.2. Definición de variables de entrada</li> <li>5.1.3. Definición de restricciones</li> <li>5.1.4. Definición de constantes</li> <li>5.1.5. Definición de variables de proceso</li> <li>5.1.6. Definición de procesos</li> </ol> </li> <li>5.2. Validaciones</li> <li>5.3. Desarrollo de ejemplo</li> </ol> </li> </ol>	<p>la herramienta y funcionamiento de cada una. Además una descripción de la forma correcta de utilización desarrollando ejemplos. Durante la clase práctica, se deben desarrollar los ejemplos propuestos para que los docentes apliquen los conocimientos vistos durante la parte teórica. En este curso únicamente se darán a conocer las primeras dos etapas del proceso solucionador de problemas: Planteamiento del problema y Análisis del problema, los cuales cubren la manera de completar cada opción, las condiciones que se deben cumplir, y las restricciones propias de cada elemento.</p>
<p>Curso 3: Módulos principales</p>	<ol style="list-style-type: none"> <li>1. Repaso de temas anteriores</li> <li>2. Construcción de flujograma                             <ol style="list-style-type: none"> <li>2.1. Explicación de funcionamiento</li> <li>2.2. Desarrollo de ejemplos                                     <ol style="list-style-type: none"> <li>2.2.1. Construir flujograma usando estructuras secuenciales</li> <li>2.2.2. Construir flujograma usando estructuras selectivas</li> </ol> </li> </ol> </li> </ol>	<p>Este curso se divide en dos partes: parte teórica y parte práctica.</p> <p>En la parte teórica se definen los conceptos utilizados para la construcción de un diagrama de flujo, las condiciones y pruebas de la funcionalidad de la herramienta aplicando ejemplos sencillos. En la parte práctica se debe desarrollar un ejemplo para construir flujogramas con cada una de las estructuras lógicas secuenciales, selectivas y repetitivas.</p>

Nombre del Curso	Temas	Descripción
	<ul style="list-style-type: none"> <li>2.2.3. Construir flujograma usando estructuras cíclicas</li> <li>2.2.4. Construir flujograma usando modularidad</li> <li>2.3. Ejecución de flujograma                             <ul style="list-style-type: none"> <li>2.3.1. Ejecutar flujograma usando estructuras secuenciales</li> <li>2.3.2. Ejecutar flujograma usando estructuras selectivas</li> <li>2.3.3. Ejecutar flujograma usando estructuras cíclicas</li> <li>2.3.4. Ejecutar flujograma usando modularidad</li> </ul> </li> <li>2.4.</li> <li>3. Desarrollo de ejemplo del proceso solucionador de problemas completo</li> </ul>	
<p>Curso 4: Módulos de generación de pseudocódigo y código fuente</p>	<ul style="list-style-type: none"> <li>1. Funcionamiento de módulo de generación de pseudocódigo.</li> <li>2. Desarrollo de ejemplo sobre generación de pseudocódigo.</li> <li>3. Funcionamiento de módulo de generación de código fuente.</li> <li>4. Desarrollo de ejemplo sobre generación de código fuente.</li> </ul>	<p>Este curso se dividirá en una parte teórica y una parte práctica. La temática del curso debe cubrir la generación de pseudocódigo y código fuente, las condiciones previas para su correcto funcionamiento y la aplicación de las opciones.</p> <p>Además se debe realizar una evaluación durante la parte práctica para conocer el nivel de conocimiento de los participantes acerca de la herramienta.</p>
<p>Curso 5: Módulos de</p>	<ul style="list-style-type: none"> <li>1. Funcionamiento de módulo de generación de pseudocódigo a</li> </ul>	<p>Este curso se dividirá en una parte teórica y una parte práctica. Se deben</p>

Nombre del Curso	Temas	Descripción
generación de pseudocódigo a flujograma y viceversa y generación de versiones.	<p>flujograma.</p> <p>2. Desarrollo de ejemplo sobre generación de pseudocódigo a flujograma.</p> <p>3. Funcionamiento de módulo de generación de versiones.</p> <p>4. Desarrollo de ejemplo sobre generación de versiones.</p>	explicar los módulos de conversión de pseudocódigo a flujograma y el módulo de generación de versiones, para casa uno se deben detallar los pasos que deben seguir para su adecuado funcionamiento.

*Tabla 4.1 – Descripción del Contenido de la Capacitación*

#### 4.1.1. Duración de la Capacitación

No hay una duración ideal para las sesiones de capacitación. La duración óptima será determinada básicamente por:

- La cantidad de temas a cubrir.
- La complejidad de los temas que van a tratarse.
- Las aptitudes básicas y la experiencia previa de los participantes.

Sin embargo, las sesiones de más de cinco o seis horas en un solo día disminuirán los niveles de energía y atención, de los participantes y el instructor de capacitaciones, y posiblemente conllevarán a reducir la efectividad.

La Tabla 4.2 muestra el estimado de la duración de cada curso de capacitación.

Nº	Nombre	Duración
1	Proceso de instalación	4
2	Generalidades y módulos principales	4
3	Módulos principales	4
4	Módulos de generación de pseudocódigo y código fuente	4
5	Módulos de generación de pseudocódigo a flujograma y viceversa y generación de versiones.	4
<b>Duración Total de la Capacitación</b>		<b>20 horas</b>

*Tabla 4.2 - Duración de la Capacitación*

La duración de los cursos, de acuerdo a la tabla anterior es de **20 horas** hábiles, equivalente a 2.5 días hábiles. Además de impartir los cursos establecidos, es necesario que se efectúen

evaluaciones del aprendizaje de los participantes, las cuales se realizarán al finalizar cada curso y estarán inmersas dentro del tiempo definido, con una duración aproximada de 10 a 15 minutos.

#### 4.1.2. Usuarios a Capacitar

De acuerdo a la naturaleza de la organización, el personal a capacitar que representa a los usuarios finales de la herramienta de software EISI - FlowChart, así como a las partes interesadas, son los que se listan a continuación:

- Jefe del Departamento de Programación y Manejo de Datos.
- Coordinador de Cátedra de Introducción a la Informática.
- Docentes de Introducción a la Informática y Programación I.
- Personal de Soporte y Mantenimiento de Computadoras.

Esta lista representa a todas las personas que tendrán uso primario o secundario de la herramienta de software. Esto incluye a todos los participantes, desde el personal de instalación de la aplicación en los centros de cómputo hasta aquellos que usarán la aplicación para impartir clases teóricas y prácticas.

La Tabla 4.3 muestra la cantidad de personas a capacitar por cada tipo de usuario.

Tipo de Usuario	Cantidad
Jefe del Departamento	1
Coordinador de cátedra	1
Docentes Universitarios	8
Personal de Soporte y Mantenimiento de Computadoras.	3
<b>TOTAL</b>	<b>13</b>

*Tabla 4.3 - Cantidad de Usuarios a Capacitar*

##### 4.1.2.1 Grupos de Capacitación

Para formar los grupos de capacitación se ha elaborado la Tabla 4.4, en donde se ha asignado una abreviatura para identificar de manera más fácil a cada tipo de usuario.

Tipo de Usuario	Código	Cantidad
Administradores de la Herramienta	ADS	2

Docentes	DOC	8
Personal de Soporte y Mantenimiento de Computadoras.	MAN	3
<b>TOTAL</b>		<b>13</b>

*Tabla 4.4 – Tipos y Cantidad de Usuarios*

#### 4.1.2.2 Horarios

Para organizar los grupos de capacitación y determinar el horario de actividades correspondiente, se ha tomado en cuenta el tipo de jornada laboral que se desarrolla dentro de. La planeación se diseña en secuencia de actividades, permitiendo poco a poco que el usuario se vaya vinculando fácilmente en el proceso de aprendizaje de la herramienta, lo cual le permite ir respondiendo de manera positiva y acertada a las actividades programadas. Con lo cual se debe de tomar en cuenta las metas del grupo y sus respectivas necesidades frente a la herramienta de software EISI - FlowChart.

A partir de lo anterior se ha considerado lo siguiente:

1. Los cursos se impartirán a través del horario:

- *Turno 1:* de 8:00 a.m. a 12:00 p.m.

Estos turnos se han definido para que los usuarios participantes asistan solamente media jornada a las capacitaciones, de tal modo que sus labores no se vean afectadas de manera significativa.

2. Las clases serán una combinación teórica y práctica en el tiempo establecido.

La Tabla 4.5 muestra el resumen de la distribución de los cursos por turno.

Curso	Turno	Usuario
Curso 1: Proceso de instalación	Turno 1	ADS MAN
Curso 2: Generalidades y Módulos principales	Turno 1	ADS, DOC
Curso 3: Módulos principales	Turno 1	ADS, DOC
Curso 4: Módulos de generación de pseudocódigo y código fuente	Turno 1	ADS
Curso 5: Módulos de generación de pseudocódigo a flujograma y viceversa y	Turno 1	ADS

generación de versiones.		
--------------------------	--	--

*Tabla 4.5 Grupos de Capacitación*

### **4.1.3. Entorno para la Capacitación<sup>37</sup>**

El entorno físico en el cual se lleva a cabo la capacitación puede tener una importante repercusión en la eficacia de la misma. La adquisición de habilidades puede verse adversamente afectada por ambientes en los cuales los que reciben la capacitación o los capacitadores mismos se sienten incómodos, o las instalaciones son inadecuadas para los requisitos de la sesión de capacitación.

Se debe definir un local para las capacitaciones, el cual puede ser una sala de reuniones, en donde se disponga del espacio suficiente y se cuente con los equipos de cómputo adecuados.

#### **4.1.3.1 Estándares del área para la capacitación**

Hay algunos estándares básicos para las áreas o edificios que se emplearán para la capacitación, que son muy recomendables:

- ✓ Acceso al transporte.
- ✓ Facilidad de acceso - debe evitarse en lo posible el uso de edificios a los que se llega por escaleras muy largas o puertas estrechas que pueden impedir el uso de equipo.
- ✓ Suficiente iluminación y circulación del aire. En la medida de lo posible, hay que evitar los entornos en los cuales no se pueden modificar las temperaturas extremas.
- ✓ Suficiente espacio para el tamaño del grupo que se capacita - los espacios menores de dos por tres metros cuadrados por participantes resultarán incómodos, excepto si las sesiones son muy cortas.
- ✓ Suficientes asientos cómodos para todos los participantes.
- ✓ Suficientes mesas y escritorios para todos los participantes para tomar notas o para las tareas de revisión.
- ✓ Ausencia de ruido en áreas adyacentes durante las sesiones de capacitación.
- ✓ Acceso a los servicios sanitarios y al agua.

Habrán requisitos adicionales de acuerdo con el contenido de la sesión de capacitación. Estos podrían incluir:

<sup>37</sup> “Entorno para la Capacitación” – Fuente: <http://aceproject.org/main/espanol/po/poe05b04.htm>.

- ✓ Un espacio adecuado y ambientado para el uso de equipos de cómputo suficientes, para simular la operación de la herramienta de software EISI - FlowChart.
- ✓ Espacio para que el grupo de capacitación se divida en grupos pequeños de discusión o para hacer ejercicios.

Es importante que las áreas para capacitación se revisen para asegurar que cuentan con las instalaciones básicas, y que las instalaciones de agua y servicios sanitarios, así como de luz eléctrica (donde se requieran) funcionan bien.

#### **4.1.3.2 Acondicionamiento del área de capacitación**

El área de capacitación debe acondicionarse con suficiente anticipación al inicio de la sesión. Esto tendrá el doble propósito de emplear con más efectividad el tiempo del grupo y de permitir la identificación (y reemplazo) de los materiales y equipo faltantes. Los requisitos de materiales y de acondicionamiento del área deben definirse detallarse.

#### **4.1.3.3 Fomentar la participación**

También hay que realizar esfuerzos para fomentar la participación en las personas asistentes, el instructor de capacitaciones debe fomentar dicha participación brindando confianza y apertura para que los usuarios expongan sus dudas o consultas respectivas.

#### **4.1.4. Equipo Tecnológico y Material Didáctico**

Considerando que se ha definido 1 turno, y la cantidad de usuarios a capacitar es de 13 personas, más el Instructor de Capacitaciones, se debe equipar el local correspondiente con 14 equipos de cómputo, que cumplan con los requerimientos mínimos para acceder a la herramienta EISI - FlowChart.

Además se debe disponer de un proyector de video (cañón), el cual utilizará el Instructor de Capacitaciones para presentar el material requerido durante la capacitación, como la ilustración de los procedimientos a seguir para el uso correcto de los módulos de la herramienta de software EISI - FlowChart.

En cuanto al material didáctico, se han considerado los siguientes elementos:

1. Personal que recibirá la capacitación
  - ✓ 1 Manual de acuerdo al tipo de usuario y curso a desarrollar.
  - ✓ 1 Bolígrafo.
2. Instructor de Capacitaciones.
  - ✓ 1 Juego de todos los manuales de la herramienta de software EISI - FlowChart.

✓ 1 Bolígrafo.

#### 4.1.5. Métodos para Impartir la Capacitación

Se realizó una investigación sobre los diferentes métodos para impartir capacitaciones<sup>38</sup>, las cuales se desarrollan actualmente en el país en empresas especializadas en proveer capacitaciones de software, dentro de las más conocidas se encuentran: Executrain y New Horizons.<sup>39</sup>

Los métodos más comunes de capacitación, se describen brevemente en la Tabla 4.6.

Metodología	Descripción
Cursos teóricos	Ayuda a crear los conocimientos teóricos necesarios para el buen uso de herramientas de software.
Cursos prácticos	Sirve para demostrar de forma práctica con ejemplos, la utilización de herramientas de software.
Cursos teórico-prácticos	Ayuda a crear los conocimientos teóricos necesarios para el uso de herramientas de software así como a reforzarlos con ejemplos prácticos de la utilización de las mismas.
Talleres prácticos	Utilizados para construir el conocimiento a través de la interacción entre los participantes y las actividades prácticas mediante ejercicios.
Cursos de autoinducción	Desarrollar la capacidad del aprendizaje de la herramienta de forma autónoma con guías de referencia y ejemplos escritos o cursos en línea autoguiados.
Tutorías	Entrenamiento personalizado teórico y práctico de la herramienta de software mediante un instructor exclusivo por persona.
Conferencia	Permite llegar a una gran cantidad de personas y transmitir un amplio contenido de información o enseñanza. Se puede emplear como explicación preliminar antes de demostraciones prácticas.

Tabla 4.6 - Descripción de metodologías existentes para capacitaciones

<sup>38</sup> Fuente: Tema “Métodos de Capacitación”

[http://www.elprisma.com/apuntes/administracion\\_de\\_empresas/capacitacionrecursoshumanos/default2.asp](http://www.elprisma.com/apuntes/administracion_de_empresas/capacitacionrecursoshumanos/default2.asp)

<sup>39</sup> Fuente: <http://www.newhorizons.com.sv/>; <http://www.executrain.com.sv/quienessomos.htm>

Considerando las condiciones en las que se encuentra el personal que recibirá las capacitaciones, así como los recursos disponibles, los atributos evaluados para las diferentes metodologías son los siguientes:

- ✓ Suficientes recursos audiovisuales.
- ✓ Menor costo.
- ✓ Mayor impacto en la construcción del conocimiento.
- ✓ Grado intermedio de habilidad previa en la utilización de herramientas de software.
- ✓ Local adecuado.

La tabla 4.7 muestra la comparación realizada y los resultados correspondientes.

Metodología	Suficientes recursos audiovisuales	Menores costos	Mayor impacto en construcción del conocimiento	Grado intermedio de habilidad previa	Local adecuado	Resultado
Cursos teóricos	✗	✓	✗	✗	✓	✓✓
Cursos prácticos	✓	✗	✓	✓	✓	✓✓✓✓
Cursos teórico-prácticos	✓	✓	✓	✓	✓	✓✓✓✓
Talleres prácticos	✗	✗	✓	✓	✗	✓✓
Cursos de autoinducción	✗	✗	✗	✗	✗	✗
Tutorías	✗	✗	✓	✓	✗	✓✓
Conferencia	✓	✓	✗	✗	✓	✓✓✓

Tabla 4.7 - Tabla comparativa de metodologías de Capacitación

De esta forma se puede apreciar que la metodología que cumple todos los atributos requeridos para el plan de capacitación son los **Cursos teórico-prácticos**.

Durante las diferentes sesiones de clase de cada curso, la metodología a utilizar es una combinación teórica y práctica. Es decir, que permita al usuario la interacción directa con el, donde se realice previamente una breve introducción a los temas que se abordarán, se expongan los pasos necesarios para realizar determinada acción a través de una demostración, y posteriormente, los usuarios realicen la misma operación con otro ejemplo, en su computadora.

Con esta metodología se pretende que los usuarios se familiaricen con el uso de la herramienta de software EISI – FlowChart y que solventen sus dudas de inmediato, obteniendo confianza en la operación de la aplicación.

#### **4.1.6. Evaluación del Conocimiento<sup>40</sup>**

Es importante asegurar que el personal capacitado tenga suficientes conocimientos sobre el desarrollo de sus tareas, relacionadas con el uso de la herramienta de software, a fin de poderlas llevar a cabo de manera competente. Los métodos de evaluación del conocimiento adquirido son una parte integral de los programas de capacitación.

Cuando se han programado múltiples sesiones de capacitación, la retroalimentación sobre los cursos que se vayan desarrollando puede ayudar a que la asimilación se realice de mejor manera, ya que se procede a recordarle al usuario los contenidos vistos en sesiones anteriores.

Por medio de la evaluación se podrá monitorear los temas asimilados por los usuarios. Por lo tanto, el Instructor de Capacitaciones deberá realizar una prueba al finalizar cada curso de capacitación y además tomar en cuenta la asistencia de las personas al curso.

En cada prueba se expondrán casos prácticos que el usuario desarrollará utilizando la herramienta de software EISI - FlowChart, cuya duración variará de acuerdo al contenido del curso, para lo cual se podrán realizar las consultas requeridas al Instructor de Capacitaciones.

La nota final se obtendrá de la siguiente manera:

- (Promedio de la nota obtenida en asistencia) + (Promedio del conjunto de pruebas realizadas).

Donde la nota mínima para aprobar cada curso es de 7.0.

La ponderación correspondiente se ha asignado de la siguiente manera:

- En cada clase se verificará la asistencia y la participación en los cursos, lo cual será equivalente a un valor total del 40% del final del curso.
- El 60% corresponde a las pruebas de evaluación de conocimientos adquiridos.

---

<sup>40</sup> “Evaluación del Conocimiento” – Fuente: <http://aceproject.org/main/espanol/po/poe05b06.htm>.

La asignación de ponderaciones se realizó tomando en cuenta que, a pesar de que la asistencia a los cursos correspondientes, es importante, el usuario no podrá aprobar la evaluación solo con haber asistido a los cursos, ya que es importante determinar si ha adquirido los conocimientos respectivos, y esto se verifica a través de una prueba o examen.

#### **4.1.6.1 Autoevaluación del personal**

Este enfoque se puede combinar con técnicas de autoevaluación. Al concluir los puntos más importantes en el plan de capacitación, se puede pedir a los propios miembros del equipo de trabajo que identifiquen las áreas en las que ellos creen que necesitan más información o entrenamiento para poder llevar a cabo sus tareas de modo eficiente. Estos temas pueden ser tratados mediante una asesoría individual durante los descansos entre las sesiones de capacitación, o bien puede modificarse el programa de las sesiones restantes de capacitación para permitir una revisión adicional y una discusión más a fondo de los temas sobre los que el grupo en general requiera mayor información.

#### **4.1.6.2 Reconocimientos**

Un método eficaz para estimular el interés durante el adiestramiento consiste en la entrega de certificados de reconocimiento a los usuarios que hayan completado exitosamente los programas de capacitación y cuya competencia en sus tareas haya sido evaluada como satisfactoria.

## Conclusiones

A través del desarrollo de este trabajo de graduación, concluimos lo siguiente:

- Cuando se inicia un proyecto de desarrollo de una aplicación, es indispensable realizar las investigaciones respectivas que permitan detectar todos los detalles de la situación actual de la entidad que requiere la aplicación, ya que de esta forma se garantiza la obtención de una herramienta ajustada a las necesidades de dicha entidad.
- El análisis y validación del listado de requerimientos de la herramienta, permitió definir los elementos necesarios para trazar el camino a seguir en el desarrollo de la misma.
- La etapa de análisis de la herramienta ayuda a traducir las necesidades del cliente en un modelo de sistema que utiliza uno o más de los siguientes componentes: software, hardware, personas y documentación.
- Es de gran importancia la colaboración y apoyo de las autoridades correspondientes de la entidad u organización, ya que con ello es posible realizar la validación de los elementos de diseño de la herramienta, lo cual permite la construcción de una herramienta eficiente y funcional.
- La documentación adecuada de la herramienta (manuales y plan de implementación), contribuye a que el funcionamiento y mantenimiento de éste se lleve a cabo de forma satisfactoria.

## **Recomendaciones**

### **A los usuarios de la Herramienta:**

- Apoyarse en los manuales que acompañan a la herramienta EISI – FlowChart para utilizarlo de manera correcta.
- Recurrir al administrador de la aplicación para solventar cualquier duda relacionada con el funcionamiento de la herramienta.

### **Al Administrador del Proyecto**

- Seguir las indicaciones mostradas en el plan de implementación, para asegurar que la puesta en marcha de la herramienta EISI – FlowChart se realice de forma adecuada.

### **A las autoridades**

- Se debe disponer del personal capacitado para que administre la herramienta EISI – FlowChart, de tal manera que se provea el mantenimiento adecuado a la herramienta.

## Bibliografía

### 1.1.Libros.

- ✓ Eric J. Braude (2003), Ingeniería de Software, Una perspectiva orientada a objetos. Primera Edición. México, ALFAOMEGA grupo editor.
- ✓ Ian Sommerville, (2005), Ingeniería del Software, Séptima Edición, Madrid, Pearson Educación.
- ✓ Osvaldo Cairó (2005), Metodología de la Programación, Algoritmos, diagramas de flujo y programas, Tercera Edición, México, ALFAOMEGA grupo editor.

### 1.2.Páginas Web

- ✓ Java, Requerimientos de instalación de Java EE 5.0, Recuperado el 1 de mayo de 2009, de [http://javasun.com/javavee/sdk/javaee5sdu7\\_realnotes.jsp](http://javasun.com/javavee/sdk/javaee5sdu7_realnotes.jsp).
- ✓ NetBeans, Requerimientos de instalación de NetBeans 6.5, recuperado el 1 de mayo de 2009, de <http://www.netbeans.org/community/releases/65/1/rel-notes.html>
- ✓ UES, Biblioteca Virtual, recuperado el 2 de mayo de 2009, de <http://virtual.ues.edu.sv/>
- ✓ El Prisma,  
[http://www.elprisma.com/apuntes/administracion\\_de\\_empresas/capacitacionrecursoshumanos/default2.asp](http://www.elprisma.com/apuntes/administracion_de_empresas/capacitacionrecursoshumanos/default2.asp), recuperado el 3 de Noviembre de 2009.

## Anexos

### Anexo 1: Presupuesto del proyecto

#### Estimación del costo total del proyecto

Para el cálculo del costo total del proyecto se consideraron los siguientes costos:

- ✓ Costos directos
- ✓ Costos indirectos

#### Costos directos

##### Recurso humano

El grupo de trabajo está formado por 4 personas, los cuales se encargaran durante 8 meses del desarrollo del proyecto, también un asesor del grupo el cual estará apoyando en la orientación y revisión de los avances que se vayan presentando a lo largo de la vida del proyecto.

Considerando que el tiempo establecido por la Escuela de Ingeniería de Sistemas Informáticos para el desarrollo del trabajo de graduación es de 8 meses y que el tiempo de trabajo semanal para los integrantes del grupo se recomienda en 4 horas diarias (24 horas a la semana, de lunes a sábado), se puede realizar un cálculo del costo del recurso humano, por medio del sueldo promedio para un desarrollador en Java calculado en base a sueldos de empresas privadas para este tipo de recurso humano.

En la tabla 1.1 se presentan los resultados de una investigación del sueldo mensual en el sector privado para un desarrollador en Java con 1 año de experiencia.

##### Salarios promedios de empresas privadas para programadores en Java

Empresa	Salario (\$ USD)
ASEINFO	700.00
CONSISA	900.00
SERCOMCA	850.00
Global Solutions	700.00
Banco G&T Continental	950.00
<b>Salario promedio</b>	<b>820.00</b>

Tabla 1.1 – Sueldo mensual para un desarrollador en Java

Con el resultado promedio obtenido se puede calcular el sueldo promedio por hora de un programador Java, tomando en cuenta que el tiempo laboral son de 44 horas a la semana en el sector privado, este cálculo se muestra en la tabla 1.2.

##### Estimación de sueldo promedio por hora

Sueldo promedio por hora	
Sueldo promedio	\$ 820.00
Horas trabajadas por mes (44 horas x 4 semanas)	176 Hrs.
<b>Sueldo promedio por hora a tiempo completo</b>	<b>\$ 4.66</b>
<b>Sueldo promedio por hora a medio completo</b>	<b>\$ 3.33</b>

Tabla 1.2 – Sueldo promedio por hora

Luego se calculó la duración total del proyecto, considerando los 8 meses con una jornada de 4 horas diarias, de lunes a sábado, mencionado anteriormente. Dicho cálculo se muestra en la tabla 1.3.

#### Estimación de duración del proyecto

Duración total del proyecto	
Duración en meses	8
Duración en semanas	32
Duración en días	192
Duración en horas	768

Tabla 1.3 – Cálculo de la duración del proyecto

El salario por hora para el asesor del proyecto, se considero una jornada promedio de 2 horas semanales, durante 8 meses. Se considero que el salario actual de un docente es de \$1,100.00 a tiempo completo y de \$550 a medio tiempo, para este caso se tomara en cuenta el salario de un docente a medio tiempo.

#### Estimación de sueldo promedio por hora del docente director del proyecto

Sueldo promedio por hora	
Sueldo promedio mensual	\$ 1,100.00
Horas promedio semanal	40 Hrs.
Horas promedio mensual	160 Hrs.
Sueldo promedio por hora a Tiempo completo	\$ 6.88
Sueldo promedio por hora a Medio tiempo	\$ 3.44

Tabla 1.4 - Cálculo del sueldo promedio por hora del asesor

El cálculo del promedio por hora a medio tiempo, se obtiene de dividir entre dos el salario promedio por hora a tiempo completo.

Por lo tanto el costo por hora del docente director es de \$ 3.44, considerando 64 horas en total a invertir durante todo el proyecto, tal como se muestra en la tabla anterior (Tabla 1.4)

Finalmente se realizo el cálculo total del recurso humano para el desarrollo del proyecto, ver datos de la tabla 1.5.

Recurso humano	Cantidad	Duración del proyecto (Hrs.)	Sueldo promedio por hora (\$USD)	Total (\$ USD)
Integrante del grupo	4	768	2.33	7,157.76
Asesor del grupo	1	64	3.44	220.16
<b>Total</b>				<b>7,377.92</b>

Tabla 1.5 – Cálculo del costo del recurso humano requerido

### Recurso tecnológico

Esta sección se compone principalmente por el hardware destinado para el desarrollo del proyecto, las especificaciones de estos elementos se pueden observar en el capítulo I sección 4, apartado 4.1 Factibilidad Técnica.

El Ministerio de Hacienda de la República de El Salvador, establece el método constante de depreciación para programas informáticos, según la LEY DE IMPUESTO SOBRE LA RENTA<sup>41</sup>, decreto N° 134, La asamblea legislativa de la República de El Salvador en el artículo 30-A. Amortización de programas informáticos o software.

Para el cálculo de la depreciación se utilizará el método de depreciación en línea recta<sup>42</sup>. Este método plantea que la depreciación es una función constante del tiempo y que las causas que la provocan tienen efectos continuos y homogéneos, supone que el activo se desgasta por igual durante cada período contable y se usa con frecuencia por ser fácil de aplicar y calcular. Dicho método toma como base la vida útil del activo.

En base al hardware a utilizar en el desarrollo del proyecto, mencionado en la factibilidad técnica, se estimará la depreciación de cada uno de estos activos, utilizando el método de depreciación antes mencionado, y para esto se utilizara la siguiente fórmula:

$$\text{Depreciación} = \text{Inversión} / \text{Vida útil fiscal}$$

Donde:

*Inversión*: Costo total del equipo.

Vida útil fiscal = Se considera 5 años para equipo nuevo y 3 años para equipo en uso o ya existente.

En la tabla 1.6 se muestra los cálculos realizados para cada recurso tecnológico a utilizar.

Equipo	Precio Unitario (\$ USD)	Vida útil	Depreciación anual	Depreciación Mensual
Equipo 1	1,400.00	3	466.67	38.89
Equipo 2	1,000.00	3	333.33	27.78
Equipo 3	380.00	3	126.67	10.56
Equipo 4	425.00	3	141.67	11.80
Impresor	45.00	3	15.00	1.25
Switch	55.00	3	18.33	1.52
<b>Total</b>			<b>\$ 1,101.67</b>	<b>\$ 91.80</b>

Tabla 1.6 – Cálculo de depreciación del recurso tecnológico

En donde la depreciación anual y la depreciación mensual, se calcularon con las siguientes formulas:

<sup>41</sup> Ministerio de Hacienda, Ley de impuesto sobre la renta, Recuperado el 2 de mayo de 2009, de [http://www.mh.gob.sv/pls/portal/docs/PAGE/MH\\_IMPUESTOS/RENTALEY/LEY\\_DE\\_RENTA\\_REF\\_DIC\\_07.PDF](http://www.mh.gob.sv/pls/portal/docs/PAGE/MH_IMPUESTOS/RENTALEY/LEY_DE_RENTA_REF_DIC_07.PDF)

<sup>42</sup> Fundamentos de Administración Financiera, Autor James C. Van Horne, Jr. Wachowicz, John., Página 20 Publicado en 2002, Pearson Educación Editorial.

Depreciación Anual = Precio Unitario / Vida útil

*Depreciación Mensual = Depreciación Anual / 12 Meses*

El cálculo del costo del equipo para el proyecto se muestra en la tabla 1.7

	Costo del equipo	Depreciación anual	Depreciación mensual	Duración del proyecto (Meses)	Costo del equipo para el proyecto
<b>Totales</b>	\$ 3,305.00	\$ 1,101.67	\$ 91.80	8	<b>\$ 734.40</b>

*Tabla 1.7 – Cálculo del costo total del recurso tecnológico*

Donde el costo del equipo para el proyecto se obtuvo a través de la siguiente fórmula:

*Costo del equipo para el proyecto = Depreciación Mensual \* 8 Meses*

En base al recurso tecnológico disponible para el desarrollo del proyecto y con el tiempo para su realización, se estimo un costo total para el recurso tecnológico de: \$ 734.40.

El total de costos directos para el desarrollo del proyecto se detalla en la tabla 1.8 que se muestra a continuación:

Recurso	Total (\$)
Recurso humano	7,377.92
Recurso tecnológico	734.40
<b>Total</b>	<b>8,112.32</b>

*Tabla 1.8 – Resumen de costos directos*

## Costos indirectos

Detalle de costos de los recursos consumibles y recursos de operación.

### Recursos consumibles

En la tabla 1.9 se detallan los recursos consumibles a utilizar a lo largo del proyecto.

Recurso	Cantidad	Costo Unitario (\$ USD)	Total (\$ USD)
Pizarra acrílica de 1.5x1 Mts.	1	35.00	35.00
Caja de plumones para pizarra	1	5.00	5.00
Cable UTP Categoría 5 (en metros)	10	0.40	4.00
Conectores RJ-45	4	0.30	1.20
Torre de 25 unidades de CD	1	6.00	6.00
Resmas de papel bond tamaño carta	10	4.30	43.00
Cartucho de tinta negra	4	21.00	84.00
Cartucho de tinta color	4	25.00	100.00
Fólder tamaño carta	25	0.12	3.00

Fastener	25	0.05	1.25
Alquiler de cañón para defensas <sup>43</sup> (horas)	6	6.00	36.00
Anillados <sup>44</sup>	9	2.50	22.50
Empastado (para 3 ejemplares)	3	20.00	60.00
<b>Total</b>			<b>\$ 400.95</b>

Tabla 1.9 – Cálculo de costos de consumibles y materiales

### Recursos de operación

Para el desarrollo del proyecto se necesitaran los siguientes recursos de operación:

- ✓ *Servicio de energía eléctrica:* Se ha estimado en base al uso de energía eléctrica para iluminación y para el uso del equipo de cómputo, el cual es un servicio residencial, en donde se tiene una capacidad de suministro contratada de 66KW actualmente. Se han considerado las tarifas publicadas en la página Web de la superintendencia General de Electricidad y Telecomunicaciones (SIGET) para la empresa distribuidora CAESS, ya que es quien suministra el servicio eléctrico. Se tomaron en cuenta 16 horas de trabajo durante cinco días a la semana para estimar el consumo eléctrico, más comisiones y cargos estipulados por la SIGET, además de la contribución mensual para la Alcaldía de Mejicanos. El valor al que asciende este consumo mensual es de **\$8.00**.
- ✓ *Servicio de agua potable:* En base a la tarifa de los recibos de meses anteriores, el consumo mensual asciende a **\$5.00**.
- ✓ *Servicio de Internet:* Se adquirió una línea dedicada DSL de AMNET de 512 kbps, con un costo mensual de **\$28.25**.
- ✓ *Servicio de teléfono celular:* Este gasto se estimó realizando un promedio de costos de acuerdo a las líneas telefónicas de cada miembro del grupo, este detalle se refleja en la tabla 1.10. El valor al que asciende el costo de teléfono celular es de **\$35.00**.
- ✓ *Transporte:* La estimación mensual de este rubro es de \$30.00 dólares mensuales. Las consideraciones tomadas para estos cálculos se detallan en la tabla 1.11 Y 1.12.

### Estimación de gasto en telefonía celular

Detalle costo por minuto según compañía de teléfono				
Cantidad de teléfonos	Compañía	Cobro por minuto misma compañía (ctvs.)	Cobro por minuto diferente compañía (ctvs.)	Total mensual
2	Digicel	\$ 0.15	\$ 0.35	\$10.00
2	Telefónica Movistar <sup>45</sup>	\$ 0.15	\$ 0.15	\$10.00
<b>Total</b>				<b>\$20.00</b>

Tabla 1.10 – Cálculo de gasto telefónico mensual

<sup>43</sup> Se ha considerado 3 defensas con una duración promedio de 2 horas, lo cual da como resultado 6 horas de alquiler para todo el proyecto.

<sup>44</sup> Se ha considerado que se impartirán 3 copias anilladas para cada defensa, una para el director docente, otra para el docente observador y otra para la escuela de sistemas.

<sup>45</sup> Plan con llamadas ilimitadas entre los dos teléfonos de la misma compañía.

### Estimación de gastos de transporte

Para estimar los gastos por transporte se tomará como base la cantidad de reuniones semanales entre los integrantes del grupo de trabajo, las cuales son:

*5 reuniones semanales \* 4 semanas = 20 reuniones mensuales*

*1 reunión semanal con docente director \* 4 semanas = 4 reuniones mensuales*

**Total de reuniones = 24 reuniones mensuales.**

La tarifa de transporte colectivo vigente es de \$0.20 centavos de dólar. Por tanto tenemos:

Cantidad de buses tomados por integrante			
Cantidad de buses semanales <sup>46</sup>	Integrante	Pasaje vigente (ctvs.)	Total mensual
1	1	\$ 0.20	\$0.80
2	2	\$ 0.20	\$1.60
<b>Total</b>			<b>\$2.40</b>

*Tabla 1.11 - Cálculo de gasto en transporte colectivo*

El gasto en combustible de vehículo semanal es de:

Cantidad de galones semanales	Precio por galón	Total mensual
3	\$ 2.30	\$27.60

*Tabla 1.12 – Cálculo de gasto en transporte colectivo*

El gasto total mensual en transporte asciende a:

Transporte colectivo + Transporte en vehículo

**\$ 2.40 + \$27.60 = \$30.00**

En la tabla 1.13 se hace un resumen de los gastos estimados para el desarrollo del proyecto.

Rubro	Duración del proyecto (meses)	Monto (Dólares \$)	Total por rubro (\$)
Servicio energía eléctrica	8	8.00	64.00
Servicio de agua potable	8	5.00	40.00
Servicio de Internet	8	28.25	226.00
Servicio de telefonía celular	8	20.00	160.00
Trasporte	8	15.00	120.00
<b>Total</b>			<b>610.00</b>

*Tabla 1.13 – Cálculo de costos de recursos operativos y alimentación*

<sup>46</sup> Para reuniones con el docente director del proyecto

## Resumen de costos

Al realizar un consolidado de los gastos involucrados en el desarrollo del proyecto obtuvimos el valor estimado del costo total del proyecto, el cual se muestra en la tabla 1.14.

Según el Ministerio de Economía de El Salvador, el porcentaje de imprevistos que se asigna en la elaboración de presupuestos para el desarrollo de proyectos es del 5% del total del costo.

<b>Categoría</b>	<b>Subtotal</b>	<b>Total</b>
<b>Costos Directos</b>		<b>8,112.32</b>
Recursos humanos	7,377.92	
Recursos tecnológicos	734.40	
<b>Costos indirectos</b>		<b>1,010.95</b>
Consumibles y materiales	400.95	
Recursos operativos y alimentación	610.00	
	<b>TOTAL</b>	<b>9,123.27</b>

*Tabla 1.14 – Resumen de costos involucrados en el proyecto*

## Anexo 2: Encuestas realizadas a docentes

### Encuesta a docentes universitarios 1



UNIVERSIDAD DE EL SALVADOR  
FACULTAD DE INGENIERÍA Y ARQUITECTURA  
ESCUELA DE INGENIERÍA DE SISTEMAS INFORMÁTICOS  
COORDINACIÓN DE TRABAJOS DE GRADUACIÓN

#### ENCUESTA A DOCENTES (IAI 115)

**Objetivo:** Conocer el punto de vista de los docentes respecto al desarrollo actual de las asignaturas de primer año que utilizan el proceso solucionador de problemas como parte del contenido de las mismas y la apertura de utilización de una herramienta que brinde soporte al proceso solucionador de problemas.

Nombre de la asignatura que imparte actualmente: \_\_\_\_\_

1. ¿Se utiliza el tópico del proceso solucionador de problemas actualmente en la asignatura?

Sí                       No                      ¿Para qué se utiliza?

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

2. ¿Utiliza alguna herramienta para desarrollar el contenido de la asignatura, específicamente para el proceso solucionador de problemas?

Sí                       No                      ¿Cuáles son y por qué las utiliza?

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

3. ¿Considera que es difícil para los estudiantes asimilar el proceso solucionador de problemas?

Sí                       No                      ¿Por qué?

4. Le parece que sería conveniente contar con una herramienta que le brinde soporte para el desarrollo del proceso solucionador de problemas que cuente con las siguientes características: Permitir elaborar la formulación del problema (uso de caja negra), Realizar el análisis de variables (entrada, salida y proceso), Elaboración de diagramas de flujo, Soporte a modularidad, Pruebas en ejecución del diagrama de flujo, Generación de pseudocódigo (a partir del diagrama de flujo elaborado), Generación de código fuente

Sí                       No                      ¿Por qué?

---

---

---

5. ¿Considera que se debería de agregar alguna otra funcionalidad a la herramienta?

Sí                       No                      ¿Cuáles?

---

---

---

6. ¿Considera que contar con una herramienta de apoyo práctico para el proceso solucionador de problemas influiría positivamente en los resultados que obtienen los estudiantes?

Sí                       No                      ¿Por qué?

---

---

---

7. Según su criterio, ¿Cuáles son los temas para los que se observa mayor deserción de los estudiantes?

---

---

---

8. ¿Cuáles son los principales inconvenientes que se tienen para el desarrollo de la asignatura, específicamente para el desarrollo del proceso solucionador de problemas?

---

---

---

---

9. ¿Cuáles son las principales ventajas que obtendría de utilizar una herramienta que brinde soporte al proceso solucionador de problemas?

---

---

## Encuesta para docentes universitarios 2



**UNIVERSIDAD DE EL SALVADOR**  
**FACULTAD DE INGENIERÍA Y ARQUITECTURA**  
**ESCUELA DE INGENIERÍA DE SISTEMAS INFORMÁTICOS**  
**COORDINACIÓN DE TRABAJOS DE GRADUACIÓN**

**OBJETIVO:** Conocer el punto de vista de los docentes para poder definir los requerimientos de la *Herramienta para el desarrollo asistido de flujogramas con soporte al proceso solucionador de problemas*.

### PARTE I - Requerimientos de distribución.

**INDICACIONES:** Seleccione las opciones que considere conveniente que deben ser incluidas en los diferentes menús de la herramienta. En caso de seleccionar otros, favor indicar cuáles opciones considera necesarias de agregar.

Menú Archivo		
	Nuevo Proyecto	
	Abrir Proyecto	
	Cerrar Proyecto	
	Guardar Proyecto	
	Guardar proyecto como	
	Imprimir	
	Configurar página	
	Salir	
	Otros.	
Menú Editar		
	Deshacer	
	Rehacer	
	Copiar	
	Pegar	
	Cortar	
	Seleccionar todo.	
	Otros.	
Menú Ver		
	Controles	
	Barra de estado	
	Barra de herramientas	
	Explorador de proyectos	
	Zoom	
	Líneas / guías	

Menú Proyecto		
	Agregar nuevo elemento	
	Agregar elemento existente	
	Configuración del proyecto	
	Otros.	
Menú Compilar		
	Validar	
	Ejecutar	
	Validar y ejecutar	
	Otros.	
Menú Configuración		
	Generación de Versiones	
	Apariencia	
	Otros	
Otros menús		

## PARTE II - Requerimientos No Funcionales (Interfaz gráfica)

**INDICACIONES:** A continuación se muestran una serie de preguntas, seleccione en cada una de ellas la alternativa que usted estime conveniente.

- ¿Cómo considera adecuado que se deben mostrar cada una de las etapas del proceso solucionador de problemas?
  - Por medio de pestañas
  - Por medio de ventanas.
  - Por medio de botones atrás – adelante.
  - Otros. \_\_\_\_\_
  
- ¿Cómo considera necesario que se visualice el entorno en el desarrollo del flujograma?
  - A través de múltiples páginas
  - Por medio de páginas secuenciales.
  - Estilo diapositivas.
  - Otros. \_\_\_\_\_
  
- En la etapa de desarrollo del flujograma, ¿cuál de las siguientes opciones considera más adecuada se deba mostrar en el entorno de trabajo?
  - Líneas guías.
  - Puntos que permitan ubicar a los usuarios.
  - El entorno de trabajo debe ser blanco.
  - Otros. \_\_\_\_\_
  
- ¿Cómo considera que deben mostrarse las opciones de los diferentes menús?
  - Por medio de iconos en la barra de herramienta estándar.
  - Únicamente en los diferentes menús.
  - Otras \_\_\_\_\_
  
- ¿Cómo considera adecuado que debe ser el color de la interfaz gráfica (ventanas) en la herramienta?
  - Transparente (Adaptable al tema del sistema operativo)
  - Color fijo predeterminado
  - Configurable por el usuario
  - Otros. \_\_\_\_\_
  
- ¿Cómo considera adecuado que debe ser el color de los controles del flujograma en la herramienta?
  - Color fijo predeterminado.
  - Diferentes colores.
  - Configurable por el usuario.
  - Otros. \_\_\_\_\_
  
- ¿Considera necesario que la herramienta permita copiar, pegar y cortar elementos en el entorno de trabajo?

- Si
  - No ¿Por qué? \_\_\_\_\_
  
- ¿Cómo considera adecuado que se almacene la información de las etapas del proceso solucionador de problemas?
  - En un solo archivo donde se guarde toda la información.
  - En archivos independientes según la etapa del proceso solucionador de problemas.
  - En archivos independientes con nemónicos que hagan referencia a cada etapa del proceso solucionador de problemas.
  - Otros. \_\_\_\_\_
  
- ¿Considera que la ventana de ejecución/depuración de flujogramas se debe mostrar?
  - Siempre, durante cualquier etapa del proceso solucionador de problemas.
  - Cuando se esté trabajando en la etapa de desarrollo del flujograma.
  - Otras. \_\_\_\_\_
  
- ¿Considera necesaria que la herramienta permita la conexión a periféricos?
  - Si
  - No ¿Por qué? \_\_\_\_\_
  
- ¿En qué formato considera adecuado que se almacene la información del/los archivo(s) del proyecto?
  - XML.
  - Archivo Binario.
  - Archivo de texto plano
  - Otros \_\_\_\_\_
  
- ¿En qué posición considera adecuado que estén ubicadas las barras de herramientas?
  - Parte superior de la pantalla
  - Parte inferior de la pantalla.
  - Lado izquierdo de la pantalla.
  - Lado derecho de la pantalla.
  - Otros. \_\_\_\_\_
  
- Al cargar la herramienta, ¿cuál de las siguientes opciones considera más adecuada que se muestre?
  - Un asistente que le permita seleccionar cualquiera de los dos tipos de proyectos.
  - La pantalla inactiva en la cual debe iniciar un nuevo proyecto por medio del menú.
  - Un proyecto siguiendo el proceso solucionador de problemas.
  - Otros. \_\_\_\_\_
  
- ¿Cómo considera adecuado que sea la administración de los controles en la herramienta para el desarrollo del flujograma?
  - Arrastrar cada uno de los controles hasta una posición en específico.

- Al dar clic sobre el control se debe ubicar en un punto en específico en el área de trabajo.
  - Dar clic sobre el control y luego dar clic sobre el área de trabajo que desee que aparezca.
  - Otros. \_\_\_\_\_
  
- Cuando el usuario este ejecutando el flujograma, ¿cómo considera que deben ser indicados los errores encontrados en el flujograma?
  - Resaltar los puntos donde están los errores dentro del flujograma
  - Mostrar una ventana que indique los errores encontrados
  - No debe indicarle los errores.
  - Otros \_\_\_\_\_
  
- Al momento de crear un nuevo proyecto teniendo un proyecto activo, ¿cuál de las siguientes opciones considera más adecuada?
  - Que se cierre el proyecto activo, consultando si desea guardar o no los cambios en el mismo.
  - Crea el nuevo proyecto en una nueva ventana sin afectar el proyecto activo.
  - Trabajar con un solo proyecto activo al mismo tiempo.
  - Otros. \_\_\_\_\_

### Requerimientos Funcionales

- A continuación se muestran diferentes opciones para el manejo de proyectos en la herramienta, según su criterio ¿cuál sería la más conveniente?
  - Que la herramienta únicamente permita crear proyectos siguiendo el proceso solucionador de problemas.
  - Que la herramienta permita únicamente crear proyectos libres, es decir, sin que se siga el proceso solucionador de problemas.
  - Ambos tipos de proyectos
  - Otros. \_\_\_\_\_
  
- Considera adecuado que la ventana de controles:
  - Permanezca estática durante la etapa del desarrollo del flujograma.
  - Se permita la ocultación/visualización de la misma
  - Otros. \_\_\_\_\_
  
- ¿Cuáles de las siguientes opciones considera más adecuada para la configuración del entorno de trabajo en la herramienta?
  - Debe permitir guardar las configuraciones que el usuario establezca.
  - Debe ser estándar, independientemente que el usuario personalice o no la herramienta.
  - Otros \_\_\_\_\_
  
- Considera necesario que la herramienta permita la ampliación/reducción del entorno de trabajo.
  - Si
  - No ¿Por qué? \_\_\_\_\_

- ¿Cuál de las siguientes opciones considera más adecuada para las barras de herramientas?
  - Deben permanecer estáticas en el área de trabajo.
  - Debe permitirse que se oculten para poder contar con más espacio de trabajo.
  - Otros \_\_\_\_\_

### PARTE III – ETAPAS DEL PROCESO SOLUCIONADOR DE PROBLEMAS.

**INDICACIONES:** A continuación se detallan una serie de alternativas para cada una de las etapas del proceso solucionador del problemas, cada uno de los elementos forma parte de los aspectos esenciales de la aplicación, las opciones que están en negrita es la opción sugerida por el equipo de trabajo, no obstante puede seleccionar la que estime conveniente en cada caso.

Así mismo cada una de las opciones cuenta con la opción “otras”, la cual nos servirá para conocer las opiniones o ideas que permitan construir un sistema de calidad.

#### Planteamiento del problema

- En la etapa del planteamiento del problema se debe contar con un diagrama de Caja Negra que por definición cuenta con Entradas, Salidas y Procesos. ¿Cuál es la forma más conveniente de mostrarlo?
  - **Presentar el diagrama de Caja Negra completo al cargar este paso, luego se mostrará el esquema de la caja negra, en el cual el usuario podrá ingresar un concepto para las salidas y entradas del sistema.**
  - Presentar cada parte del diagrama a medida se vayan completando, es decir, que arrastre rectángulos (para la entrada, proceso y salida) y flechas (para unir los rectángulos).
  - Presentar cada parte a medida el usuario lo solicite, es decir, que aparezca un asistente que le permita al usuario conocer que es lo que se quiere (salidas) y que se necesita para lograrlo (Entradas).
  - Otra.
- Cuando se hayan definido las entradas y salidas en el planteamiento del problema, ¿considera que el sistema debe validar la congruencia de la información ingresada?
  - **No es necesario, porque un texto puede estar escrito de manera diferente y significar lo mismo, por lo tanto se requeriría de un modulo inteligente que valide esto.**
  - Es importante ya que el alumno debe de darle sentido a la entrada y salida de su problema y estos dos conceptos deben de coincidir.
  - No es indispensable validar la congruencia.
  - Otros.
- Cuando un usuario define entradas y/o salidas en el planteamiento del problema, ¿cuál considera que deba ser la mejor relación con la declaración de variables?
  - **La definición de entradas y salidas en el planteamiento del problema solo son descriptivos por lo tanto no debería tener relación con la declaración de variables.**
  - Cada variable de entrada y salida que se escriba en el planteamiento del problema deberá ser identificada en el análisis de variables.
  - Otra.

Análisis del problema.

- ¿Considera adecuado definir un número máximo de variables de entrada, salida o proceso?
  - Si debe existir un número máximo de variables a definir para cada tipo. ¿Cuántas?  
\_\_\_\_\_
  - No, la cantidad de variables a definir dependen del análisis realizado por el estudiante.
  
- Para la definición de restricciones, ¿cuál de las siguientes opciones cree que será la mejor?
  - **Definir las restricciones mediante un asistente que facilite la creación de las mismas.**
  - Escribir las restricciones libremente, y al terminar la restricción validar si la expresión es correcta.
  - Escribir las restricciones libremente, y no validar si la expresión es correcta.
  - Otra.
  
- ¿Cuántas expresiones considera adecuadas manejar en una sola restricción?
  - No debe existir limite de expresiones
  - El límite de expresiones debe ser \_\_\_\_\_
  - Otra

**INDICACIONES:** A continuación se listan una serie de opciones, seleccione la / las que usted estime conveniente según corresponda.

- ¿Cuál de los siguientes operadores de desigualdad considera necesarios para la aplicación?  
**Nota:** en caso de que el operador no le parezca adecuado, favor de colocar al lado del operador el símbolo que usted considera más apropiado para este símbolo:

	Operador	Significado	Operador sugerido
<input type="checkbox"/>	<	Menor que	
<input type="checkbox"/>	>	Mayor que	
<input type="checkbox"/>	<=	Menor o igual que	
<input type="checkbox"/>	>=	Mayor o igual que	
<input type="checkbox"/>	==	Igual a	
<input type="checkbox"/>	!=	Diferente a	

Otros Operadores de desigualdad

- ¿Cuál de los siguientes operadores lógicos considera necesarios para la aplicación?

**Nota:** en caso de que el operador no le parezca adecuado, favor de colocar al lado del operador el símbolo que usted considera más apropiado para este símbolo:

	Operador	Significado	Operador Sugerido
<input type="checkbox"/>	!	No	
<input type="checkbox"/>	&&	Y (And)	
<input type="checkbox"/>		O (Or)	

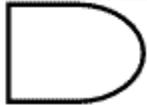
Otros Operadores Lógicos:

- ¿Cuál de las siguientes funciones matemáticas considera que se tomen en cuenta para la aplicación?

	Función	Significado	Nombre que sugiere que se le de en la herramienta para identificarla.
<input type="checkbox"/>	<b><i>Ceil(x)</i></b>	devuelve el menor entero que es mayor o igual al argumento	
<input type="checkbox"/>	<b><i>Cos(x)</i></b>	Coseno de x	
<input type="checkbox"/>	<b><i>Exp(x)</i></b>	el resultado de elevar el número e a la potencia x	
<input type="checkbox"/>	<b><i>Abs(x)</i></b>	valor absoluto de x	
<input type="checkbox"/>	<b><i>Floor(x)</i></b>	devuelve el mayor entero que es menor o igual al argumento	
<input type="checkbox"/>	<b><i>Log(x)</i></b>	Logaritmo natural (en base e) de x	
<input type="checkbox"/>	<b><i>Pow(x,y)</i></b>	Eleva x a la potencia y.	
<input type="checkbox"/>	<b><i>Sin(x)</i></b>	Seno de x	
<input type="checkbox"/>	<b><i>Sqrt(x)</i></b>	Raíz cuadrada de x	
<input type="checkbox"/>	<b><i>Tan(x)</i></b>	Tangente de x	

Otras Funciones Matemáticas:

### Anexo 3: Figuras estandarizadas de la ANSI

	El símbolo representa la terminación de la puesta en marcha o punto final del sistema.
	Una caja indica alguna operación en particular.
	Este representa una salida en pantalla.
	Un diamante representa una decisión o un punto de ramificación. Líneas que salen de los diamantes indican diferentes situaciones posibles, dando lugar a diferentes sub-procesos.
	Se trata de material o información que entra o sale del sistema. Una entrada puede ser una orden de un cliente. Una salida puede ser un producto que se entregara.
	Este símbolo se incluirá dentro de la hoja. Esto indica que el flujo continúa en el correspondiente que contiene la misma hoja, en algún otro lugar de la misma página.
	Como en el caso anterior, excepto que el flujo continúa en el símbolo correspondiente en una página diferente.
	Identifica un retraso o un cuello de botella.
	Líneas representan la secuencia y la dirección de un proceso.

*Tabla 3.1 - Figuras estandarizadas por la ANSI*

Para más información de este estándar, se puede consultar las siguientes normativas para ANSI e ISO:

- ✓ American National Standar Institute ANSI X3.6-1970, el organigrama y sus signos de uso en tratamiento de la información.
- ✓ Organización Internacional de Normalización (ISO), ISO 5807 Procesamiento de la información – Documentación de los símbolos y convenciones, programas y sistema de flujo.

## Anexo 4: Métodos de Conversión de Sistemas<sup>47</sup>

La conversión de un sistema es el proceso de cambio del sistema anterior al nuevo sistema. Existen cuatro métodos de conversión de sistemas.

- Paralelo
- Directo
- Piloto
- Por Etapas

A continuación se presenta una breve descripción de cada uno.

### Método Paralelo.

En este caso, una vez que el nuevo sistema está pronto y operativo el mismo se instala y entra en funcionamiento, sin interrumpir el uso del sistema anterior. Ambos siguen funcionando simultáneamente, hasta que se entiende oportuno dejar de utilizar el sistema anterior, dado que el nuevo funciona adecuadamente.

Es recomendable utilizar este método cuando existen algunas de las siguientes situaciones:

- El nuevo sistema es de gran dimensión (tanto por el tamaño en sí mismo del sistema - número de programas que lo integran o porque afecta a un importante número de unidades de la empresa – clientes, proveedores, sucursales, agencias, servicios, productos, etcétera). Un ejemplo podría ser la incorporación de terminales de autoservicio bancario en las sucursales de un Banco en todo el país.
- El sistema impacta en forma importante en el negocio. Por ejemplo la automatización de reservas y ventas de pasajes en una compañía aérea. O un sistema de facturación en tiempo real, o un nuevo sistema de procesamiento de tarjetas de crédito.
- De alta sensibilidad (para los trabajadores, clientes, proveedores) Por ejemplo cuando se automatiza un sistema de liquidación de haberes al personal en una empresa industrial, donde la forma de liquidación es compleja por las tareas y turnos existentes. O cuando a los clientes de una tarjeta de crédito se les proporciona un nuevo estado de cuenta con mejoras respecto al anterior.

**Ventajas:** este método tiene como mayor ventaja que es más seguro. El sistema anterior se mantiene en funcionamiento hasta tanto se haya validado el cien por ciento del nuevo.

**Desventajas:** es más costoso, dado que duplica una serie de costos operativos; la implementación puede ser más lenta, dado que al saber que aún se dispone del sistema anterior, puede existir una tendencia a demorar las correcciones del nuevo sistema; los usuarios son reacios al cambio y en la medida que el nuevo sistema presente dificultades de implementación, se intentará regresar al sistema anterior que aún está en funcionamiento.

---

<sup>47</sup> Información extraída del tema "Implementación de Sistemas", en el sitio Web:  
<http://www.ccee.edu.uy/ensenian/catsistc/docs/IMPLEMENTACION.PDF>

### **Método Directo**

Este método consiste en implementar el nuevo sistema dejando de lado el sistema anterior, a partir del momento de su instalación. La operativa se transfiere en forma abrupta, de un día para el otro.

Generalmente se trata de disponer de fines de semana largos o feriados intermedios para disponer de tiempo de reacción ante probables dificultades. Este método es recomendable cuando no existen las situaciones señaladas en el caso anterior, cuando se desea minimizar costos de implementación o cuando se pretende dejar un sistema anterior lo más rápidamente posible.

**Ventajas:** es menos costoso que el método paralelo, se obtienen los beneficios del sistema más rápidamente.

**Desventajas:** es más riesgoso, no se dispone del sistema anterior en caso de falla del nuevo sistema; se requiere una planificación más cuidadosa, poner a prueba en forma exigente el nuevo sistema y establecer el mayor número de posibles hipótesis de conflictos con los procesos de recuperación correspondientes.

### **Método Piloto**

Este método implica instalar un nuevo sistema en un área de la empresa y luego de aprobado y puesto a punto, extenderlo al resto de la misma (mediante conversiones en directo o paralelo). El sistema se prueba en su totalidad en un ámbito restringido de la empresa (una sucursal, un departamento, con un grupo de clientes, proveedores, etcétera) para validarlo y luego se generaliza. Un ejemplo puede ser un sistema de venta en locales, se prueba en un local piloto.

Este sistema es recomendable cuando existe un ámbito de prueba apropiado. Un inconveniente que puede surgir es que en el resto de los locales o departamentos de la empresa, surjan nuevos requerimientos no detectados o definidos previamente.

### **Método por Etapas**

Este método implica instalar un nuevo sistema en forma gradual. Esto significa que en cada etapa del proceso de instalación se va haciendo experiencia con el sistema nuevo, se va mejorando en función de nuevos requerimientos y superación de problemas que se van encontrando.

**Ventajas:** Cada etapa se va realizando con los recursos necesarios. Esto permite implementar nuevos sistemas con recursos más limitados.

**Desventajas:** el período de implementación puede extenderse demasiado, lo cual podría provocar falta de apoyo de los usuarios. Asimismo, si existieran problemas iniciales en la

implementación, podrían difundirse apreciaciones negativas que perjudicaran el proceso en el futuro.

Un ejemplo de aplicación de este método puede ser la instalación de cajeros automáticos. En una primera etapa podrían habilitarse transacciones de consulta de saldos y retiros. En una segunda, podrían habilitarse transacciones de depósito y transferencia entre cuentas. En una tercera, podrían ponerse en funcionamiento el pago de servicios a través de la red de cajeros y en una última etapa, podrían interconectarse dos redes de cajeros automáticos diferentes, para ofrecer mayores servicios a los clientes de ambas redes.

