



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMATICA Y ELECTRONICA
ESCUELA DE INGENIERÍA EN SISTEMAS

**“ANÁLISIS DE METODOLOGIAS DE REINGENIERÍA DE SISTEMAS
INFORMÁTICOS CASO PRÁCTICO GESTIÓN DE PROYECTOS DE GRADO DEL
CIPFIE-ESPOCH”**

TESIS DE GRADO

Previa a la obtención del título de:

INGENIERO EN SISTEMAS INFORMÁTICOS

Presentado por:

MARIELA ELIZABETH ZUMBA HIDALGO

RAÚL HUMBERTO CUZCO NARANJO

Riobamba – Ecuador

- 2012 -

AGRADECIMIENTO

Ante todo a nuestro Dios nuestro creador, quien esta presente siempre en nuestros corazones guiándonos por el camino correcto día tras día, agradecer de todo corazón a todos los maestros de nuestra querida Facultad por habernos impartido tan valiosos conocimientos.

Un especial agradecimiento a nuestras amadas familias, padres, hermanos, hijos y a todos quienes nos han rodeado de cariño y brindado de una u otra forma su apoyo incondicional a lo largo de nuestra carrera, a la vez que con su ejemplo han sido fuente de inspiración para nuestra superación y lucha constante en el aspecto profesional.

No podemos dejar pasar la oportunidad para agradecer también a nuestros grandes amigos/as que conocimos a lo largo de nuestra carrera, con quienes compartimos varias horas de estudio, de alegría, tristezas y de satisfacción, de logros y esperanzas.

“Mil gracias a todos”

Mariela, Raúl

DEDICATORIA

Todo el esfuerzo y dedicación que están reflejados en esta Tesis lo dedico a mi hija querida quien ha sido mi motor para mi superación, para esforzarme y culminar mi carrera.

A mis queridos padres Graciela y Benjamín también dedico este trabajo, a quienes debo todo lo que soy, ya que con su cariño, consejos y rectitud me brindaron su apoyo en cada momento de mi vida.

Mariela Z.

FIRMAS DE RESPONSABLES Y NOTA

NOMBRES	FIRMA	FECHA
Ing. Iván Menés DECANO DE LA FACULTAD DE INFORMÁTICA Y ELECTRÓNICA	_____	_____
Ing. Raúl Rosero DIRECTOR DE LA ESCUELA DE INGENIERÍA EN SISTEMAS	_____	_____
Ing. Raúl Rosero DIRECTOR DE TESIS	_____	_____
Dr. Geovanny Vallejo MIEMBRO DEL TRIBUNAL	_____	_____
Tlg. Carlos Rodríguez DIRECTOR DEL CENTRO DE DOCUMENTACIÓN	_____	_____

Nota: _____

DECLARACIÓN

“Nosotros, Mariela Elizabeth Zumba Hidalgo y Raúl Humberto Cuzco Naranjo, somos responsables de las ideas, doctrinas y resultados expuestos en esta tesis; y, el patrimonio intelectual de la Tesis de Grado pertenecen a la Escuela Superior Politécnica de Chimborazo”

Firma

Firma

INDICE GENERAL

PORTADA

AGRADECIMIENTO

DEDICATORIA

FIRMAS DE RESPONSABILIDAD

DECLARACIÓN

INDICE GENERAL

INDICE DE ABREVIATURAS

INDICE DE FIGURAS

INDICE DE TABLAS

INDICE DE ANEXOS

INTRODUCCION

CAPITULO I

1. MARCO REFERENCIAL

1.1 ANTECEDENTES- 14 -

1.2 JUSTIFICACIÓN- 16 -

1.3 OBJETIVO- 17 -

1.3.1 OBJETIVO GENERAL.....- 17 -

1.3.2 OBJETIVOS ESPECÍFICOS- 17 -

1.4 HIPÓTESIS- 17 -

CAPITULO II

2. MARCO TEORICO

2.1 ESTUDIO DE LA REINGENIERIA DE SISTEMAS INFORMÁTICOS- 18 -

2.1.1 INTRODUCCIÓN- 18 -

2.1.2 CONCEPTOS Y TÉRMINOS DE REINGENIERÍA- 19 -

2.1.3 DEFINICIÓN DE REINGENIERÍA.....- 19 -

2.1.4 PROPÓSITO DE LA REINGENIERÍA.....- 21 -

2.1.5 HERRAMIENTAS DE REINGENIERÍA- 21 -

2.1.6 FASES DE LA REINGENIERÍA.....- 25 -

2.1.6.1 Justificación de la reingeniería- 25 -

2.1.6.2 Análisis de los niveles de Calidad y automatización de aplicaciones	26 -
2.1.6.3 Estimación de costo/beneficio	27 -
2.1.7 ETAPAS DEL PROCESO DE REINGENIERÍA	29 -
2.1.8 VENTAJAS DE LA REINGENIERÍA	33 -

CAPITULO III

3. METODOLOGÍAS DE LA REINGENIERÍA DE SOFTWARE

3.1. MÉTODOLOGÍA DE REESTRUCTURACIÓN.....	34 -
3.2. METODOLOGÍA DE ANÁLISIS DE OPCIONES PARA REINGENIERÍA (OAR)	37 -
3.3. METODOLOGÍA DE HERRADURA.....	44 -
3.4 METODOLOGÍA CÍCLICA	47 -
3.5 METODOLOGÍA A EMPLEAR PARA EL DESARROLLO DE LA.....	49 -
3.5.1 METODOLOGÍA DE DESARROLLO	51 -
3.5.1.1 Análisis de Inventario	53 -
3.5.1.2 Reestructuración de Documentos.....	53 -
3.5.1.3 Ingeniería Inversa	54 -
3.5.1.4 Reestructuración del Código	54 -
3.5.1.5 Reestructuración de Datos	54 -
3.5.1.6 Ingeniería Hacia Adelante.....	54 -
3.5.2 METODOLOGÍA DE PROGRAMACIÓN	54 -

CAPÍTULO IV

4. REINGENIERÍA DEL SISTEMA SIGEP

4.1 ANÁLISIS DE INVENTARIO	55 -
4.2 REESTRUCTURACIÓN DE DOCUMENTOS	61 -
4.3 INGENIERÍA INVERSA	61 -
4.4 REESTRUCTURACIÓN DE DATOS	61 -
5.1 INTRODUCCIÓN	62 -

CAPÍTULO V

INGENIERÍA HACIA ADELANTE DEL SISTEMA SIGEP

CASO PRÁCTICO, "ANÁLISIS DE METODOLOGÍAS DE REINGENIERIA DE SISTEMAS INFORMATICOS CASO PRÁCTICO GESTIÓN DE PROYECTOS DE GRADO DEL CIPFIE-ESPOCH"

5.2 CAPTURA DE REQUERIMIENTOS - 62 -	
5.2.1 REQUERIMIENTOS FUNCIONALES.....	63 -
5.2.2 ACTORES.....	67 -
5.2.3 MODELO DE CASOS DE USO.....	67 -
5.2.4 ESPECIFICACIÓN DE CASOS DE USO	75 -
5.3 ANÁLISIS.....	82 -
5.3.1. MODELO DE OBJETOS	82 -

5.3.2 DIAGRAMA DE SECUENCIA.....	- 88 -
5.3.3 INTERFACES DE USUARIO.....	- 97 -
5.4 DISEÑO.....	- 103 -
5.4.1 DISEÑO DE MODELADO DE OBJETOS.....	- 103 -
5.4.2 DISEÑO DE LA ARQUITECTURA.....	- 103 -
5.4.3 DIAGRAMA DE DESPLIEGUE.....	- 104 -
5.4.4 DIAGRAMA DE COMPONENTES.....	- 105 -
5.5 IMPLEMENTACIÓN.....	- 105 -
5.5.1 PLATAFORMA.....	- 105 -
5.5.2 HERRAMIENTAS A UTILIZARSE PARA EL DESARROLLO.....	- 106 -
5.5.2.1. Herramientas para Modelado.....	- 106 -
5.5.2.2. Herramientas para desarrollo de Interfaz de Usuario.....	- 109 -
5.5.2.3. Herramientas para desarrollo de Base de Datos.....	- 110 -
5.5.2 CONSTRUCCIÓN DEL SISTEMA.....	- 112 -
5.6 PRUEBAS.....	- 112 -

CONCLUSIONES

RECOMENDACIONES

RESUMEN

SUMARY

GLOSARIO DE TERMINOS

BIBLIOGRAFIA

ANEXOS

INDICE DE ABREVIATURAS

OAR. Análisis de Opciones para Reingeniería

ECE. Establecimiento del contexto de extracción

IC. Inventario De Componentes

ACC. Análisis de componentes candidatos

POE. Plan de opciones de extracción

SOE. Selección de opciones de extracción

POO. Programación Orientada a Objetos)

UML. Unified Modeling Language

MDA. Model Driven Architecture

HTML. Hyper Text Markup Language (Lenguaje de Marcado de Hipertexto)

ASP. Active Server Pages. Páginas de Servidor Activo

INDICE DE ILUSTRACIONES

Ilustración II. 1 Esquema para determinar aplicaciones prioritarias en el proceso de reingeniería.....	- 26 -
Ilustración II. 2 Esquema del proceso de ingeniería inversa	- 31 -
Ilustración II. 3 Esquema del proceso de ingeniería hacia adelante	- 32 -
Ilustración III. 4 Beneficios reestructuración del código.....	- 36 -
Ilustración III. 5 Herramienta de reestructuración de datos	- 37 -
Ilustración III. 6 Actividades del método OAR.....	- 38 -
Ilustración III. 7 Modelo de Herradura.....	- 45 -
Ilustración III. 8 Modelo Cíclico de la Reingeniería del Software.....	- 47 -
Ilustración III. 9 Proceso básico de Reingeniería.....	- 50 -
Ilustración III. 10 Fases de la reingeniería del sistema Sigep	- 52 -
Ilustración V. 11 Casos de uso de Secretaria-CIFIE Ingresar Anteproyecto	- 67 -
Ilustración V. 12 Casos de uso de Secretaria-CIFIE Registrar Matrícula Proyecto -	68 -
Ilustración V. 13 Casos de uso de Secretaria-CIFIE Buscar	- 69 -
Ilustración V. 14 Casos de uso de Secretaria-CIFIE Imprimir Actas.....	- 70 -
Ilustración V. 15 Casos de uso de Secretaria-CIFIE Mostrar Listados.....	- 71 -
Ilustración V. 16 Casos de uso de Secretaria-CIFIE Ingresar Proyecto de Investigación	- 72 -
Ilustración V. 17 Casos de uso de Secretaria-CIFIE Ingresar Evento	- 73 -
Ilustración V. 18 Casos de uso de Secretaria-CIFIE Ingresar Publicaciones.....	- 74 -
Ilustración V. 19 Modelo de objetos para ingresar Anteproyecto.....	- 83 -
Ilustración V. 20 Modelo de objetos para registrar matrícula	- 84 -
Ilustración V. 21 Modelo de objetos para buscar	- 85 -
Ilustración V. 22 Modelo de objetos para imprimir actas.....	- 85 -
Ilustración V. 23 Modelo de objetos para mostrar listados	- 86 -
Ilustración V. 24 Modelo de objetos para ingresar proyecto de investigación.....	- 86 -
Ilustración V. 25 Modelo de objetos para ingresar evento científico	- 87 -
Ilustración V. 26 Modelo de objetos para ingresar publicación	- 87 -
Ilustración V. 27 Diagrama de Secuencia validar usuario.....	- 88 -
Ilustración V. 28 Diagrama de Secuencia de Ingresar proyecto	- 89 -
Ilustración V. 29 Diagrama de Secuencia Registrar Matrícula	- 90 -
Ilustración V. 30 Diagrama de Secuencia Buscar.....	- 91 -
Ilustración V. 31 Diagrama de Secuencia imprimir actas	- 92 -
Ilustración V. 32 Diagrama de Secuencia mostrar listados.....	- 93 -
Ilustración V. 33 Diagrama de Secuencia ingresar proyecto de investigación	- 94 -
Ilustración V. 34 Diagrama de Secuencia ingresar evento científico.....	- 95 -
Ilustración V. 35 Diagrama de Secuencia registrar publicaciones.....	- 96 -

Ilustración V. 36	Pantalla validar usuario	- 97 -
Ilustración V. 37	Pantalla de ingreso anteproyecto	- 98 -
Ilustración V. 38	Pantalla de registro de matrícula.....	- 99 -
Ilustración V. 39	Pantalla de búsqueda de Anteproyecto.....	- 100 -
Ilustración V. 40	Pantalla de búsqueda de proyectos para defensa	- 100 -
Ilustración V. 41	Pantalla para Ingresar Proyecto de Investigación	- 101 -
Ilustración V. 42	Pantalla para Ingresar evento científico	- 101 -
Ilustración V. 43	Pantalla de ingreso de publicación.....	- 102 -
Ilustración V. 44	Diagrama de Red	- 103 -
Ilustración V. 45	Diagrama de Despliegue del Sistema.....	- 104 -
Ilustración V. 46	Diagrama de Componentes del Sistema	- 105 -
Ilustración V. 47	Rational Rouse Herramienta para modelado	- 107 -
Ilustración V. 48	StarUML Herramienta para modelado	- 108 -
Ilustración V. 49	Microsoft Visio Herramienta para modelado	- 109 -
Ilustración V. 50	Macromedia Dreamweaver programa para desarrollo de páginas web	- 110 -
Ilustración V. 51	ProgreSQL Herramienta para administracion de Base de Datos.....	- 111 -

INDICE DE TABLAS

Tabla.II.I Análisis comparativo de metodologías	- 50 -
Tabla IV. II Datos generales sistema Sigep actual.....	- 56 -
Tabla IV. III Ambiente externo Hardware sistema Sigep actual	- 58 -
Tabla IV. IV Ambiente externo Software sistema Sigep actual	- 59 -
Tabla IV. V Ambiente interno sistema Sigep actual	- 60 -
Tabla V. VI Requerimiento 1 "Anteproyectos de Tesis"	- 64 -
Tabla V. VII Requerimiento 2 "Proyecto de Tesis"	- 64 -
Tabla V. VIII Requerimiento 3 "Documentos"	- 65 -
Tabla V. IX Requerimiento 4 "Consultas"	- 66 -
Tabla V. X Requerimiento 5 "Reportes"	- 66 -
Tabla V. XI Especificación de caso de uso validar usuario.....	- 75 -
Tabla V. XII Especificación de caso de uso Asignar Anteproyecto.....	- 76 -
Tabla V. XIII Especificación de caso de uso ingresar matrícula	- 77 -
Tabla V. XIV Especificación de caso de uso listar tesis vigentes.....	- 78 -
Tabla V. XV Especificación de caso de uso listar tesis vigentes.....	- 79 -
Tabla V. XVI Especificación de caso de uso mostrar listados.....	- 80 -
Tabla V. XVII Especificación de caso de uso ingresar proyecto de investigación	- 81 -
Tabla V. XVII Especificación de caso de uso ingresar evento científico	- 82 -
Tabla V. XIX Especificación de caso de uso ingresar publicaciones.....	- 82 -
Tabla V. XX Caso de Prueba Validar usuario.....	- 114 -
Tabla V. XXI Caso de Prueba registrar usuario	- 115 -
Tabla V. XXII Caso de Prueba buscar.....	- 115 -
Tabla V. XXIII Caso de Prueba Imprimir Actas.....	- 115 -
Tabla V. XXIV Caso de Prueba Mostrar Listados	- 115 -
Tabla V. XXV Caso de Prueba Ingresar proyecto de investigación	- 116 -
Tabla V. XXVI Caso de Prueba Ingresar proyecto científico	- 116 -
Tabla V. XXVII Caso de Prueba Ingresar publicaciones	- 116 -
Tabla V. XXVIIDemostración de Hipótesis.....	- 120 -

INDICE DE ANEXOS

CAPÍTULO I

MARCO REFERENCIAL

1.1 ANTECEDENTES

Actualmente la creación y modificación de programas de computadora es una tarea principalmente manual, y una tarea difícil e imprevisible. Los programas grandes suelen ser más complejos y más difíciles de depurar. Aunque la tecnología todavía está en su infancia, la reingeniería del software está empezando a tomar algunas tareas de programación, particularmente las tareas menos creativas, más repetitivas y las automatiza. Estos programas de reingeniería, escritos en idiomas especialmente diseñados, operan en el código fuente de los programas y realizan una variedad de análisis y modificaciones.

La Reingeniería de Software es analizar sistemas existentes para entender su diseño y desarrollar estrategias para extraer y reutilizar componentes. Implica la Rehabilitación y modernización de los componentes modificándolos con la ayuda de herramientas de forma automática o semi-automática.

La Reingeniería de Sistemas Informáticos ha sido y es un tema muy importante para muchas empresas, instituciones y organismos que tienen que seguir manteniendo sus aplicaciones porque sus desarrollos han sido costosos y adaptados a sus necesidades, lo que en muchos casos hace que no existan aplicaciones comerciales similares. El inconveniente es que estos sistemas con el paso de los años presentan un aspecto obsoleto, mostrando pantallas y diseños ya descartados.

La Reingeniería de Sistemas tiene por finalidad reestructurar o transformar viejos Sistemas en aplicaciones más fáciles de mantener, con entornos más agradables e integradas en nuevas plataformas de hardware/software.

Algunas de las ventajas que proporciona la Reingeniería de Sistemas son:

- ✓ Proporcionar asistencia personalizada a las tareas de mantenimiento y evolución del software.
- ✓ Reducir errores y costes.
- ✓ Mejorar los procesos de conversión y migración.
- ✓ Extender el ciclo de vida de las aplicaciones.
- ✓ Facilitar el entendimiento de los programas.

El departamento de la CIPFIE de la Escuela Superior Politécnica de Chimborazo cuenta con un sistema de Gestión de Proyectos el mismo que en la actualidad no está funcionando al cien por ciento.

El análisis de las metodologías para la Reingeniería, servirá para elegir en base a sus características una que se adapte a las necesidades de este Sistema que permita mejorar su productividad a bajo costo.

1.2 JUSTIFICACIÓN

En los últimos años ha surgido una nueva tendencia en el desarrollo de las empresas e instituciones que ha sido el resultado de los cambios cada vez más rápidos dentro del entorno de la misma. La Reingeniería de Sistemas viene a dar la pauta para nuevos cambios en la forma de operar de las mismas.

Cuando un Sistema Informático ha servido para las necesidades de una compañía o Institución por varios años, se vuelve inestable, debido a las correcciones, adaptaciones y mejoras que se han realizado. Esto deriva en que cada vez que se intenta efectuar un cambio se produzcan efectos colaterales graves e inesperados. Por esta razón es conveniente utilizar la *Reingeniería del Software*.

El análisis comparativo de las metodologías de Reingeniería de Sistemas Informáticos, que se llevará a cabo en esta investigación permitirá elegir una que se adapte a las necesidades del sistema de la CIPFIE y para posteriores sistemas con las mismas características.

La importancia de esta investigación radica en que todos los sistemas son creados de acuerdo a las necesidades de las empresas o instituciones con los programas y plataformas actuales, sin embargo como ya lo mencionamos el paso del tiempo y avances en la tecnología, origina una incompatibilidad en los sistemas o lenguajes utilizados por lo que es importante conocer como llevar a cabo una Reingeniería y sobre todo elegir una metodología adecuada según el Sistema Informático utilizado, con el fin de lograr la máxima productividad de ese Sistema.

OBJETIVO

1.3.1 OBJETIVO GENERAL

Analizar las metodologías de Reingeniería de Sistemas Informáticos para aplicarlo al Sistema de Gestión de Proyectos de Grado del CIPFIE-ESPOCH.

1.3.2 OBJETIVOS ESPECÍFICOS

- ✓ Investigar las diferentes metodologías de Reingeniería.
- ✓ Establecer las características que tiene cada una de estas metodologías.
- ✓ Evaluar los resultados obtenidos de la metodología de Reingeniería.
- ✓ Implementar con la mejor metodología seleccionada para el Sistema de centro de Investigaciones de la FIE.

1.4 HIPÓTESIS

Mediante el análisis de las metodologías y en selección de la mejor se optimizará la funcionalidad del Sistema de Gestión de Proyectos de Grado del CIPFIE-ESPOCH.

CAPITULO II

MARCO TEÓRICO

2.1 ESTUDIO DE LA REINGENIERIA DE SISTEMAS INFORMÁTICOS

2.1.1 INTRODUCCIÓN

En la actualidad cualquier entidad ya sea educacional, gubernamental o empresarial dependen en total medida del rendimiento que tengan sus **Sistemas Informáticos**. Resulta limitado imaginar el funcionamiento de una empresa, sin la intervención de algún Sistema computacional, pues estos están involucrados de algún modo en la organización, ventas, desarrollo, producción, manufactura, inventarios, publicidad, mercadotecnia, personal, costos; de manera resumida casi en todo. Una empresa, no es una entidad en sí misma, sino que está basada en productos software. La vida actual se encuentra en constante cambio y por lo tanto existe un desarrollo e innovación cada vez más rápido; y por supuesto las empresas no están excluidas de dicho cambio y se ven obligadas a actualizar constantemente sus sistemas computacionales, pues sin estos no podrían estar a la altura de las necesidades que exigen sus clientes. La gran competencia y demanda que existe en el mercado, exige que las empresas tengan lo último en tecnología, por lo que la actualización y el perfeccionamiento del software son absolutamente imprescindibles. Y es aquí

precisamente donde entran en juego las estrategias para realizar cambios en el producto software. Por lo tanto hoy en día la creación y modificación de programas de computadora es una tarea principalmente manual, y una tarea difícil e imprevisible. Los programas grandes suelen ser más complejos y más difíciles de depurar. Aunque la tecnología todavía está en su infancia, la Reingeniería del Software está empezando a tomar algunas tareas de programación, particularmente las tareas menos creativas, más repetitivas y las automatiza.

La Reingeniería de Software es analizar sistemas existentes para entender su diseño y desarrollar estrategias para extraer y reutilizar componentes. Implica la Rehabilitación y modernización de los componentes modificándolos con la ayuda de herramientas de forma automática o semi-automática.

2.1.2 CONCEPTOS Y TÉRMINOS DE REINGENIERÍA

Reingeniería: Transformación sistemática de un sistema existente.

Reingeniería de software: Modificación de un producto software, o de ciertos componentes, usando para el análisis del sistema existente técnicas de Ingeniería Inversa y, para la etapa de reconstrucción, herramientas de Ingeniería Directa, de tal manera que se oriente este cambio hacia mayores niveles de facilidad en cuanto a mantenimiento, reutilización, comprensión o evaluación.

2.1.3 DEFINICIÓN DE REINGENIERÍA

- ✓ La Reingeniería es la transformación sistemática de un Sistema existente dentro de una nueva forma de realizar el mejoramiento de la calidad en las operaciones, la capacidad del Sistema, la funcionalidad, el rendimiento evolutivo a bajo costo y las agendas o riesgos para el cliente.

- ✓ La Reingeniería de Software es analizar Sistemas existentes para entender su diseño y desarrollar estrategias para extraer y reutilizar componentes. Implica la Rehabilitación y modernización de los componentes modificándolos con la ayuda de herramientas de forma automática o semi-automática.
- ✓ Es un proceso mediante el cual se mejora un software existente haciendo uso de técnicas de ingeniería inversa y reestructuración de código.
- ✓ La reingeniería de software sirve para extraer información de diseño y programación de Sistemas existentes, para así crear sistemas nuevos sin tener que comenzar desde cero.
- ✓ La Reingeniería prepara o perfecciona Software en sí mismo, haciéndole más fácil de mantener, propiciando su evolución o haciéndole reutilizable
- ✓ La Reingeniería de Sistemas Informáticos recupera información sobre el diseño de un programa existente y utiliza esta información para reestructurar o reconstruir el programa existente, con vistas a adaptarlo a un cambio, a ampliarlo o a mejorar su calidad general, con el objetivo de conseguir una mayor facilidad de mantenimiento en el futuro (esto es lo que se denomina mantenimiento preventivo).

2.1.4 PROPÓSITO DE LA REINGENIERÍA

El propósito de la reingeniería es que los sistemas existentes tomen ventajas de las nuevas tecnologías además de habilitar el nuevo esfuerzo de desarrollo para que se aproveche las ventajas de reutilizar sistemas existentes. La Reingeniería tiene el potencial de mejorar la productividad y calidad del software a través de todo el ciclo de vida. La Reingeniería casi siempre implica cambiar la forma de un programa y mejorar su documentación. En este caso, la funcionalidad del programa no es modificada; sólo su forma se presta a modificaciones. En otros casos, la Reingeniería va más allá de la forma e incluye rediseñar la funcionalidad del programa para buscar mejores requerimientos de usuario.

Es examinar Sistemas Informáticos o Aplicaciones con el fin de perfeccionarlos, reestructurarlos o reconstruirlos, de modo que sea más fácil de mantenerlos y que muestren una mayor calidad, propiciando así su evolución y haciéndole reutilizable, evitando de esta manera el "envejecimiento del Software".

2.1.5 HERRAMIENTAS DE REINGENIERÍA

Tipos de Herramientas de Reingeniería

Son una parte importante de la Reingeniería al cambiar sustancialmente la productividad y calidad del trabajo de mantenimiento. A continuación se muestra ocho tipos básicos de herramientas. Estas no solamente ayudan a los programadores a realizar pruebas más eficientes, sino también mucho más completas e intensivas de lo que es posible con técnicas manuales. Pruebas como las de descubrir códigos muertos o variables inutilizables en programas largos y complejos antes eran imposibles y ahora con la ayuda de herramientas de reingeniería se pueden realizar fácilmente.

Estas herramientas se encuentran clasificadas como sigue:

Analizadores de programa

- ✓ Rastreadores de lógica y de datos
- ✓ Referencias cruzadas
- ✓ Delineadores (Profilers)

Métricas

- ✓ Programas monitores de estándares
- ✓ Programas analizadores de calidad
- ✓ Programas controladores de complejidad

Reestructuración

- ✓ Reestructuradores de procesos lógicos
- ✓ Estandarización de nombres de datos y definiciones

Ingeniería inversa

- ✓ Ingeniería inversa de datos
- ✓ Ingeniería inversa de lógica

Pruebas

- ✓ Generadores de datos para pruebas
- ✓ Analizadores de alcance para pruebas
- ✓ Debuggers
- ✓ Comparadores

Convertidores

- ✓ De lenguaje

Manejadores de configuración y cambios

- ✓ Manejadores de control de cambio
- ✓ Manejadores de librerías
- ✓ Generadores de código

Herramientas de redocumentación

Productos listados en orden alfabético	Análizadores de programas	Métricas	Reestructuración	Ingeniería Inversa	Pruebas	Convertidores	Manejo de configuración y cambios	Herramientas de recomendación
PL/I to COBOL/COBOL II Translator						1		
PL/M to C Translator						1		
Pro Struct			1,2					
Product Configuration Management System (V4.0.5)							1,2	
Profile Code (V3.20)	3							
PWCS Configuration Builder (V5.1)							1,2	1
PWCS for Software Configuration Management							1,2	
PWCS Version Manager (V5.1)							1,2	
Q/Advisor COBOL (V2.4)	1,2	1,2,3						
Q/Advisor COBOL for Windows (V2.4)	1,2	1,2,3						
QA FORTRAN (V6.0)		1,2,3	1,2		1,4			
Software Configuration Management Supervisor2 (SCMS2)							1,2	1
Software Profile Management Facility	3							
Software Refinery (V4.0)	1,2			1,2	1,2,4		3	1,2
Source Program Compare					4			1
STW/Advisor	1,2	1,2,3						
STW/Advisor for Windows	1,2	1,2,3						
Turbo Debugger and Tools (V3.0)	1,2,3				3			
VIA/Insight2	1,2			1,2				
VIA/Recap	1,2	1,2,3						
Visual Debugger-14	1							
Xinotech COBOL Composer (V2.0)	1,2			1,2				
XperCASE (V3.5)			1,2	1,2				1,2
XPONENT Configuration Management (CCM) for Windows							1,2	1

- ✓ Referencias cruzadas
- ✓ Generadores de diagrammas

Productos listados en orden alfabético	Analizadores de programas	Métricas	Reestructuración	Ingeniería Inversa	Pruebas	Convertidores	Manejo de configuración y cambios	Herramientas de relocalización
Aide-De-Camp Software Configuration Management System (V9.0)							1,2	1
AISLE/QualGen	1,2	1,2,3						1,2
ADK Configuration Management and Version Control (CMVC) (V2.2)							1,2	
Answer Compare/CDF-15.20					4			1
C Set++ Debugger	1				3			
C-DCC Documentation Tool for C and C++ (V5.0)								1,2
C-DCC Professional (V5.0)								1,2
C-Metric		1,2,3						
CA-PanL CM Configuration Manager							1,2	
CCC Manager (Change and Configuration Control) (V2.1)							1,2	
CISLECD ADL	1,2	1,2						
COBOL ANALYST (V3.0)	1,2							1,2
COBOL Structuring Facility (V3.0)	1,2	1,2	1,2					
CodeBreaker (V4.1)				1,2	4			
Codecheck/2 (V5.0)	1,2	1,2,3						
Coverage	1,2							
CrossView Debugger for Windows	1				3			
DCD-PC	1,2			1,2				1,2
ETI Source Compare					4			1
For Struct (V2.0)			1,2					
Language Translator						1		
Legacy Workbench (V5.1)	1,2	1,2,3	1,2	1,2				1,2
Legacy Workbench for Windows (V5.1)	1,2	1,2,3	1,2	1,2				1,2
LISP to C Translator (Rel.3.2)						1		
Logscope	1,2	1,2,3	1,2	1,2				
MASM to C Translator						1		
McCabe Slice Tool (V4.1)	1,2				2			
Microsoft Source Profiler (V1.0)	2,3				4			
Microsoft Source Profiler for Windows (v.1.0)	2,3				4			
MultiScope Debugger for Windows (V2.01)	1				3			
Non-IBM COBOL to IBM COBOL/COBOL II Translator						1		
PC/MC/UX/VCMetric (V4.0)	1,2	1,2,3						1
PL/1 to C Translator						1		

2.1.6 FASES DE LA REINGENIERÍA

2.1.6.1 Justificación de la reingeniería

Esto implica, por un lado, convencer a la dirección sobre el proceso de reingeniería, necesidad imperiosa de cambiar, creando a posteriori un comité de dirección destinado a hacerse cargo del proyecto de Reingeniería. Por otro lado, en esta misma fase se deberá preparar a la fuerza de trabajo para el compromiso y el cambio. La mayoría de las organizaciones sólo toman en consideración los procesos de Reingeniería cuando el costo de un nuevo desarrollo es de masiado alto. En cualquier caso, y aunque a primera vista parezca la única o la mejor opción, es necesario confirmar la necesidad de reconstruir el Sistema. Para esto se puede dar una idea de los costos del proyecto y del valor del Software actual dentro del negocio mediante algunos elementos como: evaluaciones de costo del mantenimiento; para lo cual se recomienda tres criterios para medir los procesos de mantenimiento: "dominio del impacto" o proporción de instrucciones y elementos de datos afectados por una tarea de mantenimiento con respecto al total de instrucciones y elementos de datos del sistema; "esfuerzo empleado", que es el número de horas dedicadas a tareas de mantenimiento, con lo que se puede obtener una media del número de horas por tarea de mantenimiento; y "tasa de errores de segundo nivel", que es el número de errores provocados por acciones de mantenimiento. Si se observa que estas tres medidas aumentan, es muy probable que los costos de mantenimiento se incrementen con el tiempo. El análisis de la calidad del software actual y la evaluación del valor de negocio del sistema será realizado por la máxima dirección de la organización.

2.1.6.2 Análisis de los niveles de Calidad y automatización de aplicaciones

En esta fase se determina la calidad técnica y el valor de negocio de cada aplicación medular en la empresa, con el objetivo de construir una lista de aplicaciones ordenada según sus prioridades en el proceso de reingeniería. La calidad técnica de un producto es una medida relativa, dependiente de cada organización, que se calcula en función de diversas características (complejidad ciclomática o errores/klcd).

Para cada variable que interviene en la calidad técnica se fijan límites inferior y superior que representan los valores máximos y mínimos de calidad. Para hallar el nivel de calidad de la variable considerada se puede utilizar la siguiente fórmula:

$$Ci = 1 - \frac{\text{Medida actual} - \text{Límite inferior}}{\text{Límite superior} - \text{Límite inferior}}$$

De donde se obtiene una gráfica que nos ayuda a determinar que aplicaciones son prioritarias y cuales no en el proceso de Reingeniería. Asociando un punto de un plano para cada aplicación, e interpretando el valor de negocio y la calidad técnica como coordenadas de estos puntos.



Ilustración 1 Esquema para determinar aplicaciones prioritarias en el proceso de Reingeniería

De esta forma, las aplicaciones ubicadas en el cuadrante superior izquierdo tienen alta calidad y bajo valor de negocio, por lo que no requieren Reingeniería; las situadas en el cuadrante inferior izquierdo tienen poco valor en ambos parámetros, por lo que pueden ser desarrolladas de nuevo o remplazadas por productos comerciales; las del superior derecho tienen un gran valor de negocio y alta calidad: se les puede aplicar Reingeniería, pero sin excesiva prioridad; las del inferior derecho tienen alto valor de negocio y baja calidad técnica, por lo que serán las primeras candidatas a la Reingeniería.

2.1.6.3 Estimación de costo/beneficio

El siguiente paso debe ser determinar los costos de cada proyecto de Reingeniería que se vaya a enfrentar: si éstos son superiores a los beneficios, la Reingeniería no será una opción viable y la aplicación deberá ser desarrollada de nuevo o bien adquirirse en el mercado.

Para estimar los costos de la Reingeniería, se tienen ciertas ventajas respecto a ese cálculo en proyectos de Ingeniería directa, debido a que como se apoyará el proceso sobre una aplicación ya realizada y que satisface ciertos requisitos, entonces no es necesario calcular factores influyentes como el número de líneas de código, sentencias ejecutables, elementos de datos, accesos a archivos, etc., ya que son medidas que se pueden tomar directamente de la aplicación.

Se recomienda utilizar como variables para calcular los costos las que se ofrecen a continuación, y que deben ponderarse en función de su influencia en el costo total.

- Número de líneas de código no comentadas.

- Costo de los casos de prueba, que se calcula multiplicando el costo medio de cada caso de prueba por el número de éstos, que es función de la complejidad ciclomática del problema.
- Número de accesos a archivos, bases de datos y campos. En la ponderación de estas entradas/salidas consideramos la complejidad de las estructuras de información y el grado de independencia de la aplicación respecto de los datos.
- Número de operaciones que realizan los usuarios de la aplicación, número de ventanas, número de informes, etc., para el caso de las interfaces de usuario.

Una vez que se ha calculado el costo de la Reingeniería, la **última etapa es compararlos** con los beneficios esperados.

El beneficio proporcionado para seguir manteniendo el producto sin Reingeniería es el siguiente:

$$BM = [P3 - (P1 + P2)] * P16$$

Deberá retocarse la fórmula cuando los diversos costos varíen de un año para otro. Si se desarrolla de nuevo el sistema, se obtiene este beneficio:

$$BD = [(P12 - (P10 + P11)) * (P16 - P14) - (P13 * P15)] - BM$$

El beneficio producido por la Reingeniería es:

$$BR = [(P6 - (P4 + P5)) * (P16 - P8) - (P7 * P9)] - BM$$

Donde:

P1 = Costo de mantenimiento actual para una aplicación (anual).

- P2 = Costo de operación de una aplicación (anual).
- P3 = Valor del negocio actual (anual).
- P4 = Costo previsto de mantenimiento tras la reingeniería (anual).
- P5 = Costo previsto de operaciones tras la reingeniería (anual).
- P6 = Valor de negocio previsto tras la reingeniería (anual).
- P7 = Costo estimado de la reingeniería.
- P8 = Duración estimada de la reingeniería.
- P9 = Factor de riesgo de la reingeniería.
- P10 = Costo previsto de mantenimiento tras el redesarrollo (anual).
- P11 = Costo previsto de operaciones tras el redesarrollo (anual).
- P12 = Valor de negocio previsto del nuevo sistema (anual).
- P13 = Costo estimado del redesarrollo.
- P14 = Duración estimada del redesarrollo.
- P15 = Factor de riesgo del redesarrollo.
- P16 = Vida esperada del sistema.

2.1.7 ETAPAS DEL PROCESO DE REINGENIERÍA

La Reingeniería de Sistemas Informáticos abarca las siguientes etapas:

2.1.7.1 Análisis de Inventario.- Este proceso consiste en el estudio de la antigüedad, importancia de la aplicación en el negocio y el proceso de mantenimiento actual, entre otros criterios, para estudiar la posible conveniencia de la Reingeniería. Todas las organizaciones de software deberían tener un inventario de todas sus aplicaciones. El inventario tal vez no sea más que un modelo en una hoja de cálculo que contenga información que proporcione una descripción detallada (tamaño, edad, importancia para el negocio) de las aplicaciones activas.

Los candidatos a la Reingeniería aparecen cuando se ordena esta información en función de su importancia para el negocio, longevidad, mantenibilidad actual y otros criterios localmente importantes. Es entonces cuando es posible asignar recursos a las aplicaciones candidatas para el trabajo de Reingeniería.

Es importante señalar que el inventario deberá visitarse con regularidad, el estado de las aplicaciones puede cambiar en función del tiempo y, como resultado, cambiarán las prioridades para la Reingeniería.

2.1.7.2 Reestructuración de Documentos.- En este proceso se puede optar por una de tres opciones: Evitar la documentación de los módulos estáticos que no van a sufrir cambios, documentar sólo lo que se va a modificar y documentar toda la información del Sistema, si es que este es fundamental para el negocio.

La documentación débil es la marca de muchos sistemas heredados. ¿Pero que se hace acerca de ellos? ¿Cuáles son las opciones? Crear documentación consume mucho tiempo, si el Sistema funciona vivirá con lo que tenga. La documentación debe actualizarse pero se tiene recursos limitados. Se utiliza un enfoque de "documentar cuando se toque". El sistema es crucial para el negocio y debe volver a documentarse por completo incluso en este caso un enfoque inteligente es recortar la documentación a un mínimo esencial. Cada una de estas opciones es viable. Una organización de software debe elegir la más apropiada para cada caso.

2.1.7.3 Ingeniería Inversa.- En este proceso se extraen modelos de alto nivel de abstracción que ayuden a la comprensión de la aplicación para poder modificarla y que sirvan como punto de partida para el siguiente proceso. Estos datos se deben almacenar en un repositorio que permita que las personas o herramientas que lleven a cabo los siguientes pasos lo encuentren disponible. De esta manera se conforma también la documentación de análisis y diseño de la aplicación que facilitará su posterior mantenimiento.

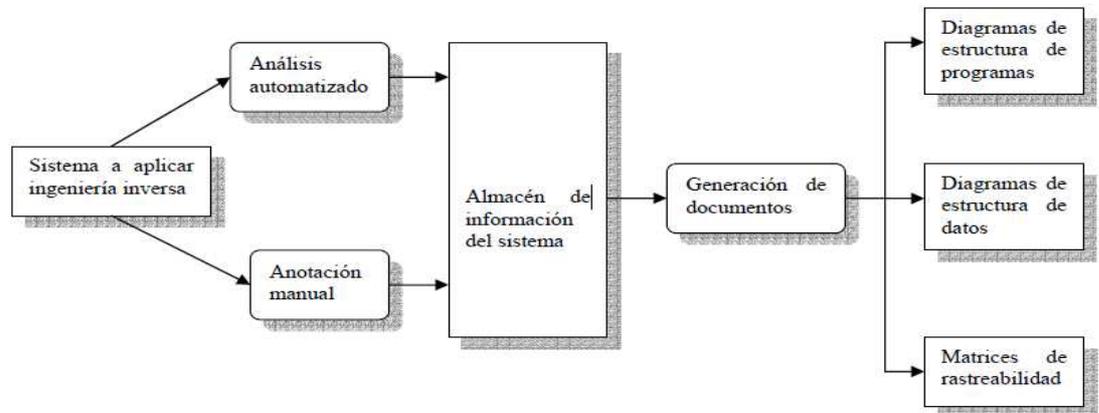


Ilustración 2 Esquema del proceso de Ingeniería Inversa

2.1.7.4 Reestructuración del Código.- La reestructuración de datos es una actividad de Reingeniería a gran escala. En la mayoría de los casos, la reestructuración de datos comienza con una actividad de ingeniería inversa. La arquitectura de datos actual se analiza con minuciosidad y se definen los modelos de datos necesarios, se identifican los objetivos de datos y los atributos, y después se revisa la calidad de las estructuras de datos existentes.

Cuando la estructura de datos es débil (por ejemplo, actualmente se implementan archivos planos, cuando un enfoque relacional simplificaría muchísimo el procesamiento), se aplica una reingeniería a los datos.

Dado que la arquitectura de datos tiene una gran influencia sobre la arquitectura del programa, y también sobre los algoritmos que lo pueblan, los cambios en datos darán lugar invariablemente a cambios o bien de arquitectura o bien de código.

2.1.7.5 Ingeniería hacia adelante/Ingeniería Directa.- En un mundo ideal, las aplicaciones se reconstruyen utilizando un "motor de reingeniería" automatizado. En el motor se insertaría el programa viejo, que lo analizaría, reestructuraría y después regeneraría la forma de exhibir los mejores aspectos de la calidad del software. Después de un espacio de tiempo corto, es probable que llegue a aparecer este

“motor”, pero los fabricantes de CASE han presentado herramientas que proporcionan un subconjunto limitado de estas capacidades y que se enfrentan con dominios de aplicaciones específicos. Lo que es más importante, estas herramientas de Reingeniería cada vez son más sofisticadas.

La ingeniería directa no solo recupera la información de diseño a partir del software existente, también utiliza esta información para alterar o reconstruir el sistema existente con la finalidad de mejorar su calidad global. En la mayoría de los casos el software sometido a reingeniería vuelve a implementar la función del sistema existente y también añade nuevas funciones o mejoras.

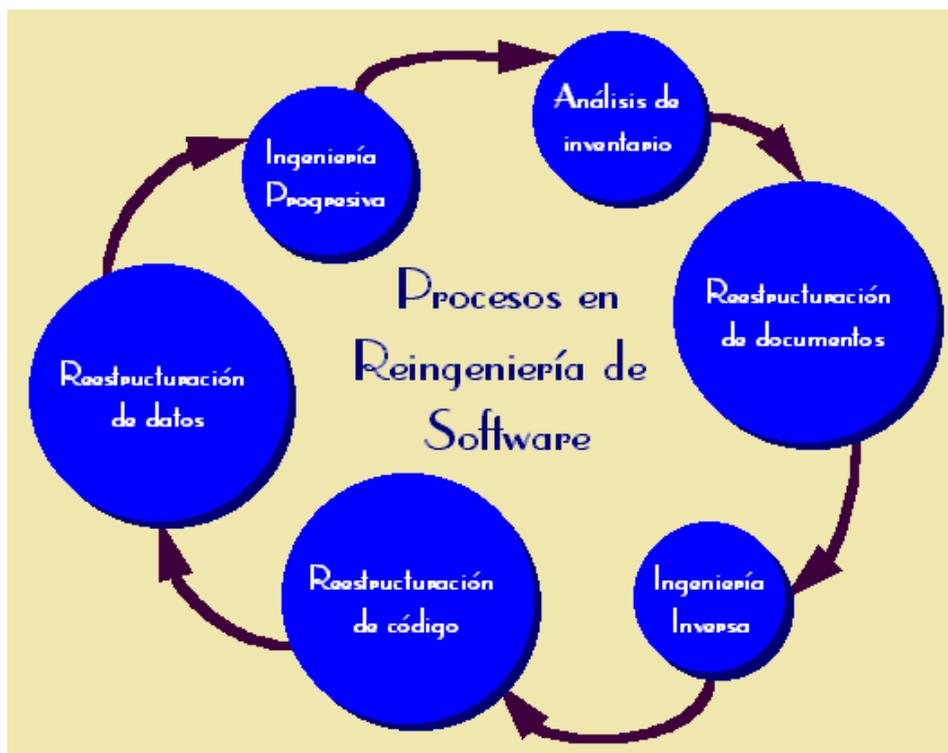


Ilustración 3 Esquema del proceso de Ingeniería hacia Adelante

2.1.8 VENTAJAS DE LA REINGENIERÍA

- ✓ Se reduce el riesgo en la actualización del software.
- ✓ Se reduce el costo en la actualización del software.
- ✓ Se aumenta la calidad y confiabilidad del software.
- ✓ Hace que el software mas fácilmente modificable
- ✓ Es un catalizador para la automatización del mantenimiento de software.
- ✓ Ayuda a las organizaciones a recuperar sus inversiones en software.
- ✓ Amplía las capacidades de las herramientas CASE
- ✓ Reducir los riesgos evolutivos en una organización.
- ✓ Puede actuar como catalizador para la aplicación de técnicas de inteligencia artificial para resolver problemas de Reingeniería

CAPITULO III

METODOLOGÍAS DE LA REINGENIERÍA DE SOFTWARE

3.1. MÉTODOLÓGÍA DE REESTRUCTURACIÓN

La reestructuración del software modifica el código fuente y/o los datos en un intento de hacerlo adecuado para cambios futuros. En general no modifica la arquitectura global del programa. Tiende a centrarse en los detalles de diseño individuales y en las estructuras de datos locales definidas dentro de los módulos. Si el esfuerzo de la reestructuración se extiende más allá de los límites de los módulos y abarca a la arquitectura del software, la reestructuración pasa a ser ingeniería progresiva.

Beneficios que se pueden lograr cuando se reestructura el software (ARNOLD):

- ✓ Da lugar a programas de mayor calidad, con mejor documentación y menos complejidad.

- ✓ Reduce la frustración entre los Ingenieros del Software que deben de trabajar con el programa, mejorando la productividad y haciendo más sencillo el aprendizaje.
- ✓ Reduce el esfuerzo requerido para llevar a cabo las actividades de mantenimiento.
- ✓ Hace que el software sea más sencillo de comprobar y de depurar.

La reestructuración se produce cuando la arquitectura básica de la aplicación es sólida y solamente existe un subconjunto de todos los módulos y todos los datos que requiera extensa modificación.

3.1.1. Reestructuración del código

Se lleva a cabo para conseguir un diseño que produzca la misma función pero con una mayor calidad que el programa original. Las técnicas de reestructuración del código modelan la lógica del programa empleando Algebra Booleana y a continuación aplican una serie de reglas de transformación que dan lugar a una lógica reestructurada.

3.1.2. Reestructuración de los datos.

Antes de que pueda comenzar la reestructuración de datos, es preciso llevar a cabo una ingeniería inversa, un análisis del código fuente. Todas las sentencias del lenguaje de programación que contengan definiciones de datos, descripciones de archivos, E/S y descripciones de interfaz se evalúan en primer lugar. El objetivo es extraer elementos y objetos de datos, para obtener información acerca del flujo de datos, así como comprender las estructuras de datos ya existentes que se hayan implementado. Esta actividad se denomina a veces análisis de datos.

Una vez finalizado el diseño comienza el rediseño de datos:

- ✓ Paso de estandarización de rediseño de datos: Clarifica las definiciones de datos para lograr una consistencia entre nombre de objetos de datos, o entre formatos de registro físicos en el seno de una estructura de datos o formato de archivo existente.
- ✓ Racionalización de nombres de datos: Asegura que todas las convenciones de denominación de datos se ajusten a los estándares locales y asegure también que se eliminen los alias a medida que fluyen los datos a través del sistema.

Cuando la reestructuración va más allá de la estandarización y de la racionalización, se efectúan modificaciones físicas en las estructuras de datos, ya existentes con objeto de hacer que el diseño de datos sea más efectivo. Esto puede significar una traducción de un formato de archivo a otro, o en algunos casos, una traducción de un tipo de base de datos a otra.

BENEFICIOS DE LA REESTRUCTURACION DE CODIGO

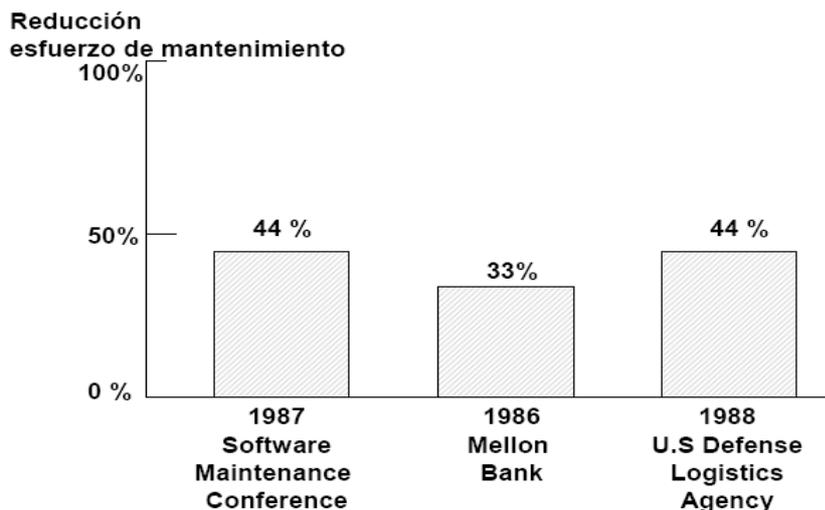


Ilustración 4 Beneficios reestructuración del código

HERRAMIENTA DE REESTRUCTURACION DE DATOS



Ilustración 5 Herramienta de reestructuración de datos

3.2. METODOLOGÍA DE ANÁLISIS DE OPCIONES PARA REINGENIERÍA (OAR)

El Análisis de Opciones para Reingeniería (OAR), es un método sistemático, de arquitectura central y de toma de decisiones para la identificación y extracción de componentes dentro de grandes y complejos sistemas de software, identifica componentes de arquitectura potencialmente relevantes y analiza los cambios requeridos para usarlos en una línea de producción de software o nuevas arquitecturas de software. En esencia, OAR proporciona un conjunto de opciones de extracción junto con estimación de costos, esfuerzo y riesgos asociados con estas opciones.

Los siguientes puntos son motivos para el cambio:

- Componentes existentes casi siempre eran pobremente estructurados y documentados.
- Componentes existentes diferían en niveles de granuralidad.
- No había una guía clara sobre como salvar componentes.

OAR proporciona un acercamiento sistemático para direccionar esos puntos y tomar decisiones requeridas para el costo efectivo y eficiente de extraer componentes de sistemas heredados. El método OAR consiste de cinco actividades principales con tareas escalables.

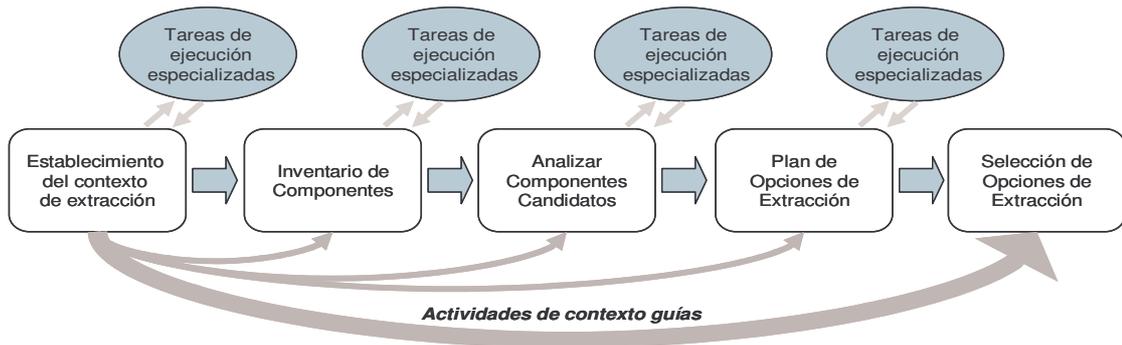


Ilustración 6 Actividades del método OAR

3.2.1 Establecimiento del contexto de extracción (ECE).

Consiste en entrevistar a los accionistas y estudiar la línea de producción de la organización o nuevos requerimientos de sistema, base heredada y expectativas para la extracción de componentes heredados.

3.2.2 Inventario De Componentes (IC).

Después del ECE, el equipo OAR identifica los componentes del sistema heredado que potencialmente pueden ser extraídos para usarlos en una línea de producción o en una nueva arquitectura de software, esta actividad resulta en un inventario de los componentes heredados candidatos. La actividad de IC tiene seis tareas:

1. Identificar características de los componentes necesarios:
 - Determina las características requeridas de los potenciales componentes heredados.
2. Identifica las características satisfactorias de los componentes:
 - Crea una tabla de componentes.
 - Filtra los componentes que no satisfacen las características.

3. Compara las necesidades de componentes:
 - Compara los componentes heredados.
4. Inventario de componentes candidatos:
 - Actualiza la tabla de componentes.
5. Produce tópicos de extracción
 - Revisa cualquier tópico de extracción.
6. Revisión del calendario OAR:
 - Actualiza el calendario de OAR.

3.2.3 Análisis de componentes candidatos (ACC).

Es analizar el conjunto de candidatos de componentes heredados para extraer los tipos de cambios que son requeridos. Esta actividad tiene seis tareas:

1. Selección de componentes deseables.
2. Identifica los componentes "Tal como están y de caja negra"
3. Identifica componentes de caja blanca.
4. Determina cambios requeridos.
5. Producción de tópicos de extracción.
6. Revisa el calendario OAR.

3.2.4 Plan de opciones de extracción (POE).

El equipo OAR también filtra una vez más los componentes candidatos y analiza el impacto de agregación de diferentes componentes.

El POE tiene siete tareas:

1. Selecciona componentes favorables.
2. Ejecución de intercambio de componentes
3. Forma opciones de componentes.

4. Determina costos comparativos y esfuerzos.
5. Analiza dificultad o riesgo.
6. Producción de tópicos de extracción.
7. Revisa el calendario OAR.

3.2.5 Selección de opciones de extracción (SOE).

Finalmente, los miembros del equipo seleccionan la mejor opción de extracción o combinación de opciones para programas y consideraciones técnicas, ellos preparan un resumen que presenta y justifica sus elecciones.

La actividad SOE tiene cinco tareas:

1. Elegir la mejor opción:
2. Verificación de opción:
3. Identifica componentes necesarios satisfechos.
4. Presentación de descubrimientos.
5. Producción de resumen.

3.2.6 Tareas Especializadas.

Cada actividad también tiene un potencial conjunto de actividades especializadas para direccionar circunstancias que pueden de otro modo imposibilitar el cumplimiento de la actividad. Estas tareas pueden aplicarse bajo las siguientes condiciones:

- El criterio existente.
- Mas trabajo puede ser requerido para que la actividad sea razonablemente.
- Se requieren datos adicionales para completar una tarea particular.

Existe una necesidad de complementar tareas estándares OAR

3.2.7 Estructura De Actividades

Cada actividad está compuesta de tareas y sub-tareas diseñadas para contestar un conjunto de preguntas de actividades específicas. Esas preguntas definirán la actividad y también servirán como una lista de comprobación para ser incluidas en los criterios de cada actividad.

3.2.8 Criterios de entrada.

Los siguientes criterios de entrada se analizan para determinar si hay suficientes datos y tecnológica disponible para ejecutar la actividad exitosamente. Los criterios de entrada son:

- La organización ha decidido mover una línea de producción.
- La arquitectura de línea de producción ha sido definida.
- Un conjunto de metas de extracción y objetivos han sido establecidos.
- Uno de tres sistemas heredados ha sido identificado como la fuente de componentes para la extracción.
- Un "equipo de extracción" ha sido establecido.

3.2.9 Artefactos de entrada.

Cada actividad se basa en un conjunto de artefactos de soporte. Los siguientes artefactos se requieren para el establecimiento del contexto de extracción:

- Metas y objetivos.
- Requerimientos de la línea de producción y componentes.
- Documentación de la línea de producción, arquitectura y especificaciones.
- Visión general del sistema heredado y documentación de componentes.
- Reportes de experiencia de otros esfuerzos de extracción/reingeniería.
- Perfil de calendario OAR típico.

La actividad ECE se divide en diez tareas. Esas tareas son ejecutadas en orden. Varias de estas tareas son subdivididas. Para ayudar al equipo OAR a ejecutar las tareas y sub-tareas, el SEI (Software Engineering Institute) desarrollo plantillas de ejecución y datos.

Las tareas logran lo siguiente:

1. Revisión de metas y objetivos.
2. Revisión de Línea de producción / Nuevos requerimientos de sistema.
3. Revisión y selección de componentes necesarios.

3.2.10 Estructura de tareas.

4. Revisión de sistemas heredados y documentación.
5. Determinar controladores de extracción.
6. Validar metas y objetivos.
7. Identificar componentes candidatos.
8. Producción de tópicos de extracción.
9. Evaluar el estado de preparación.
10. Revisión del calendario OAR.

3.2.11 Ejemplo de ejecución y plantillas de datos.

Especifica como ejecutar la revisión de documentación de componentes, la plantilla de datos "EMC-4.2-DT" sugiere que los siguientes tipos de datos deben ser recolectados:

- Lista de componentes heredados.
- Documentación de componentes
- Código fuente de componentes heredados.

3.2.12 Validación.

Algunos de los criterios de validación para el ECE son los siguientes:

- Las expectativas.
- los sistemas de información heredados son identificados.
- Un conjunto de 10 a 20 componentes de línea de producción necesarios con alto potencial han sido seleccionadas para ser satisfechas a través de la extracción.
- Controladores y prioridades son identificados y entendidos.
- Un conjunto de 15 a 30 componentes heredados han sido seleccionados como candidatos para la extracción.
- El nivel de preparación de extracción ha sido evaluado.

3.2.13 Artefactos de salida.

Establecer contextos de extracción produce los siguientes artefactos de salida:

- Un conjunto de relevantes sistemas de información heredados y documentación.
- Un conjunto de componentes de línea de producción necesarios.
- Programas principales y controladores técnicos.
- Una conjunto de componentes heredados y documentación de componentes asociados.
- Evaluación del estado de preparación.
- Lista de tópicos de extracción e inquietudes.
- Revisión calendario OAR.

3.2.14 Criterios de salida.

Antes de completar el establecimiento del contexto de extracción, es necesario cubrir los siguientes criterios de salida:

- El equipo de extracción ha identificado un conjunto razonable de líneas de producción necesarias y componentes candidatos.

- La expectativa de la organización es consistente con los niveles del estado de preparación.
- El perfil del calendario OAR ha sido revisado.
- Todos los artefactos de salida de esta actividad han sido producidos.
- Todas las preguntas fundamentales han sido contestadas satisfactoriamente.

3.2.15 Ejemplos de tareas especializadas.

Si los criterios de salida no son cumplidos, puede ser apropiado una tarea especializada. Ejemplos de tareas especializadas:

- Impulsar los esfuerzos para identificar los controladores de extracción y resolver los problemas programáticos y tópicos técnicos.
- Impulsar la definición de los requerimientos de línea de producción y componentes necesarios en términos de funcionalidad e interfaces.
- Aislado e identificando la funcionalidad e interfaces de los componentes heredados.
- Desarrollando los niveles requeridos de documentación para los componentes heredados.

3.3. METODOLOGÍA DE HERRADURA

Los tres procesos básicos – análisis de un sistema existente, transformación lógica y desarrollo de un nuevo sistema – forman la base del modelo de herradura. La riqueza del modelo de herradura son los tres niveles de abstracción que pueden ser adoptados para las descripciones lógicas.

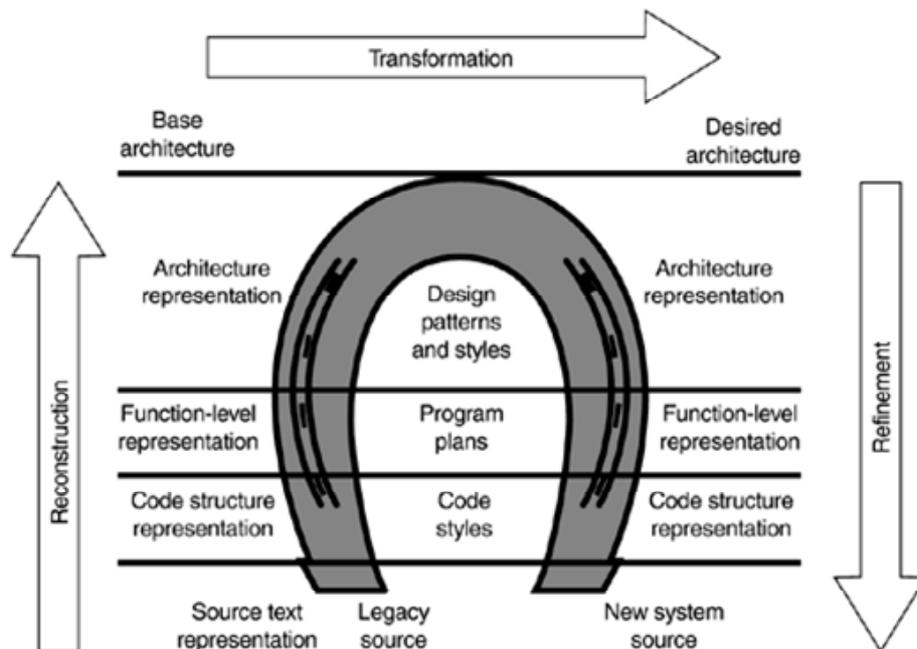


Ilustración 7 Modelo de Herradura

El primer proceso recupera la arquitectura por medio de la extracción de artefactos desde el código fuente. Esta estructura recuperada es analizada para determinar si esta se adapta a la arquitectura antes diseñada.

El segundo proceso es la transformación de arquitectura. En este caso, la arquitectura antes construida es recuperada y es reingenierada para hacerla una nueva arquitectura deseable.

El tercer proceso del modelo de herradura usa el "Architecture-Based Development (ABD)" [Bass99] para ejemplificar la arquitectura deseable. En este proceso, ya empaquetados los tópicos son decididas e interconectadas las estrategias elegidas.

Los Tres Niveles Del Modelo Herradura

En el modelo herradura existen tres niveles:

- "Representación de la estructura de código".
- "Representación del nivel funcional".
- "Nivel conceptual".

3.3.1 Nivel de código.

En el nivel de código existen dos sub-niveles, transformación textual (o basado en cadena) y transformación basado en el árbol de sintaxis. "1A" textual (sintáctico) y "1B" AST-based (semántico).

Transformación Textual:

En el nivel 1A, son realizadas a través de varias comparaciones de cadenas simples y reemplaza métodos. Las transformaciones del nivel 1A son relativamente simples, directas y relativamente baratas. Estas transformaciones son elegidas por las organizaciones cuando el problema es suficientemente simple (quick and dirty).

Transformación al árbol de sintaxis:

En el nivel 1B, las transformaciones a la estructura de código basada en árboles de sintaxis (ASTs) soportan cambios que son inmunes a las variaciones superficiales de sintaxis. La estructura de código basado en árboles de sintaxis es normalmente usada para implementar nuevos lenguajes (COBOL a C++).

3.3.2 Transformaciones a nivel funcional.

Nivel "2" tiene que ver con el re-empaquetado de funcionalidad. La encapsulación de un módulo de funcionalidad por un diferente ambiente, es un ejemplo de transformación a nivel funcional. Transformaciones a nivel funcional va más allá de simples transformaciones a la estructura del código, pero no va más allá que una transformación de arquitectura.

3.3.3 Transformaciones a nivel de arquitectura.

Nivel "3" involucran cambios a los bloques básicos de la arquitectura. El nivel de arquitectura es el más abstracto y lejos del alcance de las transformaciones. Son hechas cuando es necesario un cambio a la estructura principal debido a las principales modificaciones o deficiencias en los sistemas de información heredados.

3.3.4 Interacción entre niveles.

Las transformaciones a la estructura del código se pueden dar sin hacer cambios de nivel funcional o cambios a la arquitectura. Las transformaciones del nivel más bajo soportan las transformaciones de niveles más altos. Sin embargo, las transformaciones de nivel superior no soportan transformaciones de nivel bajos porque esas transformaciones se pueden dar independientemente de las transformaciones de niveles superiores, uno de los principales propósitos del modelo

herradura es elevar el nivel de abstracción y brindar noticiones de la arquitectura para las tareas de Reingeniería.

3.4 METODOLOGÍA CÍCLICA

Este modelo define seis actividades las cuales se muestran en la figura de abajo. En algunas ocasiones, estas actividades se producen de forma secuencial y lineal, pero esto no siempre es así. Por ejemplo, puede ser que la Ingeniería inversa (la comprensión del funcionamiento interno de un programa) tenga que producirse antes de que pueda comenzar la reestructuración de documentos.

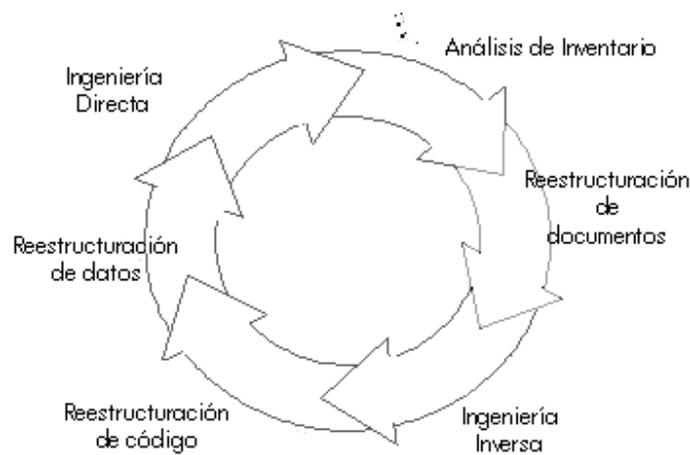


Ilustración 8 Modelo Cíclico de la Reingeniería del Software

El paradigma de la Reingeniería mostrado en la figura es un modelo cíclico. Esto significa que cada una de las actividades presentadas como parte del paradigma puede repetirse en otras ocasiones. Para un ciclo en particular, el proceso puede terminar después de cualquier de estas actividades.

Actividades Del Modelo Cíclico.

Las actividades que se definen en el modelo cíclico: Análisis de inventario, Reestructuración de documentos, Ingeniería inversa, Reestructuración de código, Reestructuración de datos e Ingeniería directa.

3.4.1 Análisis de inventario.

Todas las organizaciones de software deberán disponer de un inventario de todas sus aplicaciones. El inventario puede que no sea más que una hoja de cálculo con la información que proporciona una descripción detallada (por ejemplo: tamaño, edad, importancia para el negocio) de todas las aplicaciones activas, los candidatos a la Reingeniería aparecen cuando se ordena esta información en función de su importancia para el negocio.

Es importante destacar que el inventario deberá revisarse con regularidad

3.4.2 Reestructuración de documentos.

Una documentación escasa es la marca de muchos sistemas de información heredados.

- Opción 1: La creación de documentación consume muchísimo tiempo. El sistema funciona, y ya nos ajustaremos con lo que se tiene.
- Opción 2: Es preciso actualizar la documentación, pero se dispone de recursos limitados. Se utilizará un enfoque "del tipo documentar si se modifica".
- Opción 3: El sistema es fundamental para el negocio, y es preciso volver a documentarlo por completo, un enfoque inteligente consiste en reducir la documentación al mínimo necesario.

3.4.3 Ingeniería inversa.

Una ingeniería inversa con éxito precede de una o más especificaciones de diseño y fabricación para el producto, mediante el examen de ejemplos reales de ese producto. La ingeniería inversa del software es el proceso de análisis de un programa con el fin de crear una representación de programa con un nivel de abstracción más elevado que el código fuente.

3.4.4 Reestructuración del código.

El tipo más común de reingeniería es la reestructuración del código. Para llevar a cabo esta actividad, se analiza el código fuente mediante una herramienta de reestructuración, se indican las violaciones de las estructuras de programación estructurada, y entonces se reestructura el código.

3.4.5 Reestructuración de datos.

Un programa que posea una estructura de datos débil será difícil de adaptar y de mejorar. De hecho, para muchas aplicaciones, la arquitectura de datos tiene más que ver con la viabilidad a largo plazo del programa que el propio código fuente, cuando la estructura de datos es débil, se aplica una Reingeniería a los datos.

Dado que la arquitectura de datos tiene una gran influencia sobre la arquitectura del programa, los cambios en datos darán lugar invariablemente a cambios o bien de arquitectura o bien de código.

3.4.6 Ingeniería directa (forward engineering).

La ingeniería directa, que se denomina también renovación o reclamación, no solamente recupera la información de diseño de un software ya existente, sino que, además, utiliza esta información en un esfuerzo por mejorar su calidad global.

3.5 ANÁLISIS COMPARATIVO DE LAS METODOLOGÍAS

Vamos a realizar el análisis de las metodologías investigadas en base a las características de cada una y a las características o pasos mínimos que debería poseer una metodología para obtener buenos resultados.

METODOLOGÍA	CARACTERÍSTICAS	PASOS
METODOLOGÍA DE REESTRUCTURACIÓN	No modifica la arquitectura global del programa. Se centra más en detalles de diseño. Se utiliza cuando la estructura básica de la aplicación es sólida y solamente un subconjunto de todos los módulos requiere una intensa modificación.	1.- Reestructuración del código 2.- Reestructuración de los datos
METODOLOGÍA DE ANÁLISIS DE OPCIONES PARA REINGENIERÍA (OAR)	Es una metodología sistemática, que permite la identificación y extracción de componentes dentro de grandes y complejos sistemas de software. OAR proporciona un conjunto de opciones de extracción junto con estimación de costos, esfuerzo y riesgos asociados con estas opciones.	1.- Establecimiento del contexto de extracción. 2.- Inventario de componentes. 3.- Analizar componentes candidatos. 4.- Plan de opciones de extracción. 5.- Selección de opciones de extracción.

<p>METODOLOGÍA DE HERRADURA</p>	<p>Basa su metodología en 3 procesos básicos de Reingeniería.</p> <p>En el modelo herradura existen tres niveles:</p> <p>“Representación de la estructura de código”.</p> <p>“Representación del nivel funcional”.</p> <p>“Nivel conceptual”.</p>	<p>1.- Análisis de un sistema existente.</p> <p>2.- Transformación lógica.</p> <p>3.- Desarrollo de un nuevo sistema.</p>
<p>METODOLOGÍA CÍCLICA</p>	<p>En esta metodología cada una de las actividades puede repetirse en otras ocasiones permitiendo que la Reingeniería sea más completa.</p> <p>Abarca varios pasos importantes que permite no dejar ningún cabo suelto y obtener un producto software de calidad mediante la Reingeniería.</p>	<p>1.- Análisis de inventarios.</p> <p>2.- Reestructuración de documentos.</p> <p>3.- Ingeniería inversa.</p> <p>4.- Reestructuración de código.</p> <p>5.- Reestructuración de datos.</p> <p>6.- Ingeniería directa.</p>

Tabla I Análisis comparativo de metodologías

Según algunos autores los pasos mínimos que debe tener un proceso de Reingeniería son los que se ilustran en la siguiente figura:

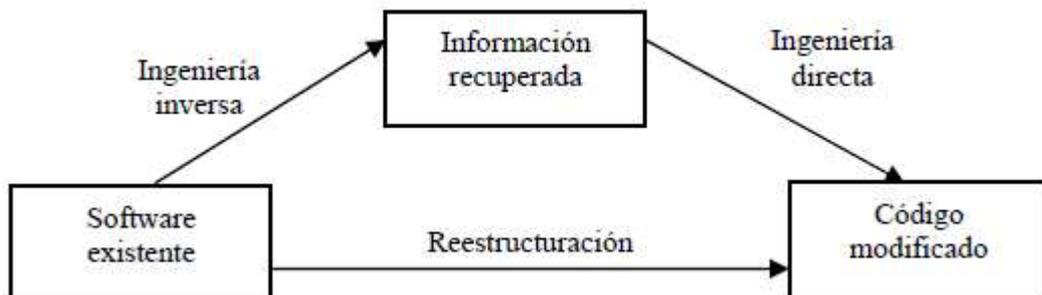


Ilustración 9 Proceso básico de Reingeniería

Software Existente: Este paso consiste en comprender el Software donde el diseño del sistema se recupera desde su código fuente con actividades como análisis de dependencias, comprensión del programa, detección, extracción y almacenamiento del diseño.

Información recuperada: El segundo paso incluye todas las actividades que se realizan para transformar el software existente en un software diferente, más fácil de mantener, entre ellas están: descomposición, reestructuración, remodularización, redocumentación, etc.

Análisis: Según las características y de los requisitos mínimos que debe poseer una buena metodología de Reingeniería de software se llega a la conclusión de que la Metodología Cíclica es la más completa y la que se ajusta a nuestro caso particular, antes de realizar una Reingeniería de un Sistema Informático es necesario analizar la viabilidad de realizar una Reingeniería en base a costos, producción etc, y luego elegir la metodología adecuada para obtener un software de calidad.

3.6 METODOLOGÍA A EMPLEAR PARA EL DESARROLLO DE LA REINGENIERÍA DEL SISTEMA CIPFIE-ESPOCH

Los Sistemas Informáticos que sean sometidos a un proceso de Reingeniería con el fin de mejorar su calidad en las operaciones, la capacidad del sistema, la funcionalidad y el rendimiento del mismo, requieren de una adecuada metodología.

3.6.1 METODOLOGÍA DE DESARROLLO

De acuerdo al estudio realizado de las diferentes metodologías se ha logrado elegir una de ellas que se ajusta a las características de nuestro sistema.

Metodología Cíclica:

1. Análisis de Inventario

2. Reestructuración de documentos
3. Ingeniería Inversa
4. Reestructuración de código
5. Reestructuración de datos
6. Ingeniería hacia Adelante (Directa)

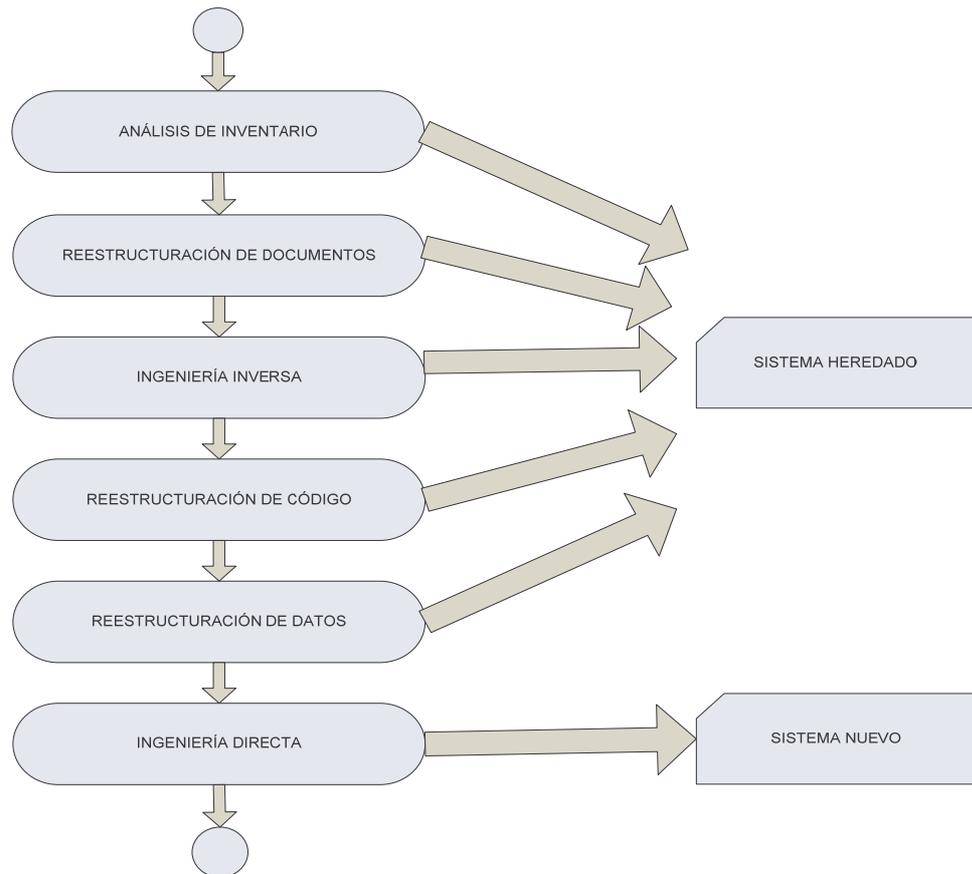


Ilustración 10 Fases de la reingeniería del sistema Sigep

En la ilustración se observa que las fases:

- ✓ Análisis de Inventario
- ✓ Reestructuración de Documentos
- ✓ Ingeniería Inversa
- ✓ Reestructuración de Código
- ✓ Reestructuración de Datos

Están enfocadas en el sistema heredado, con el fin de aprovechar al máximo las cosas buenas del sistema y que sirvan como insumo para la última fase que es la Ingeniería Hacia Adelante del sistema nuevo.

3.6.1.1 Análisis de Inventario

3.6.1.1.1 Datos Generales del Sistema Actual

En esta sección se incluirá la siguiente información:

- ✓ Nombre del Sistema
- ✓ Objetivo general del Sistema
- ✓ Funcionalidad del Sistema
- ✓ Desarrolladores del Sistema
- ✓ Fecha de realización
- ✓ Descripción del Sistema

3.6.1.1.2 Inventario del Sistema Actual

- ✓ Documentación del Sistema
- ✓ Fuentes del Sistema
- ✓ Manuales del Sistema
- ✓ Ejecutables del Sistema
- ✓ Pruebas
- ✓ Scripts de Base de Datos del Sistema
- ✓ Cambios importantes efectuados con fechas y tiempos
- ✓ Bitácora de Errores

3.6.1.1.3 Problemática Actual

Descripción detallada de los problemas que presenta el Sistema Actual

3.6.1.2 Reestructuración de Documentos

Se dispone de 3 opciones para el manejo de la documentación del sistema actual:

OP 1: Se evitará la modificación de la documentación en programas estáticos.

OP 2: En caso de que existan variaciones en el programa, se modificará la documentación solamente en las partes que se han realizado los cambios.

OP 3: Se documentará toda la información en el caso de que el Sistema Informático cambie en su mayor parte.

3.6.1.3 Ingeniería Inversa

En esta etapa se realizará un análisis del Sistema Actual con el propósito de determinar si los datos, diseño, arquitectura y especificaciones son recuperables para el Sistema propuesto.

3.6.1.4 Reestructuración del Código

La reestructuración del código se lleva a cabo para conseguir un diseño que produzca la misma función pero con mayor calidad que el programa original.

3.6.1.5 Reestructuración de Datos

Si la funcionalidad de la Base de Datos del Sistema cambia, es necesario también realizar una reestructuración de la arquitectura del mismo.

3.6.1.6 Ingeniería Hacia Adelante

En esta parte se realizará los cambios necesarios tanto en el diseño como en los módulos con el fin de mejorar el rendimiento del sistema.

3.6.2 METODOLOGÍA DE PROGRAMACIÓN

La metodología de programación que se utilizará es la POO (programación Orientada a Objetos), ya que la lógica y estructura son las más utilizadas hoy en día.

CAPITULO IV

REINGENIERÍA DEL SISTEMA "SiGep"

En este capítulo se desarrollará las diferentes fases de acuerdo a la metodología seleccionada y descrita a breves rasgos en el capítulo anterior.

La información que se describe a continuación fue recopilada de la tesis: *"Integración de la Metodología Ágil XP el estudio de Windows Communication Foundation como Alternativa Metodológica para el desarrollo de Software orientado a Servicios caso practico CIPFIE-ESPOCH"*, cuyos autores son: Luis Alberto Rivera Riera y Juan Pablo Tixe Alulema.

4.1 ANÁLISIS DE INVENTARIO

4.1.1 Datos Generales del Sistema Actual

Nombre del Sistema	"SiGep"
Objetivo del Sistema	Planificar, ejecutar, asesorar, coordinar y controlar desarrollo de proyectos de investigación, tesis, memorias y prácticas de pregrado, regido en el reglamento interno del Centro de Investigación de la Facultad.

<p>Funcionalidad del Sistema</p>	<ul style="list-style-type: none"> ✓ Permitirá realizar acciones como el registro de un Proyecto ✓ Propuesto, Asignación de Tribunal Evaluador, etc.; sobre ✓ Proyectos que se encuentren en esta Etapa. ✓ Permitirá realizar acciones relacionadas al registro de ✓ Matricula, Avances; así como la Asignación de Delegados para ✓ Defensa Final y el registro de su finalización de Proyectos. ✓ Facilitará la impresión de documentos necesarios para el flujo del proceso relacionado a los Proyectos de Tesis. ✓ Presenta reportes estadísticos a nivel gerencial con el fin de ✓ apoyar en la toma de decisiones. ✓ Proporciona el acceso a consultas generales de Proyectos de ✓ Tesis; presentando información dinámica en dependencia de ✓ su etapa. <p>Permite administrar los parámetros y catálogos del Sistema.</p>
<p>Desarrolladores del Sistema</p>	<p>Luis Alberto Rivera Riera Juan Pablo Tixe Alulema</p>
<p>Fecha de realización</p>	<p>Año 2009</p>
<p>Descripción del Sistema</p>	<p>SiGep pretende automatizar el proceso correspondiente a la gestión de Tesis pertenecientes a la Facultad de Informática y Electrónica; así mismo estará integrado al Sistema Académico de la Escuela Superior Politécnica de Chimborazo con el fin de utilizar información relacionada a Docentes y Estudiantes.</p>

Tabla II Datos generales sistema Sigep actual

4.1.2 Inventario del Sistema Actual

4.1.2.1 Documentación del Sistema

A continuación se lista la documentación recuperada del sistema anterior:

- ✓ Requerimientos Funcionales del Sistema

- ✓ Diagrama de Flujo de Datos
- ✓ Diagrama de Actividades
- ✓ Diagramas de Iteración
- ✓ Modelo de Interfaces de Usuario
- ✓ Descripción de Clases

4.1.2.2 Fuentes del Sistema

Los archivos fuente del Sistema "SiGep", se encuentran en poder del Departamento de **DESITEL** de la ESPOCH.

4.1.2.3 Manuales del Sistema

- ✓ Manual del Sistema
Páginas 120- 205 de la Tesis mencionada.
- ✓ Manual de Usuario

4.1.2.4 Ejecutables del Sistema

Estos se encuentran en poder del Departamento de **DESITEL** de la ESPOCH.

4.1.2.5 Pruebas

No se encuentra ningún plan de pruebas para el sistema "SiGep".

4.1.2.6 Scripts de Base de Datos del Sistema

En el Departamento de DESITEL se encuentra el script de la Base de Datos del Sistema Actual.

4.1.2.7 Cambios importantes efectuados con fechas y tiempos

No se encuentran documentos en los cuales conste los cambios realizados en el Sistema, sin embargo en forma verbal nos informaron que realizaron unos pequeños cambios en el Sistema para su mejor funcionamiento.

4.1.2.8 Bitácora de Errores

No existe registro de los errores producidos durante la utilización del sistema actual.

4.1.3 Problemática Actual

4.1.3.1 Evaluación del Sistema

✓ **Ambiente Externo**

- Hardware
- Software

✓ **Ambiente Interno**

- Sistema

Ambiente Externo Hardware:

INDICADOR	PARÁMETRO	CALIFICACIÓN
Fabricante/Proveedor	1.No existe	
	2. No existe pero es soportado	
	3. Existe, poco soporte	X
	4. Existe, poco soporte	
Edad	1.Mayores a 6 años	
	2.Entre 4 y 6 años	X
	3.Entre 2 y 4 años	
	4.Menores de 2 años	
Costo de mantenimiento	1.Costo alto	X
	2.Costo medianamente alto	
	3.Costo medianamente bajo	
	4.Costo bajo	
Errores	1.Frecuentes	
	2.A menudo	
	3.Casuales	
	4.Ocasionales	X
Requerimiento soporte técnico	1.Alto	
	2.Medio Alto	
	3.Medio bajo	X
	4.Necesita muy poco	
Desempeño	1.Bajo desempeño	
	1.Medio bajo	
	1.Medio alto	
	1.Buen desempeño	X

Tabla III Ambiente externo Hardware sistema Sigep actual

La evaluación obtenida es de 14, que representa el 58,33%, esto indica si tenemos que no tenemos que cambiar el HW.

Ambiente Externo Software:

El sistema "SiGep", actualmente se encuentra instalado sobre un servidor con Sistema Operativo Linux Centos.

INDICADOR	PARÁMETRO	CALIFICACIÓN
Fabricante/Proveedor	1.No existe	
	2. No existe pero es soportado	
	3. Existe, poco soporte	X
	4. Existe, poco soporte	
Edad	1.Mayores a 6 años	
	2.Entre 4 y 6 años	
	3.Entre 2 y 4 años	
	4.Menores de 2 años	
Costo Licencia	1.Costo bajo	
	2.Costo medio bajo	
	3.Costo medio alto	
	4.Costo alto	X
Frecuencia de arreglos Parches	1.Frecuentes	X
	2.A veces	
	3.Rara vez	
	4.Casi nunca	
Errores	1.Frecuentes	X
	2.A menudo	
	3.Casuales	
	4.Ocasionales	
Desempeño	1.Bajo desempeño	X
	2.Medio bajo	
	3.Medio alto	
	4.Buen desempeño	
Costo de mantenimiento	1.Costo alto	X
	2.Costo medio alto	
	3.Costo medio bajo	
	4.Costo bajo	
Calidad-soporte técnico	1.No necesita soporte	
	2.Muy poco	
	3.Ocasionalmente	X
	4.Soporte con equipo completo	

Tabla IV Ambiente externo Software sistema Sigep actual

Ambiente Interno Sistema:

Las características del Sistema son:

INDICADOR	PARÁMETRO	CALIFICACIÓN
Tamaño(puntos de función)	1.Mayores de 500	
	2. Entre 500 y 400	
	3. Entre 300 y 400	X
	4. Menor de 300	
Complejidad	1.Muy complejo	X
	2.Alta	
	3.Media	
	4.Baja	
Documentación	1.Nada	
	2.Poco	
	3.Casi todo	X
	4.Todo	
Datos	1.Archivos planos	
	2.Archivos manuales	X
	3.Base de Datos Relacional	
	4.Base de Datos O.O	X
Ambiente de pruebas	1.no existe	X
	2.Relativamente bajo	
	3.Relativamente alto	
	4.buen ambiente	
Interfaces de Usuario	1.Malas	
	2.Confusas	X
	3.Amigables	
	4.Buenas(Distribuídas)	
Registro de mantenimiento	1.Nada	X
	2.Poco	
	3.Alto	
	4.Excesivo	
Soporte	1.Regular	X
	2.Bueno	
	3.Muy bueno	
	4.Excelente	
Seguridad	1.Baja	
	2.Media baja	
	3.Media alta	X
	4.Alta	
Vínculos con otros sistemas	1.No existe relación	
	2.Poca relación con otros SI	
	3.Alta relación con unos SI	X
	4.Se relaciona con cualquier SI	

Tabla V **Ambiente interno sistema Sigep actual**

4.2 REESTRUCTURACIÓN DE DOCUMENTOS

Una vez revisadas y analizada la documentación del Sistema Actual, se optó por cambiar toda la documentación existente debido a que el sistema actual realiza sólo consumo de Servicios y es una aplicación de Escritorio, en cambio el sistema propuesto es una Página Web

La documentación que se va a sustituir entre otras las principales són:

- ✓ Modelos de Casos de Uso
- ✓ Arquitectura del Sistema
- ✓ Manual de Sistema
- ✓ Manual de Usuario

4.3 INGENIERÍA INVERSA

Esta Ingeniería se aplicó en lo siguiente:

- ✓ A nivel de código se realizó una ingeniería inversa total ya que la Programación del Sistema propuesto se lo realizó en un Lenguaje diferente.
- ✓ A nivel de diseño, especificaciones y arquitectura se recuperó cierta parte como la Base de Datos y el Diseño para orientarlo a la Web.
- ✓ A nivel de datos se migró los datos existentes en una Hoja de Cálculo de Excel a la nueva Base de Datos del Sistema.

4.4 REESTRUCTURACIÓN DE DATOS

La funcionalidad del sistema varía por tanto la reingeniería de datos se modificaría con el aumento o supresión de algunos campos.

CAPITULO V

INGENIERIA HACIA ADELANTE

5.1 INTRODUCCIÓN

En este capítulo con la fase siguiente que es la ingeniería progresiva o ingeniería hacia adelante, en ésta fase se tomará como referencia cierta información de la tesis anterior, ésta información servirá para reconstruir el sistema existente con el fin de lograr su máximo potencial, cabe mencionar que la documentación en su mayoría si se va a desarrollar ya que el sistema anterior era un sistema sobre una plataforma de escritorio, en cambio el sistema propuesto está sobre plataforma Web.

5.2 CAPTURA DE REQUERIMIENTOS

Mediante el análisis del documento del sistema existente, se ha logrado recopilar un listado de requerimientos funcionales y no funcionales, los mismos que fueron revisados, analizados y actualizados en caso de requerirse, de acuerdo a esto los requerimientos son los siguientes:

5.2.1 REQUERIMIENTOS FUNCIONALES

Cabe recalcar que se revisó la documentación existente sobre los Requerimientos, y se encontró que se tiene Requerimientos Funcionales, los mismos que han sido revisados y los que se detalla a continuación:

R1: ANTEPROYECTOS DE TESIS		
Crear Proyecto Propuesto		
ENTRADAS	PROCESOS	SALIDAS
Nombre del Proyecto. (R) Escuela. (R) Carrera. (R) Área. (R) Lugar de Realización. (R) Director del Proyecto. (R) Estudiantes proponentes (R)	Verificar campos requeridos. Validar la no existencia de un proyecto con el mismo nombre. Enviar a guardar los datos.	Mensaje de resultado de proceso.
Asignar Tribunal Evaluador		
Proyecto Seleccionado (R) Fecha de Evaluación. (R) Fecha de Entrega de Actas. (R) Docentes pertenecientes al Tribunal. (R)	Validar fecha entrega de Actas no menor a la fecha Evaluación. Validar la existencia de Docentes en Tribunal asignado. Enviar a guardar datos.	Mensaje de resultado de proceso.
Registrar Informe Evaluación		
Proyecto Seleccionado. (R) Fecha de Evaluación. (R) Calificación. (R) Observaciones. (O) Nuevo Tema. (O)	Verificar existencia de datos requeridos. Validar fecha de Evaluación no menor a fecha de asignación. Enviar a guardar los datos.	Mensaje de resultado de proceso.
Registrar Asesor Anteproyecto		
Proyecto Seleccionado. (R) Número Resolución. (R) Fecha Resolución. (R) Identificador Asesor. (O)	Verificar existencia de datos requeridos. Validar la no participación	Mensaje de resultado de proceso.

Nombre Asesor. (O)	del Asesor asignado en el proyecto con otros roles.	
	Enviar a guardar los datos.	

Tabla VI Requerimiento 1 "Anteproyectos de Tesis"

R2: PROYECTOS DE TESIS		
Matricular Proyecto		
ENTRADAS	PROCESOS	SALIDAS
Proyecto seleccionado. (R) Número Comprobante. (R) Fecha Resolución. (R) Fecha Fin Matricula. (R) Observaciones. (O)	Verificar tipo de datos correctos. Validar fecha fin no menor a fecha resolución. Enviar a guardar datos.	Mensaje de resultado de proceso.
Asignar Delegado Defensa		
Proyecto seleccionado. (R) Fecha de Defensa asignado. (R) Fecha Entrega Actas calificación. (R) Número Resolución. (R) Fecha Resolución. (R) Identificador Delegado. (R) Nombre Delegado. (R)	Verificar existencia de campos requeridos. Validar consistencia en fechas asignadas. Enviar a guardar datos.	Mensaje de resultado de proceso.
Registrar Información Final		
Proyecto seleccionado. (R) Código de Tesis Finalizada. (R) Fecha Finalización. (R) Resumen. (R) Palabras Claves. (R) Calificación Proyecto. (R)	Verificar existencia de campos requeridos. Validar fecha finalización no mayor a la actual. Validar calificación no mayor a 20.	Mensaje de resultado de proceso.

Tabla VII Requerimiento 2 "Proyecto de Tesis"

R3: DOCUMENTOS
Acta Evaluación Propuesto

ENTRADAS	PROCESOS	SALIDAS
Proyecto seleccionado. (R)	Buscar información relacionada al proyecto seleccionado. Cargar datos del Proyecto en respectivo Reporte.	Reporte cargado.
Acta Evaluación Escrita Tesis		
Proyecto Seleccionado. (R)	Buscar información relacionada al proyecto seleccionado. Cargar datos del Proyecto en respectivo Reporte.	Reporte cargado.
Acta Evaluación Oral		
Proyecto seleccionado. (R)	Buscar información relacionada al proyecto seleccionado. Cargar datos del Proyecto en respectivo Reporte.	Reporte cargado.
Certificado Tesis Dirigidas por Docente		
Identificador del Docente. (R)	Buscar tesis finalizadas pertenecientes al Docente. Imprimir certificado.	Certificado impreso.

Tabla VIII **Requerimiento 3 "Documentos"**

R4: CONSULTAS		
Listado Propuestos Sin Evaluación		
ENTRADAS	PROCESOS	SALIDAS
Proyecto seleccionado. (R)	Buscar Proyectos Propuestos que han sido asignados un Tribunal Evaluador y aun no han sido registrado su factibilidad. Presentar listado de Proyectos encontrados.	Listado de Proyectos encontrados
Listado de Tesis Vigentes		
	Buscar Proyectos en Etapa Tesis que aun no han sido finalizadas.	Listado de Proyectos encontrados.

	Presentar listado de Proyectos encontrados.	
Búsqueda General Proyectos		
Etapa. (O) Estado Tesis. (O) Nombre Proyecto. (O) Identificación Estudiante. (O) Nombre Estudiante. (O) Lugar Realización. (O)	Buscar Proyectos que coincidan con los criterios de búsqueda ingresados. Presentar listado de Proyectos encontrados.	Listado de Proyectos encontrados.

Tabla IX Requerimiento 4 "Consultas"

R5: REPORTES		
Proyectos Por Empresas		
ENTRADAS	PROCESOS	SALIDAS
Identificador de Empresa. (R) Fecha Inicio. (R) Fecha Fin. (R)	Buscar Proyectos pertenecientes a la Empresa. Generar Reporte Estadístico con la información encontrada. Presentar Reporte Estadístico.	Reporte Estadístico.
Proyectos Por Escuelas		
Identificador de la Escuela. (R) Fecha Inicio. (R) Fecha Fin. (R)	Buscar proyectos pertenecientes a la Escuela. Generar Reporte Estadístico con la información encontrada. Presentar Reporte Estadístico.	Reporte Estadístico.
Proyectos Por Director		
Identificador del Docente. (R) Fecha Inicio. (R) Fecha Fin. (R)	Buscar proyectos pertenecientes al Docente. Generar Reporte Estadístico con la información encontrada. Presentar Reporte Estadístico.	Reporte Estadístico.

Tabla X Requerimiento 5 "Reportes"

5.2.2 ACTORES

Los actores según los requerimientos son:

- **Administrador:** Es la persona encargada de ingresar información para el buen desempeño del sistema, a la vez que controla y brinda mantenimiento al mismo.
- **Secretaria CIPFIE:** Es la persona encargada de registrar toda la información de proyectos y anteproyectos de tesis, así también proporcionar información de la misma.
- **Estudiantes:** Son quienes solicitan información sobre los proyectos de investigación realizados.

5.2.3 MODELO DE CASOS DE USO

Ingresar Anteproyecto

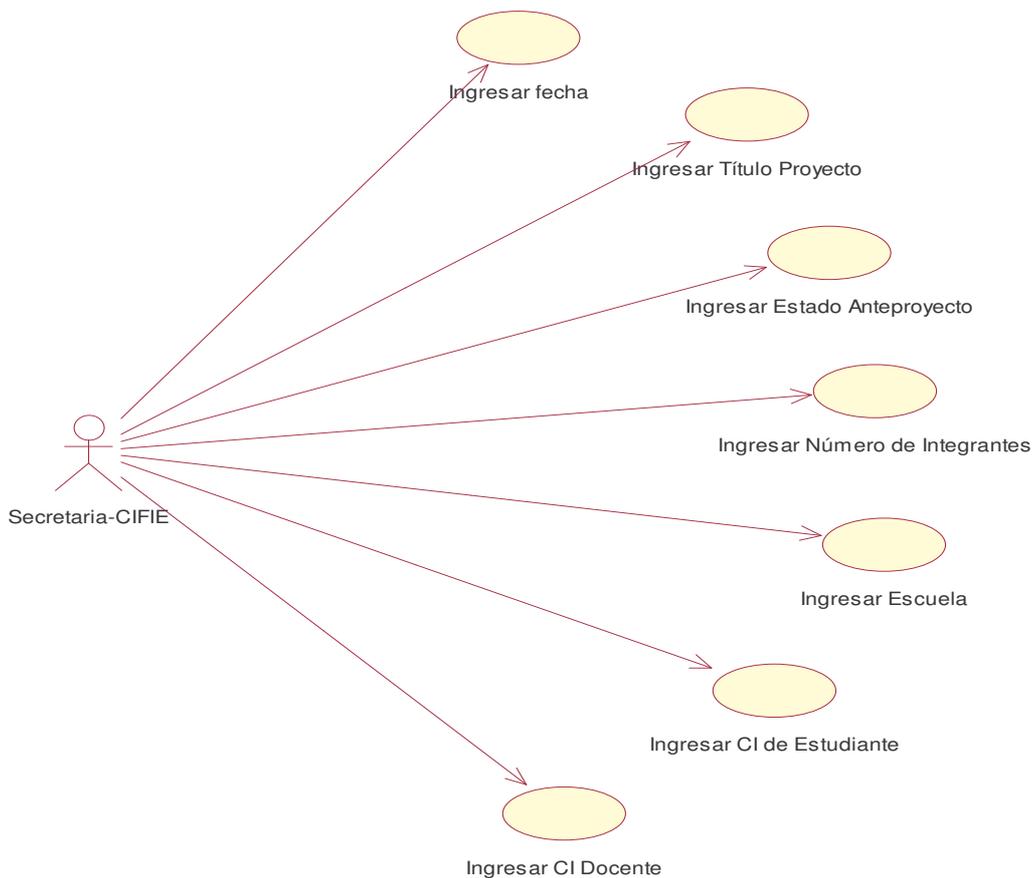


Ilustración 11 Casos de uso de Secretaria-CIFIE Ingresar Anteproyecto

Registrar la Matrícula del Proyecto

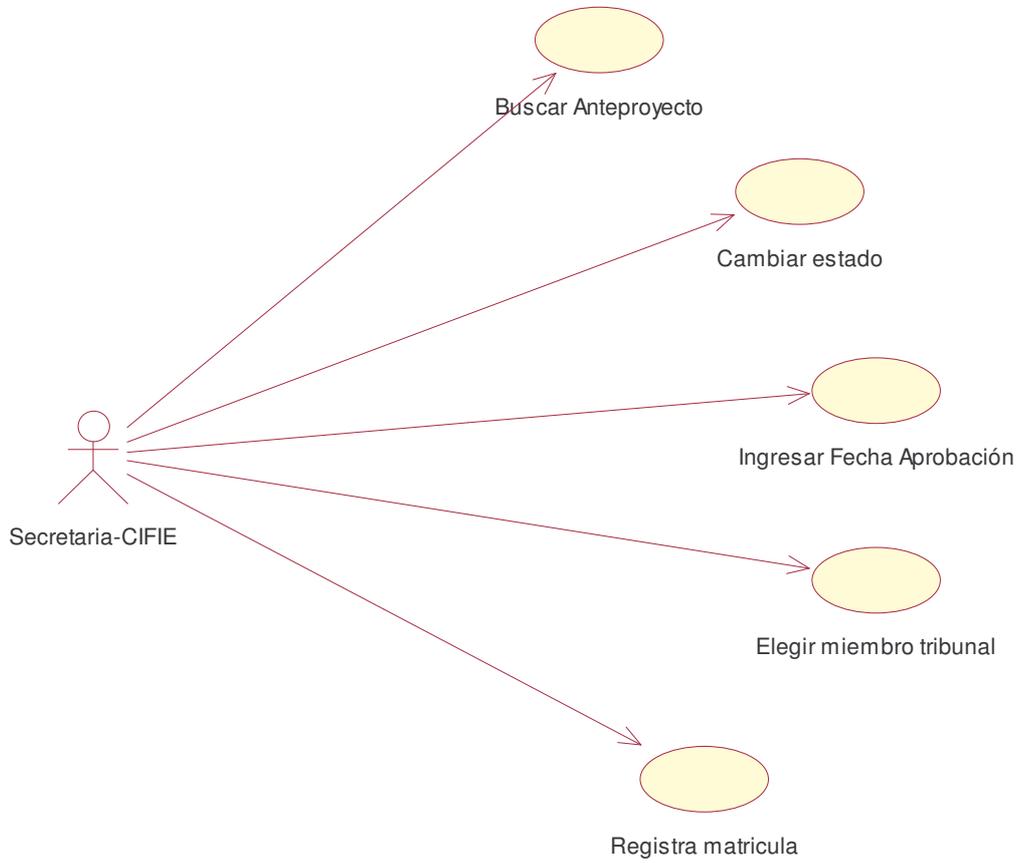


Ilustración 12 Casos de uso de **Secretaria-CIFIE Registrar Matrícula Proyecto**

Buscar Proyecto

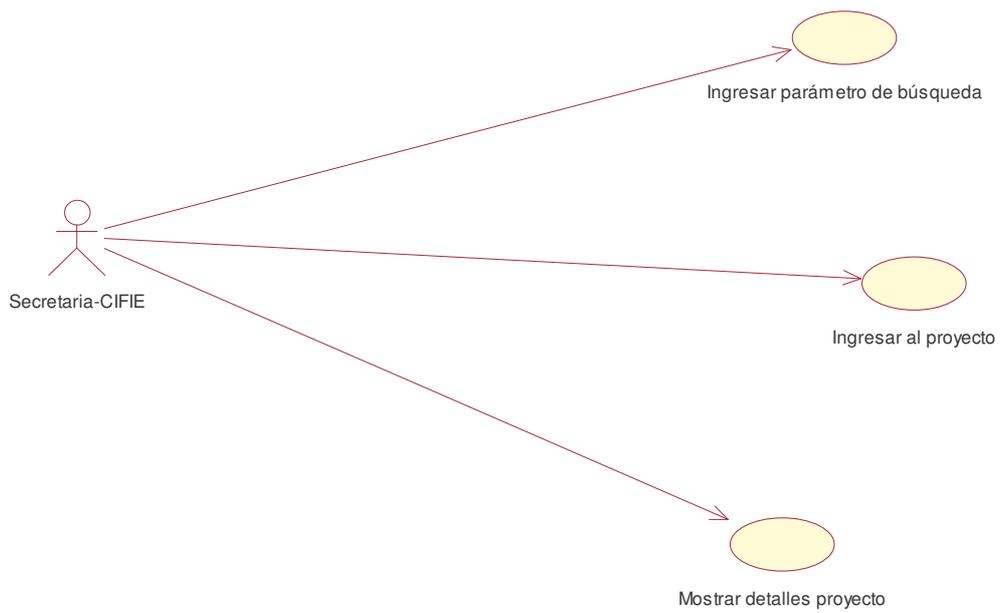


Ilustración 13 Casos de uso de **Secretaria-CIFIE** Buscar

Imprimir Actas

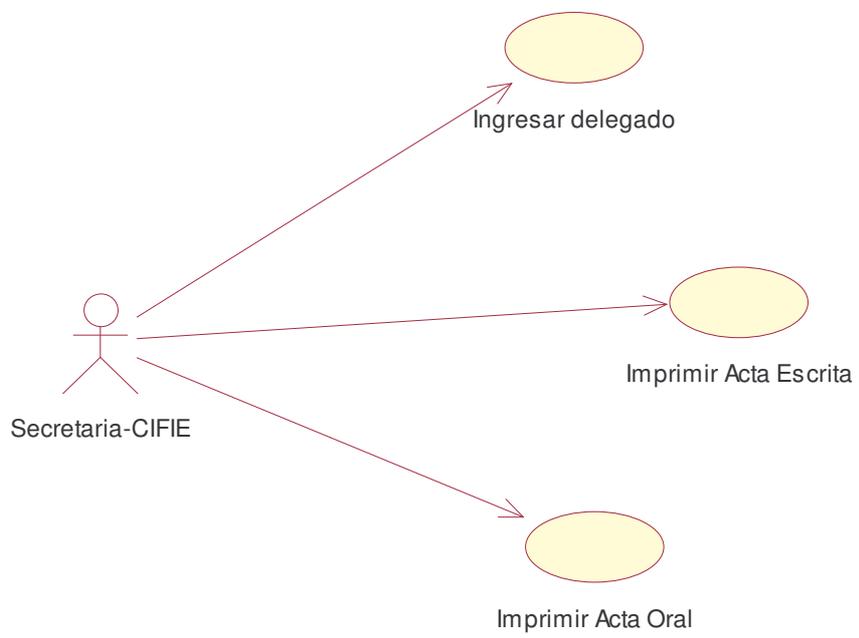


Ilustración 14 Casos de uso de Secretaria-CIFIE Imprimir Actas

Mostrar Listados

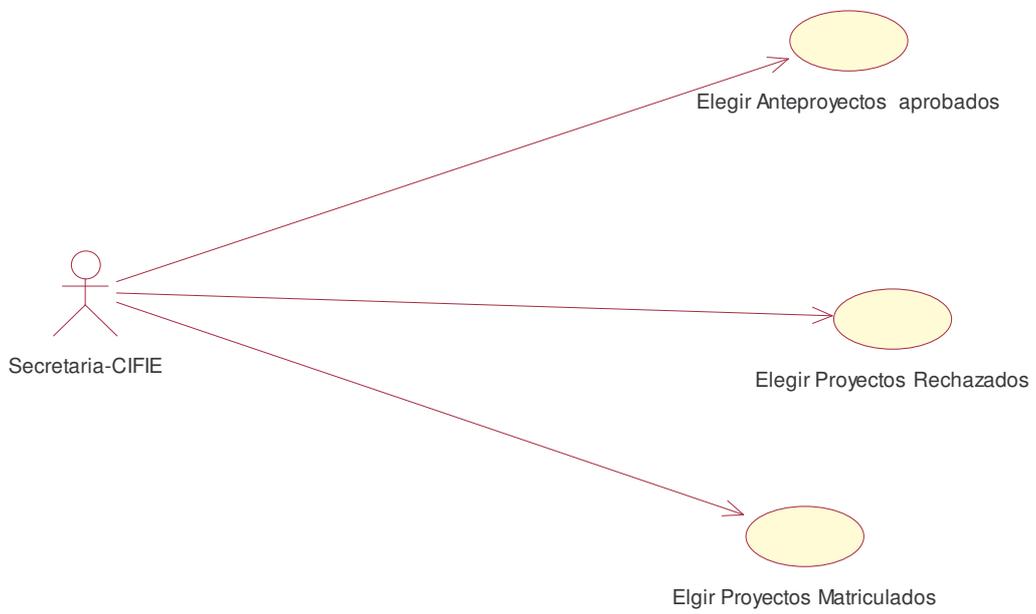


Ilustración 15 Casos de uso de Secretaria-CIFIE Mostrar Listados

Ingresar Proyecto de Investigación

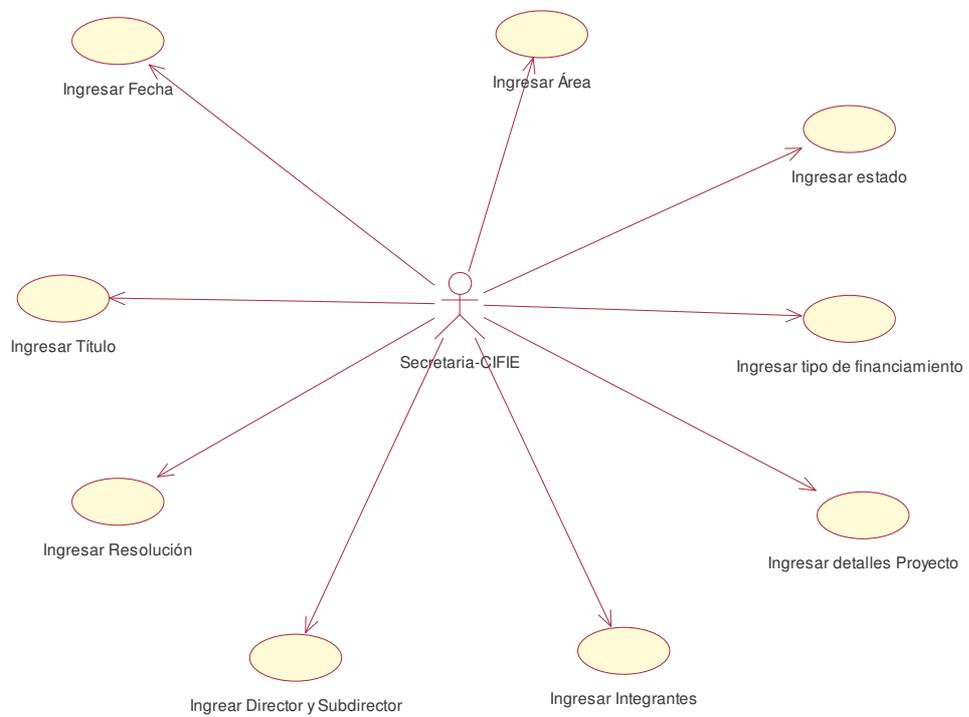


Ilustración 16 Casos de uso de Secretaria-CIFIE Ingresar Proyecto de Investigación

Ingresar Evento

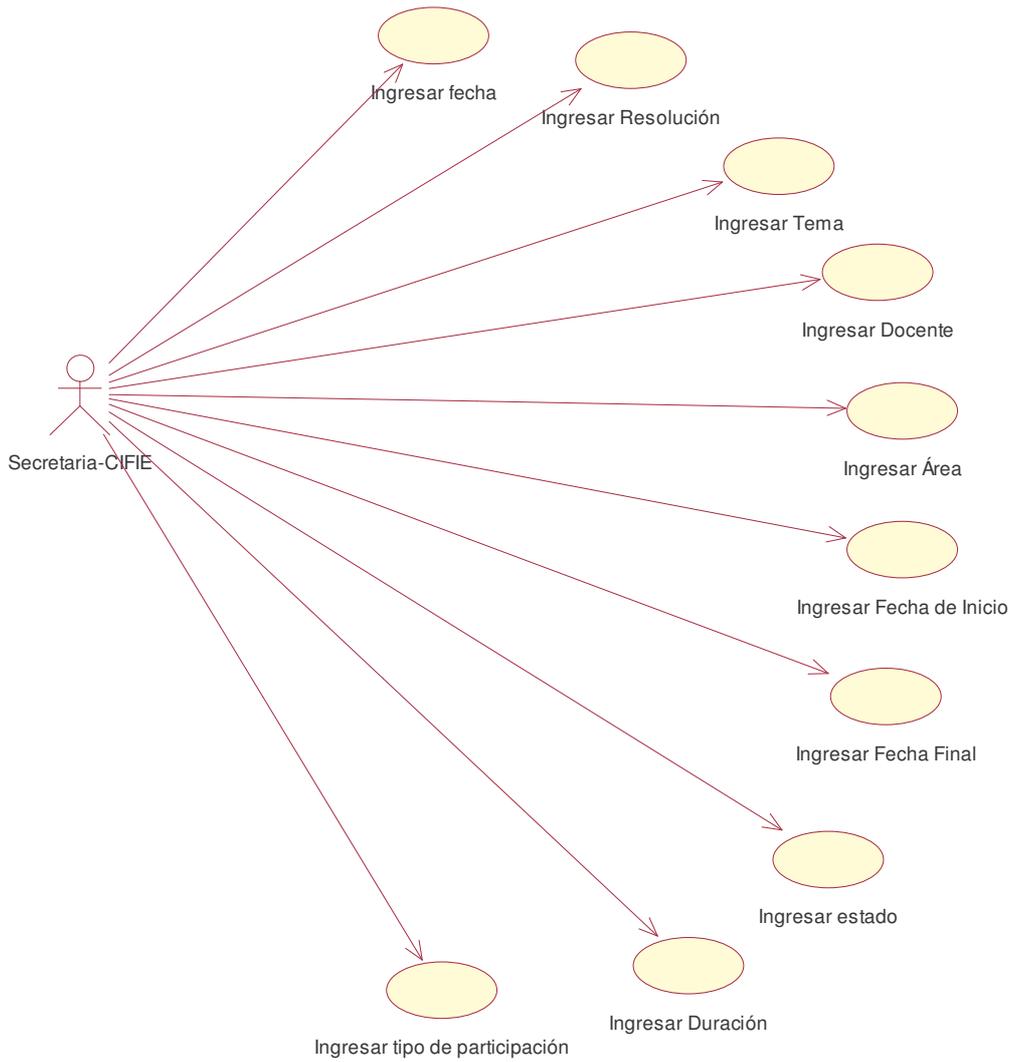


Ilustración 17 Casos de uso de Secretaria-CIFIE Ingresar Evento

Registrar Publicaciones

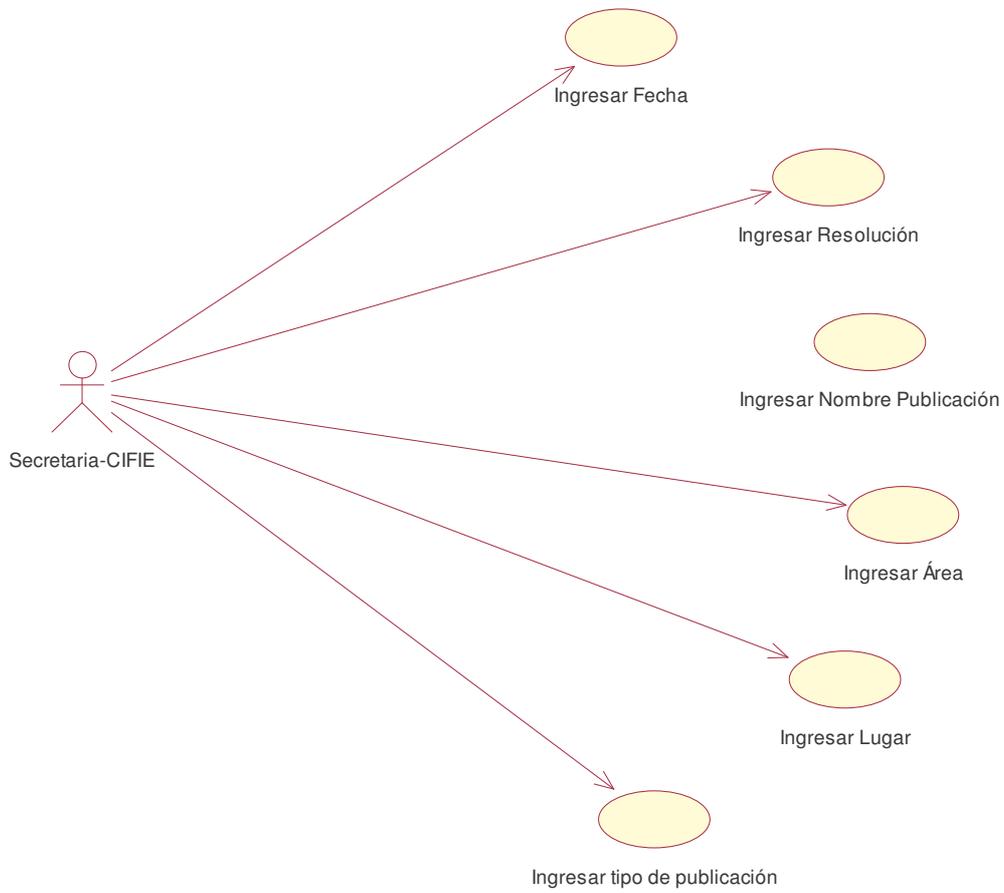


Ilustración 18 Casos de uso de Secretaria-CIFIE Ingresar Publicaciones

5.2.4 ESPECIFICACIÓN DE CASOS DE USO

5.2.4.1 Validar Usuario

CASO DE USO:		VALIDAR USUARIO	
CÓDIGO:	CU01	REFERENCIA REQUERIMIENTO:	
		REFERENCIA CASO DE USO:	
DESCRIPCIÓN GENERAL: Permitirá verificar si se encuentra registrado el usuario en el sistema.			
ACTORES: Administrador			
PRE-CONDICIONES: El Usuario deberá estar registrado en el Sistema.			
POS-CONDICIONES:			
CURSO NORMAL			
PASO	ACTOR(ES)	SISTEMA	
CN1.1		Muestra una pantalla de validación	
CN1.2	Ingresa login y password		
CN1.3		Valida Datos, muestra pantalla de acuerdo al usuario.	
CURSO DE ERROR			
PASO	ACTOR(ES)	SISTEMA	
CE1.1	Ingresa login y password		
CE1.2		Usuario No registrado, muestra Pantalla de inicio.	

Tabla XI Especificación de caso de uso validar usuario

5.2.4.2. Ingresar Anteproyecto

CASO DE USO:		INGRESAR ANTEPROYECTO	
CÓDIGO:	CU02	REFERENCIA REQUERIMIENTO:	
		REFERENCIA CASO DE USO:	
DESCRIPCIÓN GENERAL: Permitirá ingresar los Datos sobre el Anteproyecto que los estudiantes proponentes presenten en Secretaría.			
ACTORES: Secretaria-CIFIE			
PRE-CONDICIONES: El Usuario deberá estar registrado en el Sistema.			
POS-CONDICIONES:			
CURSO NORMAL			
PASO	ACTOR(ES)	SISTEMA	
CN1.1		Muestra una pantalla de validación	
CN1.2	Ingresar login y password		
CN1.3		Valida Usuario	
CN1.4	Llena formulario de ingreso de Anteproyecto		
CN1.5		Valida Datos ingresados	
CN1.6	Acepta datos ingresados		
CN1.7		Guarda información	

Tabla XII Especificación de caso de uso Asignar Anteproyecto

5.2.4.3. Registrar Matrícula

CASO DE USO:		REGISTRAR MATRÍCULA	
CÓDIGO:	CU03	REFERENCIA REQUERIMIENTO:	
		REFERENCIA CASO DE USO:	
DESCRIPCIÓN GENERAL: Permitirá registrar la matrícula de un Anteproyecto Aprobado.			
ACTORES: Secretaria-CIFIE			
PRE-CONDICIONES: El Usuario deberá estar registrado en el Sistema.			
POS-CONDICIONES:			
CURSO NORMAL			
PASO	ACTOR(ES)	SISTEMA	
CN1.1		Muestra una pantalla de validación	
CN1.2	Ingresa login y password		
CN1.3		Valida Usuario	
CN1.4	Buscar Anteproyecto		
CN1.5		Muestra pantalla con datos del Anteproyecto	
CN1.6	Ingresa datos para registrar matrícula		
CN1.7		Valida datos	
CN1.8	Acepta datos de matrícula		
CN1.9		Guarda información	

Tabla XIII Especificación de caso de uso ingresar matrícula

5.2.4.4. Buscar

CASO DE USO:		BUSCAR	
CÓDIGO:	CU04	REFERENCIA REQUERIMIENTO:	
		REFERENCIA CASO DE USO:	
DESCRIPCIÓN GENERAL: Permitirá a la Secretaria-CIFIE buscar y mostrar información dado un parámetro.			
ACTORES: Secretaria-CIFIE			
PRE-CONDICIONES: El Usuario deberá estar registrado en el Sistema.			
POS-CONDICIONES:			
CURSO NORMAL			
PASO	ACTOR(ES)	SISTEMA	
CN1.1		Muestra una pantalla de validación	
CN1.2	Ingresa login y password		
CN1.3		Valida Usuario	
CN1.4	Ingresa parámetro de búsqueda		
CN1.5		Muestra pantalla con datos del proyecto	

Tabla XIV Especificación de caso de uso listar tesis vigentes

5.2.4.5. Imprimir Actas

CASO DE USO:		IMPRIMIR ACTAS	
CÓDIGO:	CU05	REFERENCIA REQUERIMIENTO:	
		REFERENCIA CASO DE USO:	
DESCRIPCIÓN GENERAL: Permitirá a la Secretaria-CIFIE imprimir ya sea el Acta Oral o Escrita de acuerdo al requerimiento.			

ACTORES: Secretaria-CIFIE		
PRE-CONDICIONES: El Usuario deberá estar registrado en el Sistema.		
POS-CONDICIONES:		
CURSO NORMAL		
PASO	ACTOR(ES)	SISTEMA
CN1.1		Muestra una pantalla de validación
CN1.2	Ingresa login y password	
CN1.3		Valida Usuario
CN1.4	Ingresa delegado	
CN1.5		Valida datos
CN1.6	Solicita impresión	
CN1.7		Muestra pantalla para enviar a Imprimir
CN1.8	Manda a imprimir	
CN1.9		Imprime acta solicitada

Tabla XV Especificación de caso de uso listar tesis vigentes

5.2.4.6. Mostrar Listados

CASO DE USO:	MOSTRAR LISTADOS		
CÓDIGO:	CU06	REFERENCIA REQUERIMIENTO:	
		REFERENCIA CASO DE USO:	
DESCRIPCIÓN GENERAL: Permitirá a la Secretaria-CIFIE mostrar los listados de proyectos o anteproyectos de acuerdo al estado en el que se encuentren (en proceso, aprobado o rechazado).			
ACTORES: Secretaria-CIFIE			
PRE-CONDICIONES: El Usuario deberá estar registrado en el Sistema.			

POS-CONDICIONES:		
CURSO NORMAL		
PASO	ACTOR(ES)	SISTEMA
CN1.1		Muestra una pantalla de validación
CN1.2	Ingresa login y password	
CN1.3		Valida Usuario
CN1.4	Ingresa parámetro para mostrar	
CN1.5		Valida y muestra listado.

Tabla XVI Especificación de caso de uso mostrar listados

5.2.4.7. Ingresar Proyecto de Investigación

CASO DE USO:		INGRESAR PROYECTO DE INVESTIGACIÓN	
CÓDIGO:	CU07	REFERENCIA REQUERIMIENTO:	
		REFERENCIA CASO DE USO:	
DESCRIPCIÓN GENERAL: Permitirá a la Secretaria-CIFIE ingresar y registrar un proyecto de investigación presentado por un Docente o Estudiante.			
ACTORES: Secretaria-CIFIE			
PRE-CONDICIONES: El Usuario deberá estar registrado en el Sistema.			
POS-CONDICIONES:			
CURSO NORMAL			
PASO	ACTOR(ES)	SISTEMA	
CN1.1		Muestra una pantalla de validación	
CN1.2	Ingresa login y password		
CN1.3		Valida Usuario	
CN1.4	Llena el formulario de		

	registro del proyecto de investigación	
CN1.5		Valida datos
CN1.6	Acepta datos ingresados	
CN1.7		Guarda información

Tabla XVII Especificación de caso de uso ingresar proyecto de investigación

5.2.4.8. Ingresar Evento Científico

CASO DE USO:		INGRESAR EVENTO CIENTÍFICO	
CÓDIGO:	CU08	REFERENCIA REQUERIMIENTO:	
		REFERENCIA CASO DE USO:	
DESCRIPCIÓN GENERAL: Permitirá a la Secretaria-CIFIE ingresar y registrar la participación de los Docentes en un evento académico científico.			
ACTORES: Secretaria-CIFIE			
PRE-CONDICIONES: El Usuario deberá estar registrado en el Sistema.			
POS-CONDICIONES:			
CURSO NORMAL			
PASO	ACTOR(ES)	SISTEMA	
CN1.1		Muestra una pantalla de validación	
CN1.2	Ingresar login y password		
CN1.3		Valida Usuario	
CN1.4	Llena el formulario de registro de participación en el evento académico científico.		
CN1.5		Valida datos	
CN1.6	Acepta datos ingresados		

CN1.7		Guarda información
-------	--	--------------------

Tabla XVIII Especificación de caso de uso ingresar evento científico

5.2.4.9. Ingresar Publicaciones

CASO DE USO:		INGRESAR PUBLICACIONES	
CÓDIGO:	CU09	REFERENCIA REQUERIMIENTO:	
		REFERENCIA CASO DE USO:	
DESCRIPCIÓN GENERAL: Permitirá a la Secretaria-CIFIE ingresar y registrar una publicación (libro, texto básico, artículo científico).			
ACTORES: Secretaria-CIFIE			
PRE-CONDICIONES: El Usuario deberá estar registrado en el Sistema.			
POS-CONDICIONES:			
CURSO NORMAL			
PASO	ACTOR(ES)	SISTEMA	
CN1.1		Muestra una pantalla de validación	
CN1.2	Ingresar login y password		
CN1.3		Valida Usuario	
CN1.4	Llena el formulario de registro de publicación.		
CN1.5		Valida datos	
CN1.6	Acepta datos ingresados		
CN1.7		Guarda información	

Tabla XIX Especificación de caso de uso ingresar publicaciones

5.3 ANÁLISIS

5.3.1. MODELO DE OBJETOS

5.3.1.1. Modelo de Objetos para Ingresar Proyecto

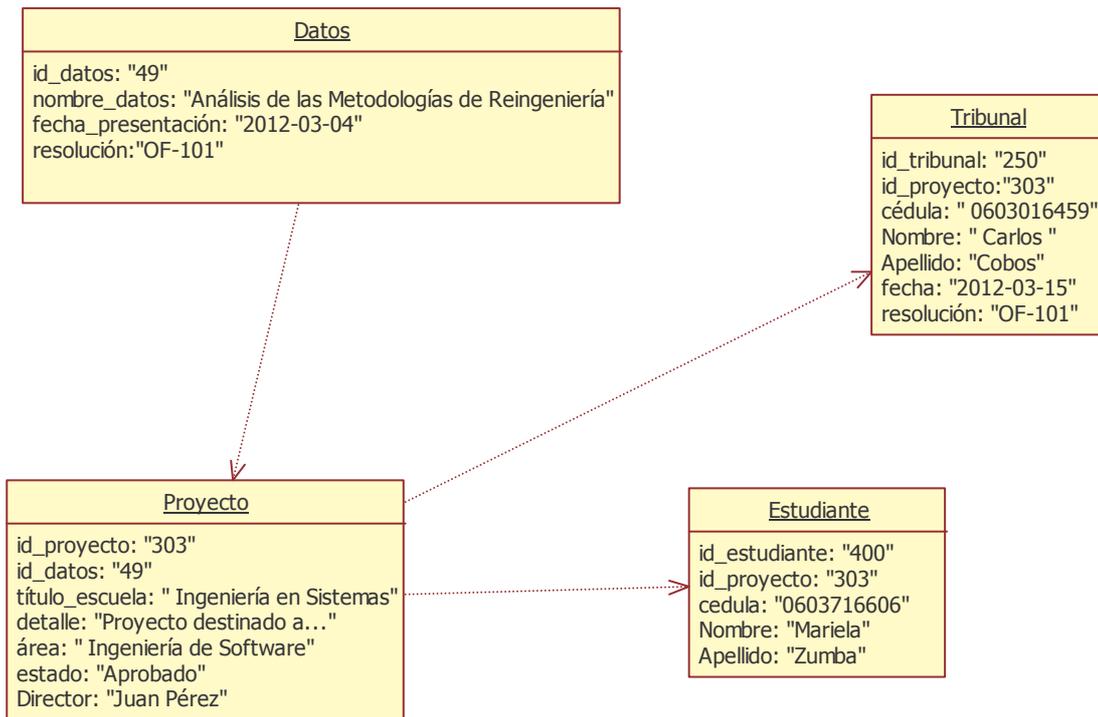


Ilustración 19 Modelo de objetos para ingresar Anteproyecto

5.3.1.3. Modelo de Objetos para Registrar Matricula

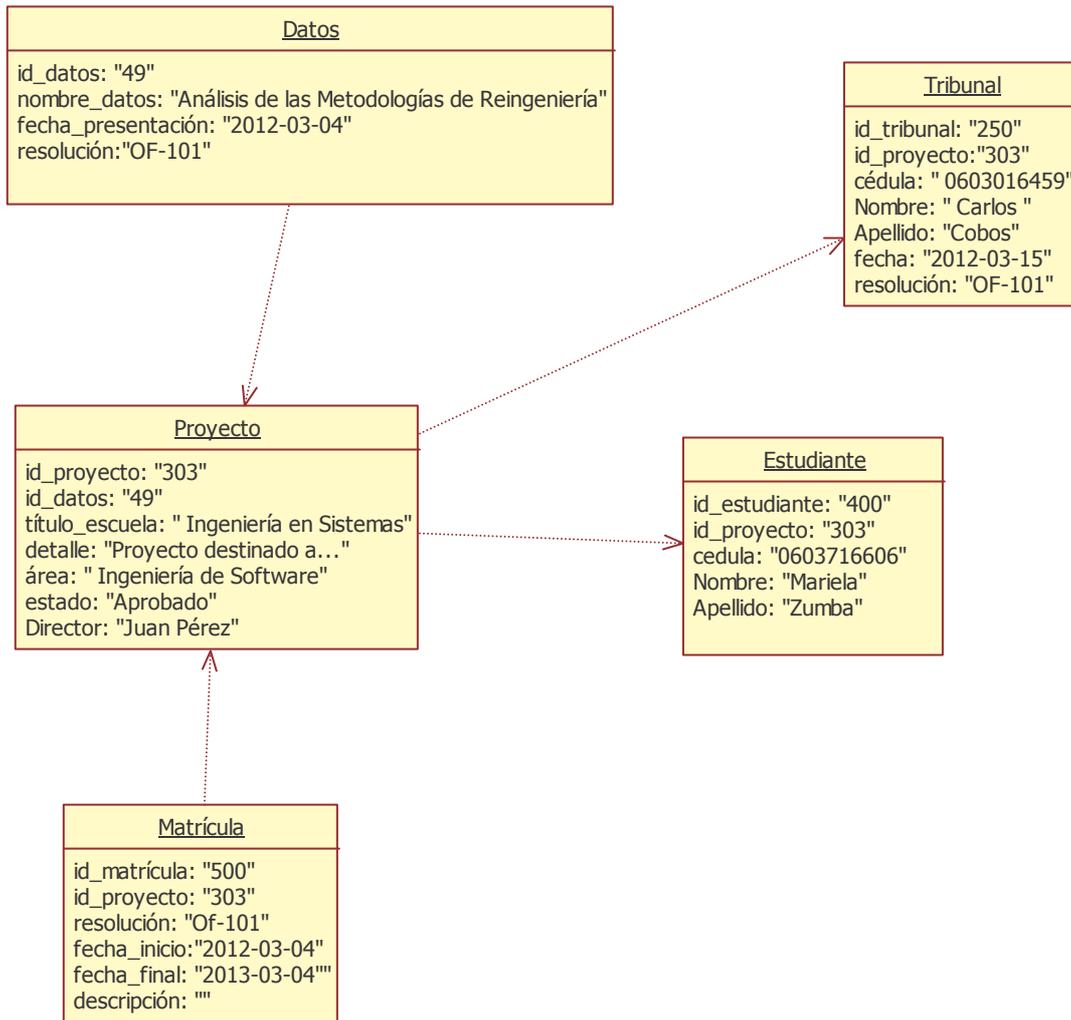


Ilustración 20 Modelo de objetos para registrar matrícula

5.3.1.4. Modelo de Objetos para Buscar

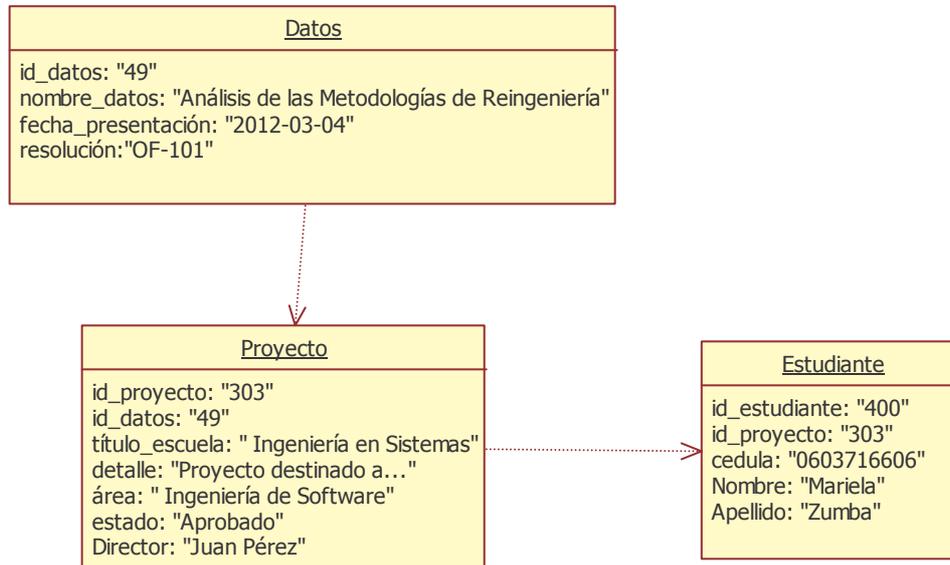


Ilustración 21 Modelo de objetos para buscar

5.3.1.6. Modelo de Objetos para Imprimir Actas

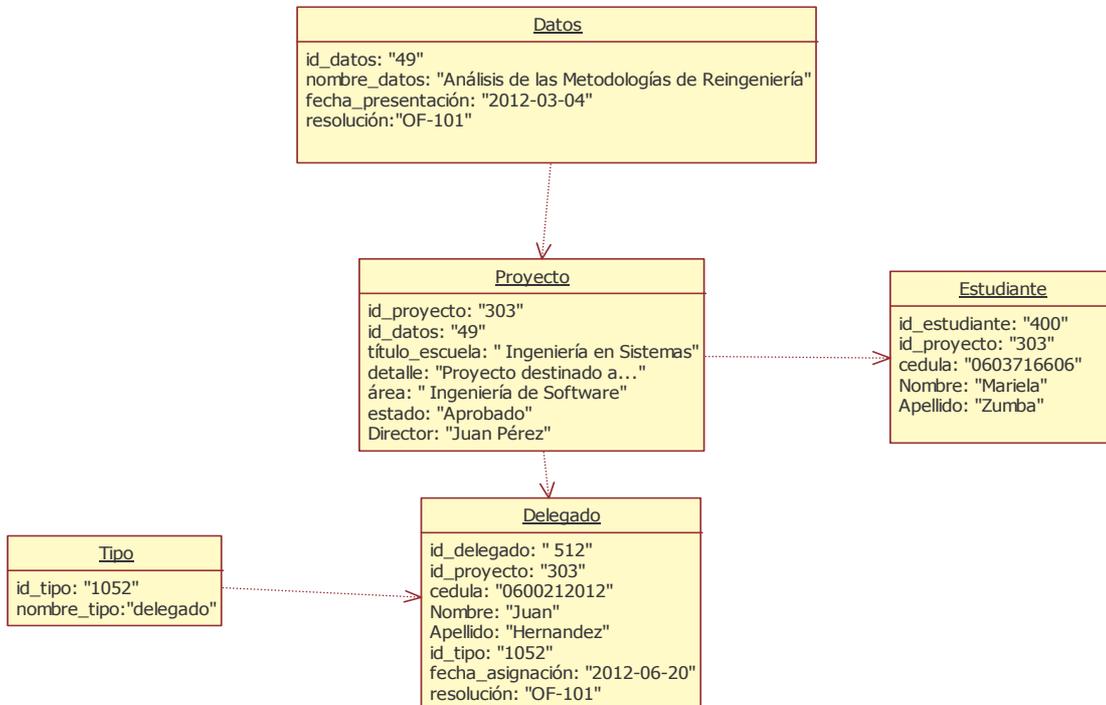


Ilustración 22 Modelo de objetos para imprimir actas

5.3.1.7. Modelo de Objetos para Mostrar Listados

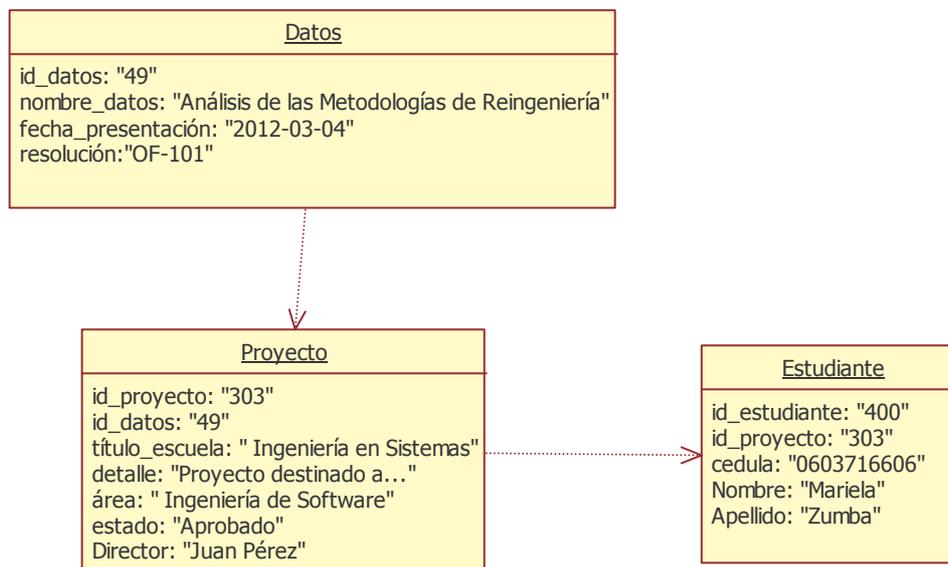


Ilustración 23 Modelo de objetos para mostrar listados

5.3.1.8. Modelo de Objetos para Ingresar Proyecto de Investigación

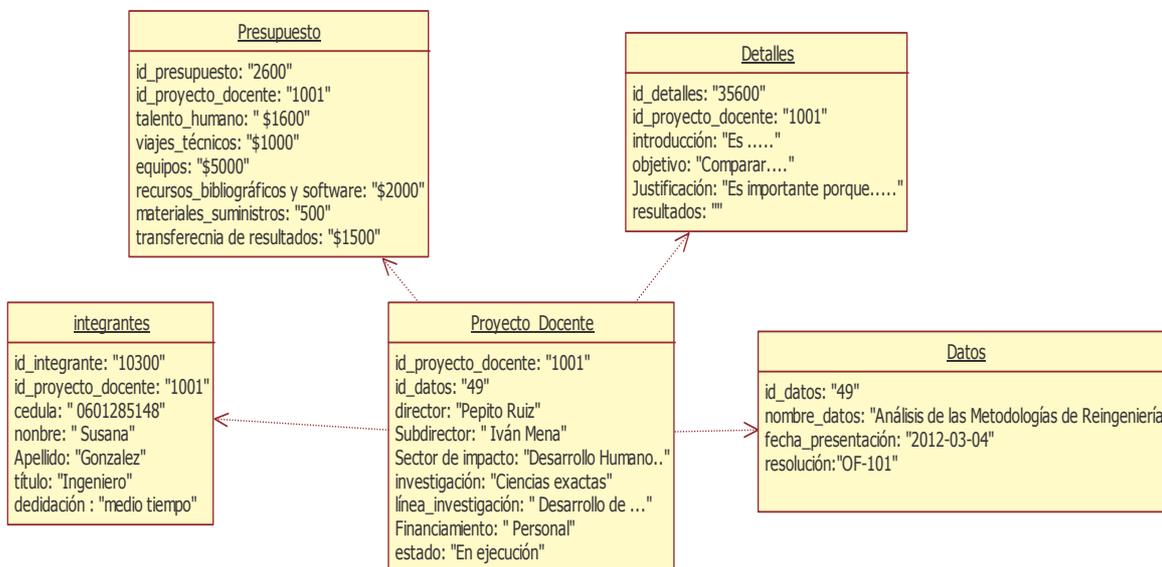


Ilustración 24 Modelo de objetos para ingresar proyecto de investigación

5.3.1.9. Modelo de Objetos para Ingresar Evento Científico

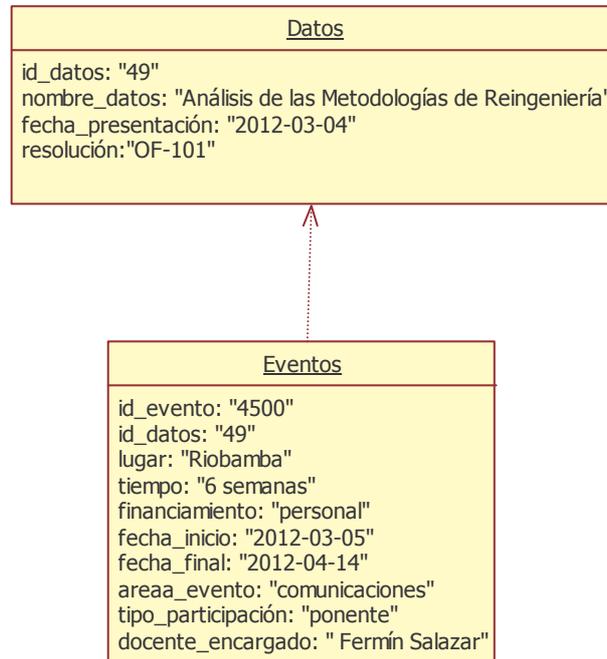


Ilustración 25 Modelo de objetos para ingresar evento científico

5.3.1.10. Modelo de Objetos para Ingresar Publicación.

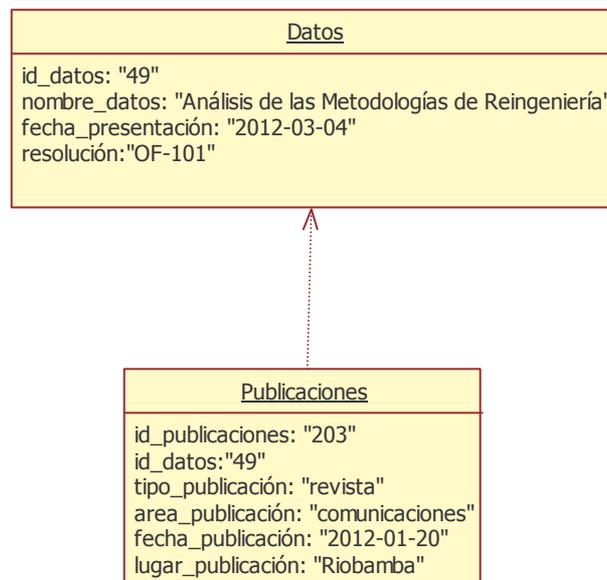


Ilustración 26 Modelo de objetos para ingresar publicación

5.3.2 DIAGRAMA DE SECUENCIA

5.3.2.1. Validar Usuario

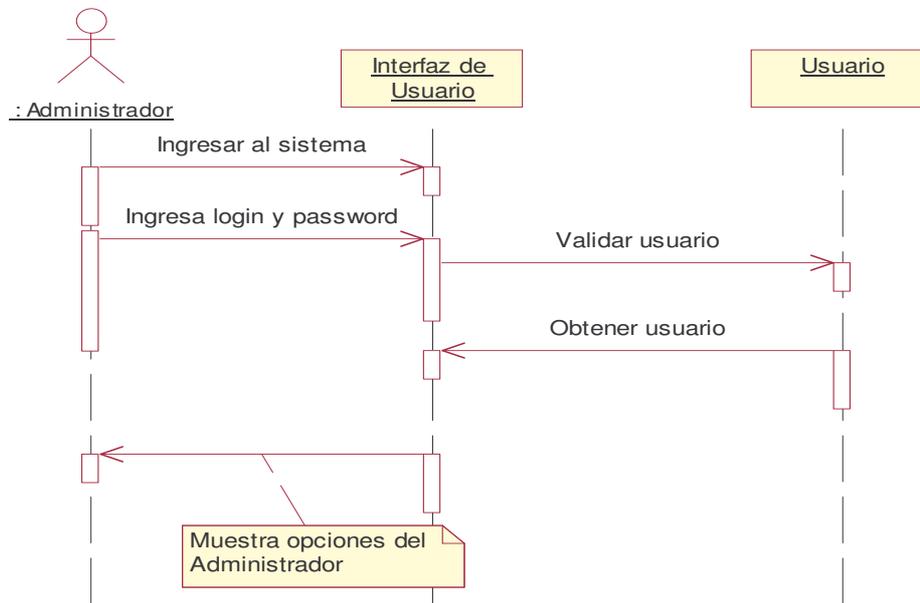


Ilustración 27 Diagrama de Secuencia validar usuario

5.3.2.2. Ingresar Proyecto

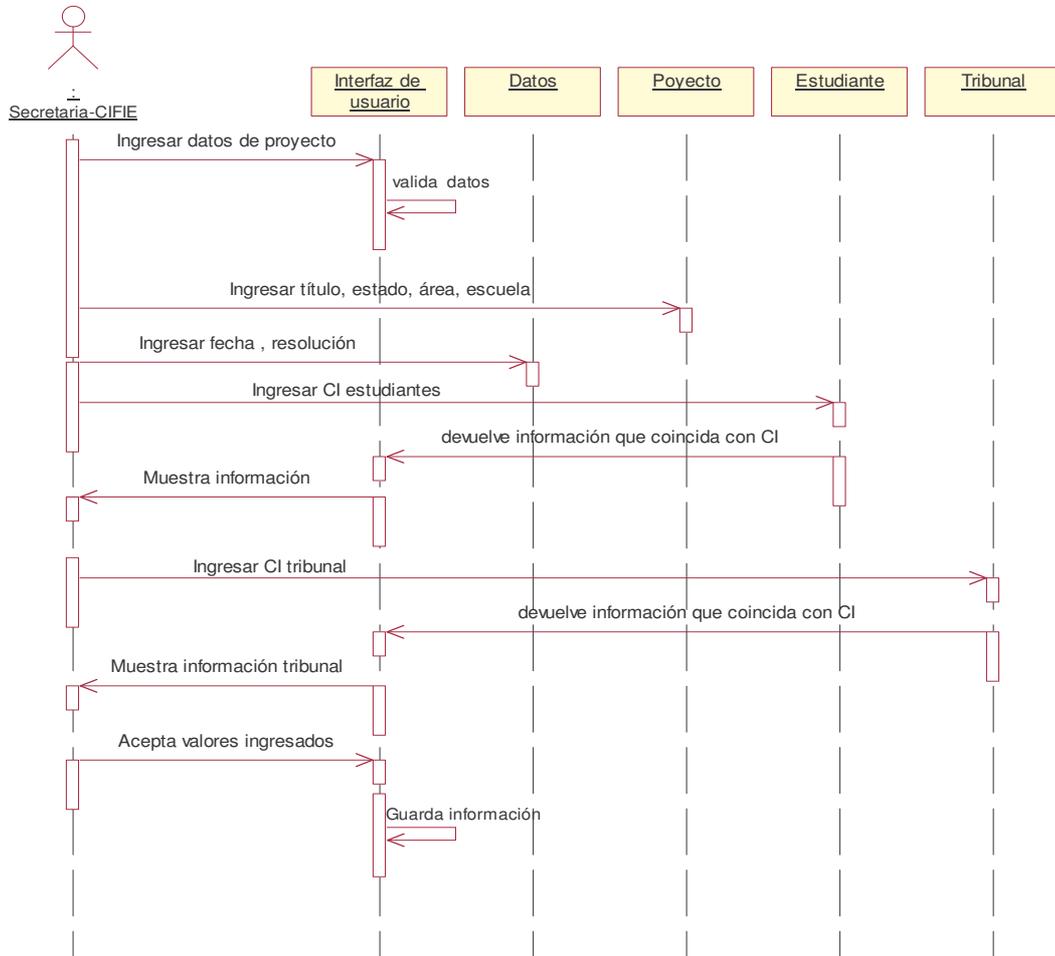


Ilustración 28 Diagrama de Secuencia de Ingresar proyecto

5.3.2.3. Registrar Matrícula

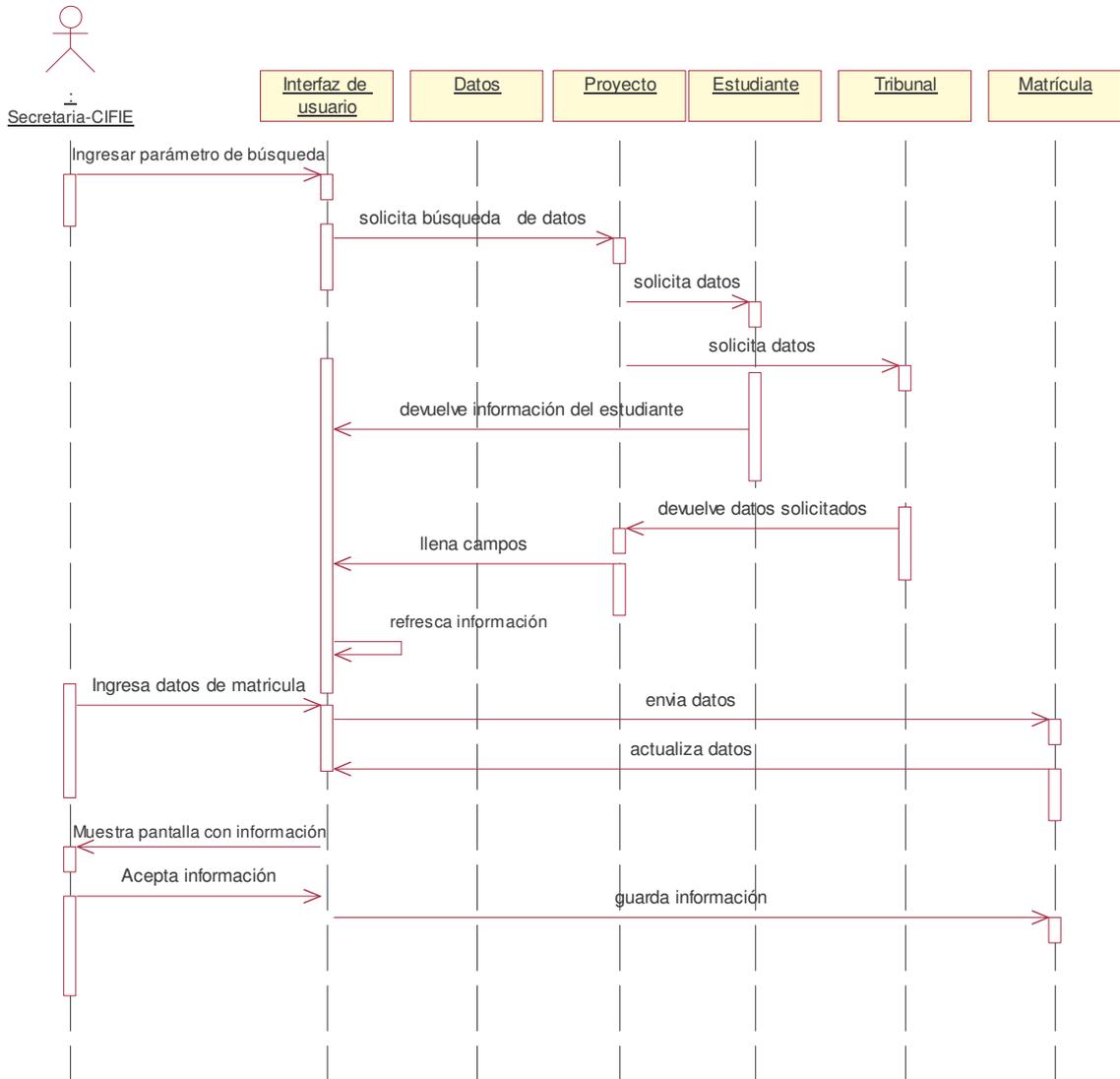


Ilustración 29 Diagrama de Secuencia Registrar Matrícula

5.3.2.4. Buscar

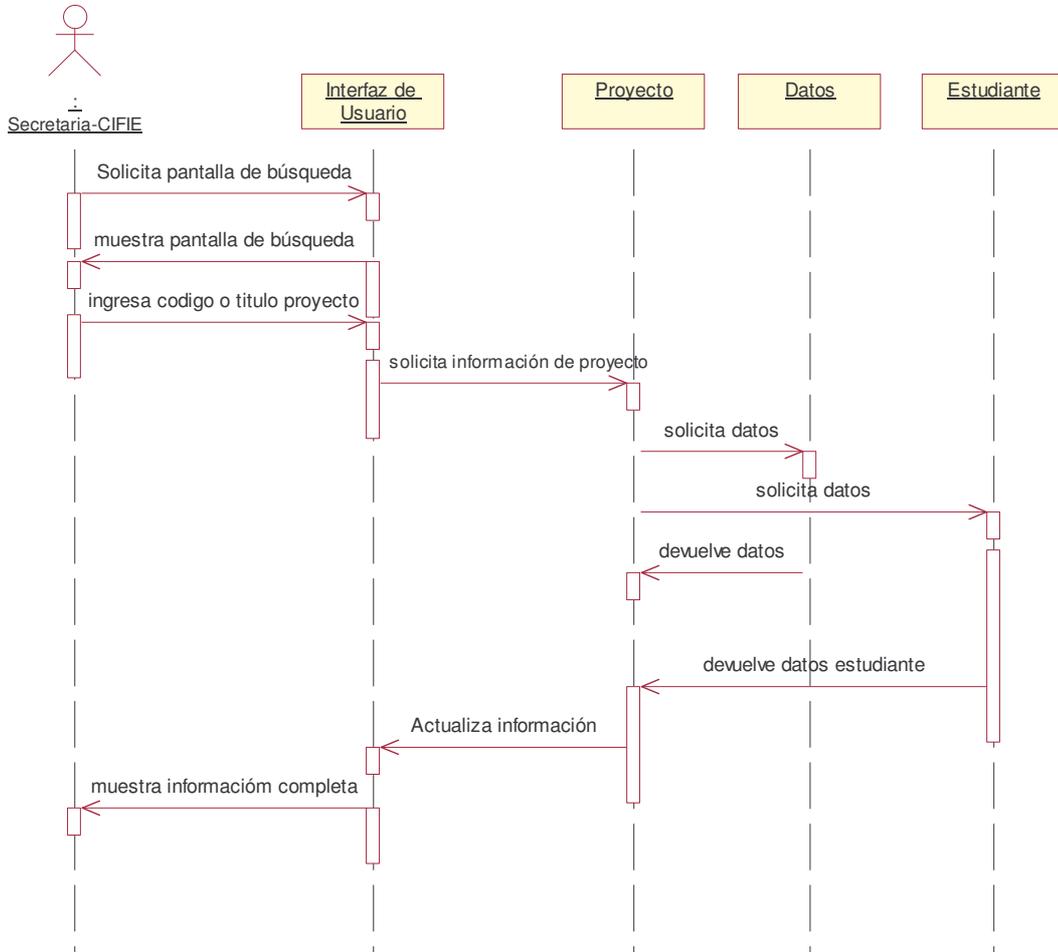


Ilustración 30 Diagrama de Secuencia Buscar

5.3.2.5. Imprimir Actas

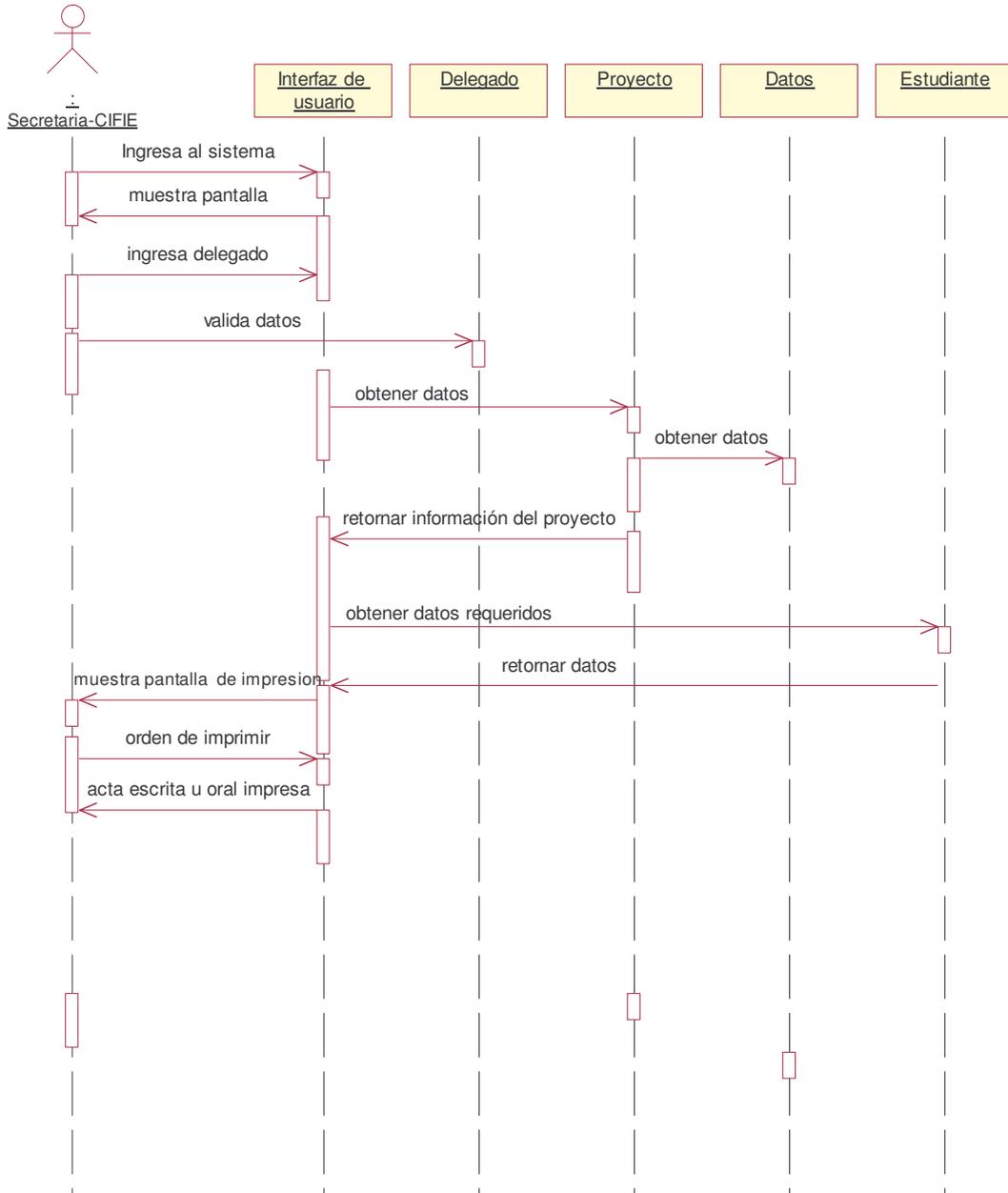


Ilustración 31 Diagrama de Secuencia imprimir actas

5.3.2.6. Mostrar Listados

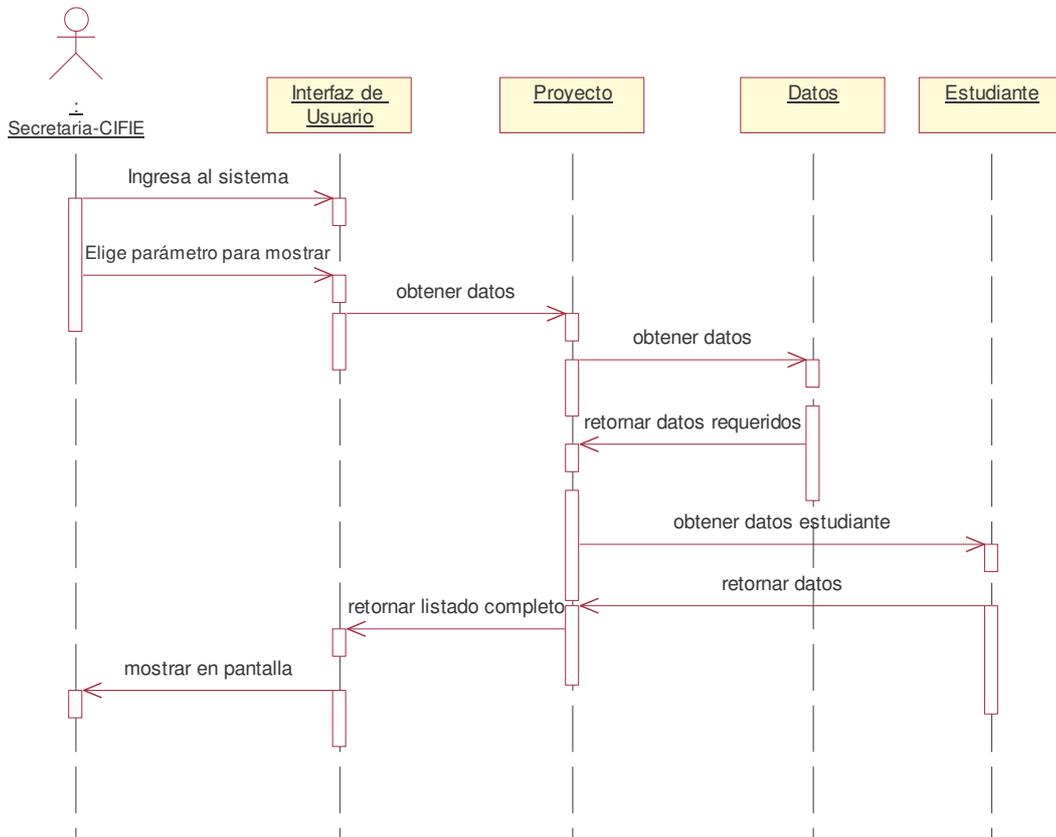


Ilustración 32 Diagrama de Secuencia mostrar listados

5.3.2.7. Ingresar Proyecto de Investigación

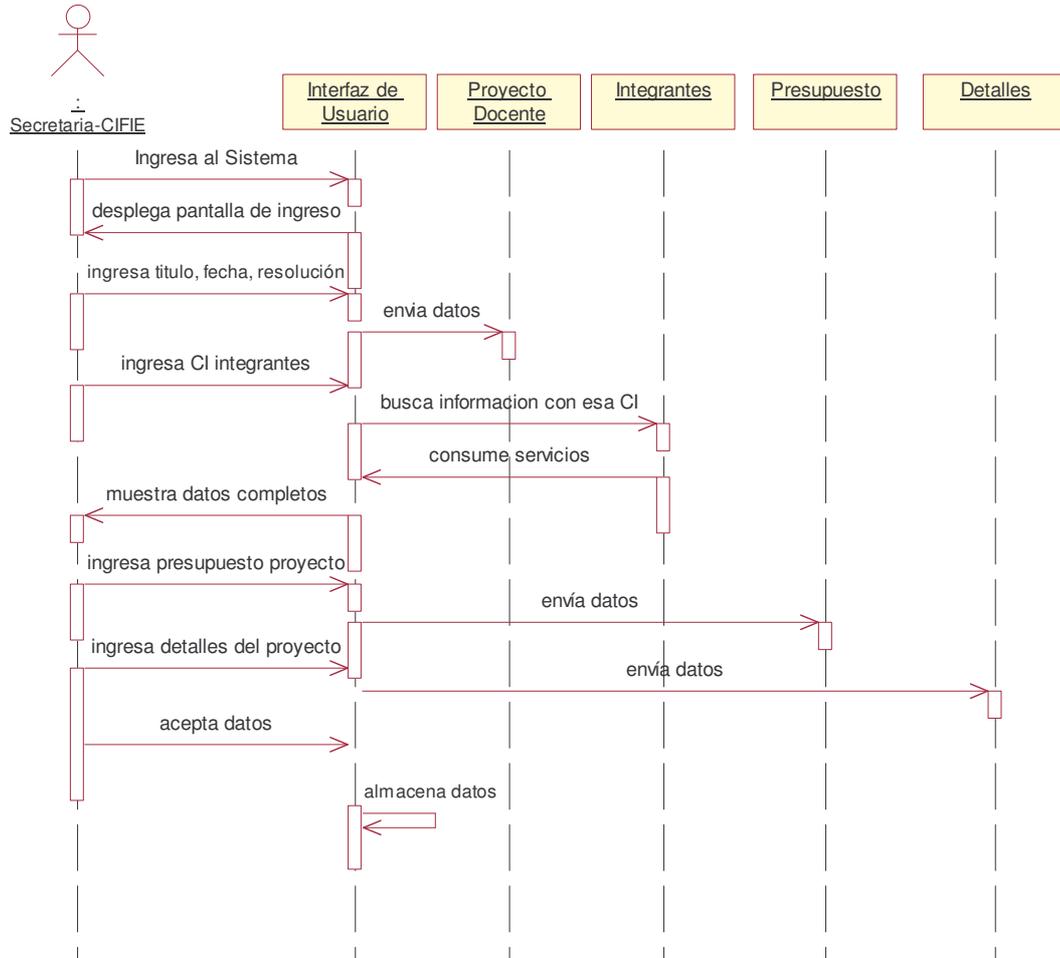
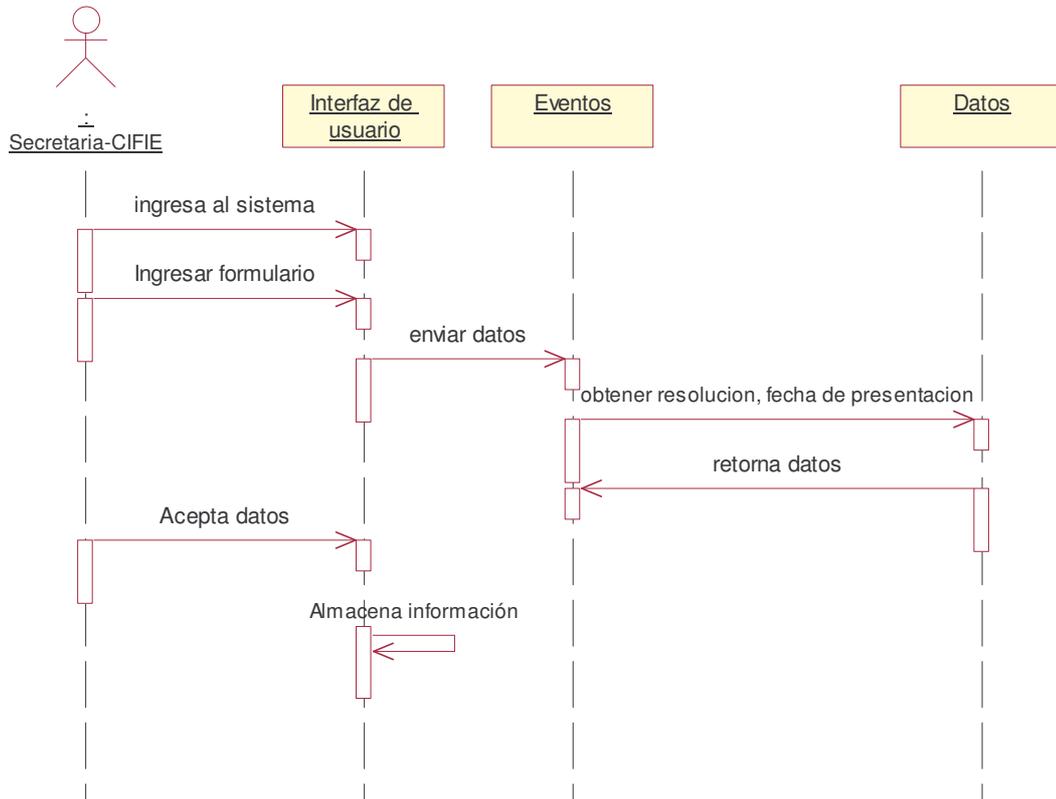


Ilustración 33 Diagrama de Secuencia ingresar proyecto de investigación

5.3.2.8. Ingresar Evento Científico



Ilustraci3n 34 Diagrama de Secuencia ingresar evento cientifico

5.3.2.9. Ingresar Publicaciones

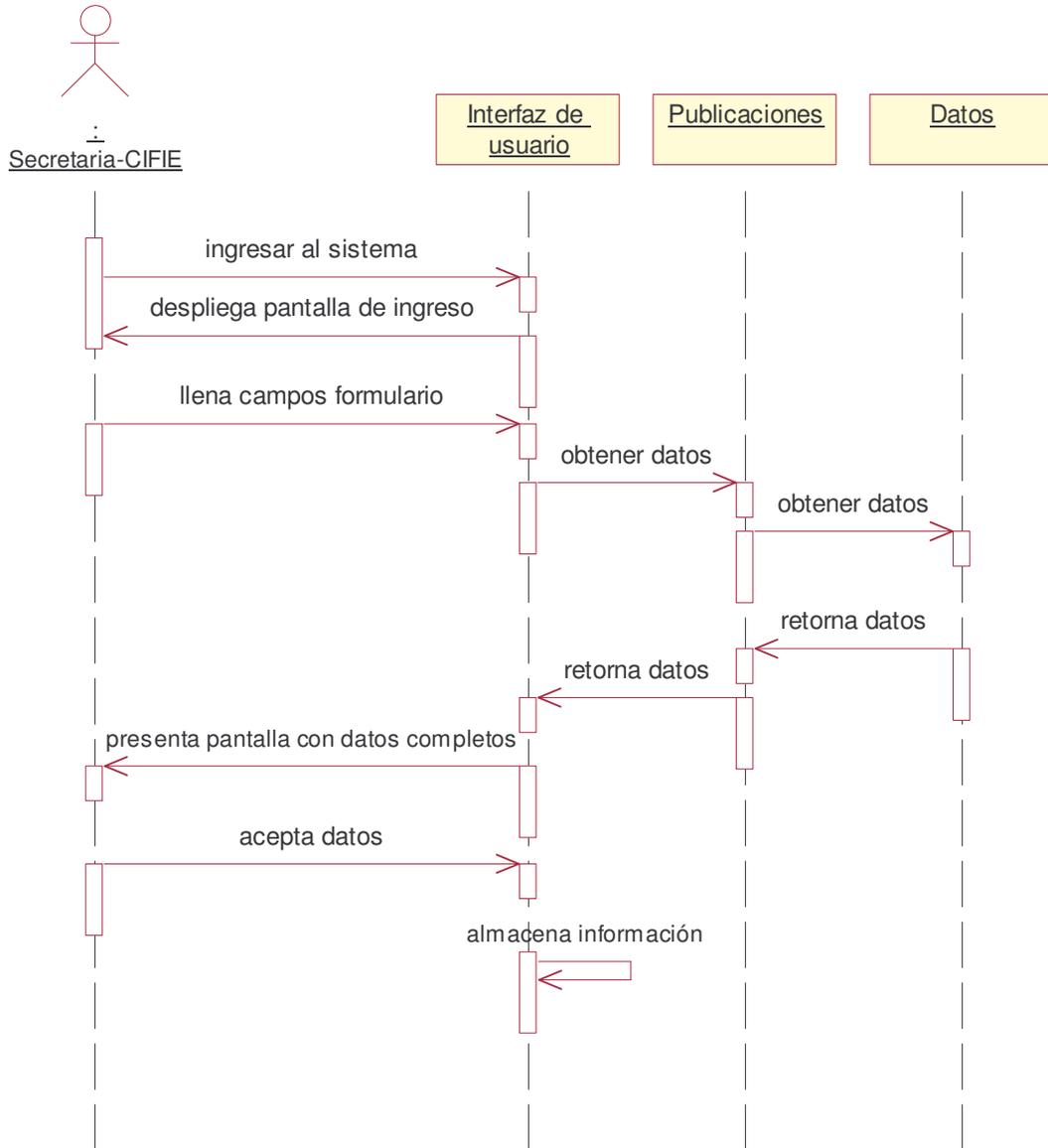


Ilustración 35 Diagrama de Secuencia registrar publicaciones

5.3.3 INTERFACES DE USUARIO

A Continuación se presenta las pantallas del nuevo Sistema SiGep, de acuerdo al usuario y a las actividades que puede realizar el mismo en el sistema.

5.3.3.1 Pantalla de Validación de usuario

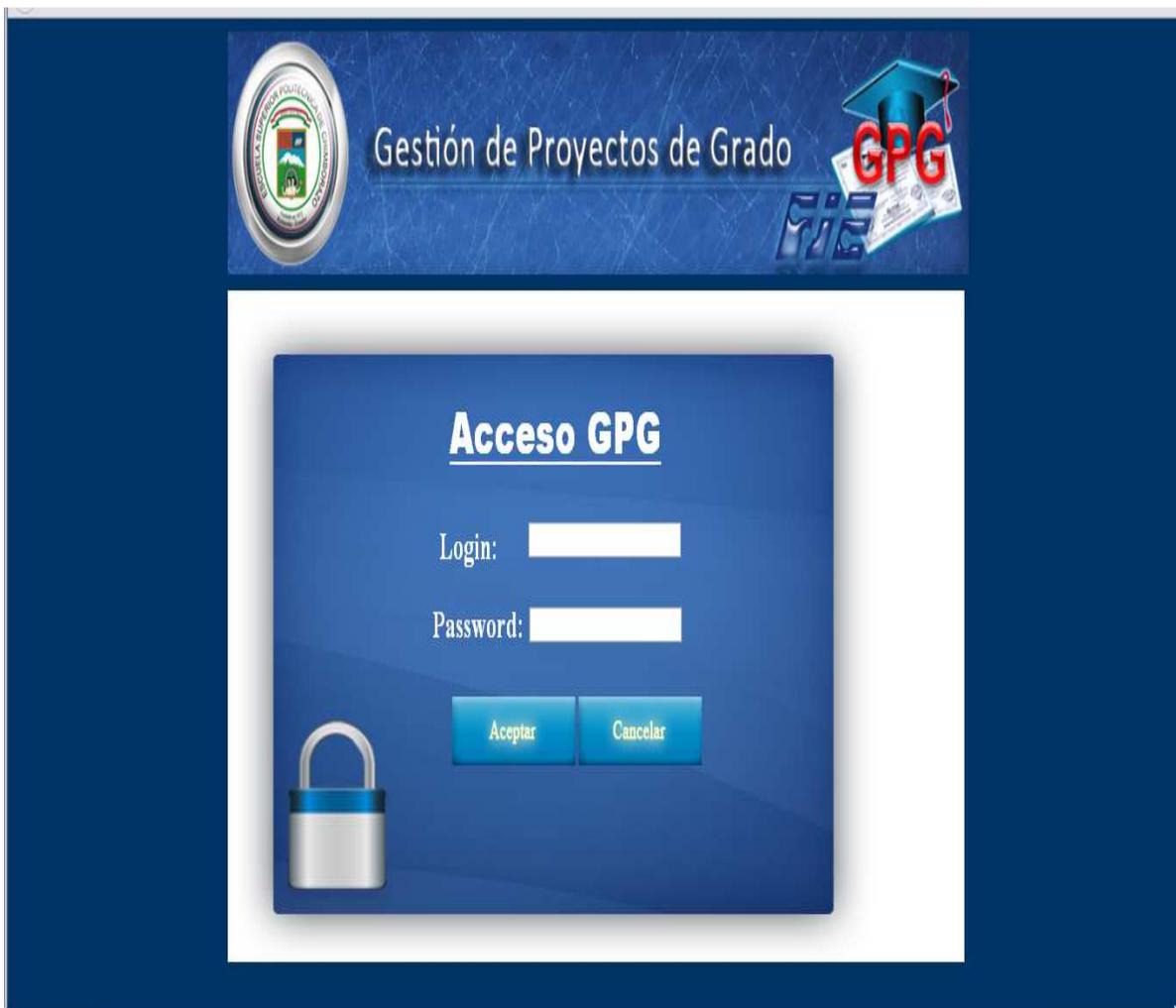


Ilustración 36 Pantalla validar usuario

5.3.3.2 Pantalla de Ingreso Anteproyecto

Projecto Proyecto Eventos Publicaciones Ayuda
Estudiante Investigación

PRESENTACION DE ANTEPROYECTOS

Fecha:	<input type="text"/>	Título proyecto:	<input type="text"/>	Estado:		
Area:	<input type="text"/>	Resolución	<input type="text"/>	En Proceso	Aprobado	Rechazado
Escuela:	sistemas	Integrantes:	<input type="radio"/> 1 <input type="radio"/> 2	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Estudiante 1						
Cédula:	<input type="text"/>					
Nombres:	<input type="text"/>					
Director	<input type="text"/>	Cédula	<input type="text"/>			
Tribunal de analisis						
Cédula	<input type="text"/>	Nombres	<input type="text"/>			
Cédula	<input type="text"/>	Nombres	<input type="text"/>			
				Aceptar	Cancelar	

Ilustración 37 Pantalla de ingreso anteproyecto

5.3.3.3 Pantalla de Registro de Matrícula

Código	Título del Proyecto	Estado	Continuar	Nombres
49	hola estamos probando esto	Aprobado	Siguiente	11111 1111 1111
49	hola estamos probando esto	Aprobado	Siguiente	2222 2222 2222

Integrantes:	Estudiante		
Cédula:	111111111-1	Nombre:	11111
Cédula:	22222222-2	Nombre:	2222

Tribunal de dirección				
Director	Cédula:	33333333-3	Nombres:	3333 3333 3333
Miembro	Cédula:	44444444-4	Nombres:	4444 4444 4444
Miembro	Cédula:	55555555-5	Nombres:	5555 5555 5555

Resolucion:	Fecha Inicio:	Fecha Fmal:
1212121212	1969-12-31 19:00:00.0	1969-12-31 19:00:00.0
131313	1969-12-31 19:00:00.0	1969-12-31 19:00:00.0
141414	1969-12-31 19:00:00.0	1969-12-31 19:00:00.0

Ilustración 38 Pantalla de registro de matrícula

5.3.3.4. Pantalla de Búsqueda



Ilustración 39 Pantalla de búsqueda de Anteproyecto

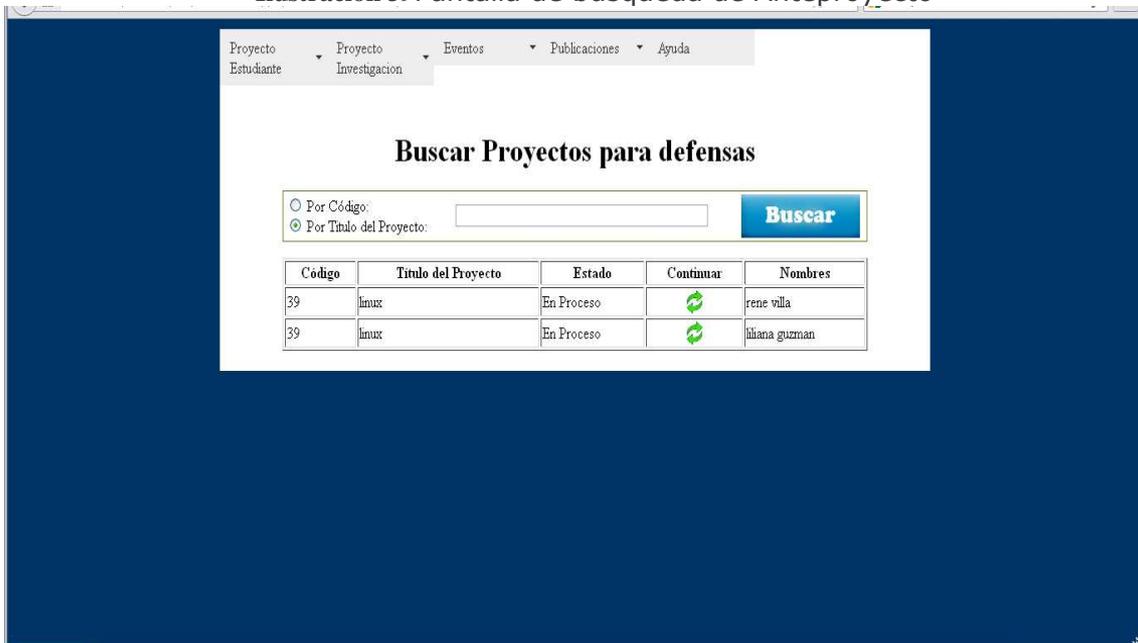


Ilustración 40 Pantalla de búsqueda de proyectos para defensa

5.3.3.5. Pantalla para Imprimir Actas

5.3.3.6. Pantalla para Mostrar Listados

5.3.3.7. Pantalla para Ingresar Proyecto de Investigación

The screenshot shows a web application interface for entering teacher projects. At the top, there is a navigation menu with options: Proyecto Estudiante, Proyecto Investigacion, Eventos, Publicaciones, and Ayuda. The main heading is 'PROYECTOS DOCENTES'. The form is divided into several sections: 'Datos Generales' with fields for Fecha, Resolución, and Título; 'Director Proyecto' and 'Subdirector Proyecto' fields; 'Area' and 'Estado del Proyecto' fields; a table for 'Sector de impacto', 'Area de Investigación', and 'Lineas de Investigación' with dropdown menus; 'Lugar', 'Duración', and 'Fecha Inicio' fields; 'Formulado', 'En Ejecución', and 'Ejecutado' status fields with radio buttons; 'Fecha Final' and 'U. Responsable' fields; 'Financiamiento' section with radio buttons for 'Institucional', 'Personal', 'Ambos', and 'Otros'; 'Integrantes' section with fields for 'Docentes' and 'Estudiantes' including 'Nombre', 'Cargo', and 'Cédula'; and 'Detalle Formal del Proyecto' section with an 'Introducción' field.

Ilustración 41 Pantalla para Ingresar Proyecto de Investigación

5.3.3.8. Pantalla para Ingresar Evento Científico

The screenshot shows a web application interface for entering academic scientific events. At the top, there is a navigation menu with options: Proyecto Estudiante, Proyecto Investigacion, Eventos, Publicaciones, and Ayuda. The main heading is 'PARTICIPACION EN EVENTOS ACADEMICOS CIENTIFICOS'. The form includes: 'Datos Generales' with fields for Fecha, Resolución, and Tema; 'Nombre Docente' field; 'Area' field; 'Sector de impacto' dropdown menu; 'Lugar', 'Duración', and 'Fecha Inicio' fields; 'Formulado', 'En Ejecución', and 'Ejecutado' status fields with radio buttons; 'Fecha Final' and 'U. Responsable' fields; 'Tipo de Participación' section with radio buttons for 'Ponente' and 'Asistente'; and 'Aceptar' and 'Cancelar' buttons at the bottom.

Ilustración 42 Pantalla para Ingresar evento científico

5.3.3.9. Pantalla para Ingresar Publicaciones

Proyecto Proyecto Eventos Publicaciones Ayuda
Estudiante Investigación

PUBLICACIONES

Datos Generales:			
Fecha	<input type="text"/>	Resolución:	<input type="text"/>
Nombre Publicación	<input type="text"/>		
Area:	<input type="text"/>	Lugar:	<input type="text"/>
Tipo Publicación			
Libros	<input type="radio"/>	Textos Basicos	<input type="radio"/>
		Articulos Cientificos	<input type="radio"/>
		Revistas	<input type="text"/>
		Indexadas	<input type="radio"/>
		No Indexadas	<input type="radio"/>
<input type="button" value="Aceptar"/>		<input type="button" value="Cancelar"/>	

Ilustración 43 Pantalla de ingreso de publicación

5.4 DISEÑO

5.4.1 DISEÑO DE MODELADO DE OBJETOS

5.4.2 DISEÑO DE LA ARQUITECTURA

La Arquitectura global del Sistema Sigep es la siguiente:

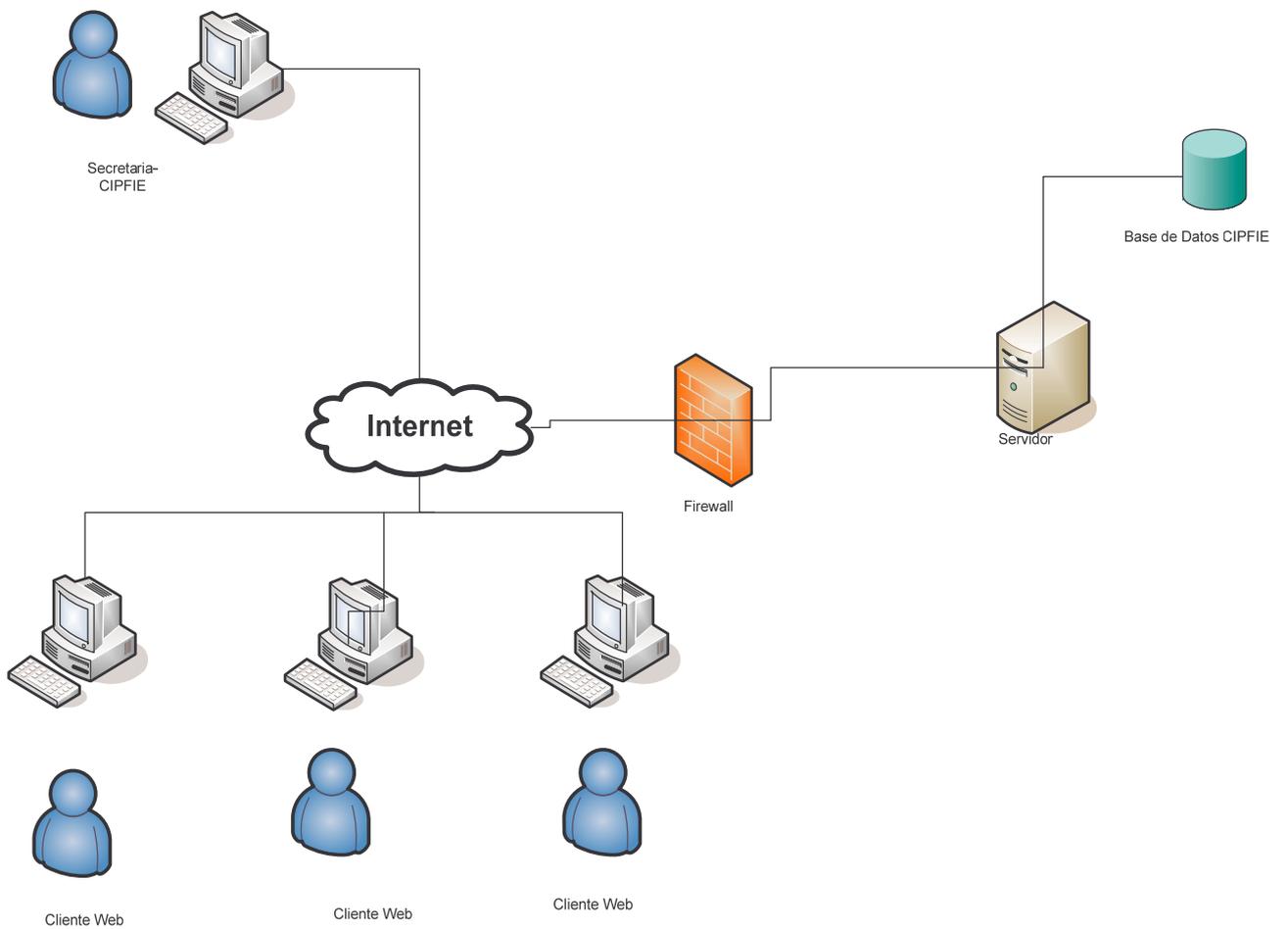


Ilustración 44 Diagrama de Red

Como podemos observar en el diagrama para que funcione el Sistema Sigep se necesita del Servidor de Base de Datos, Servidor de Aplicaciones, Servidor Web que a

la vez actuará como Firewall. Además el Sistema Sigep correrá bajo el Sistema Operativo Linux.

Tipos de usuario:

Usuario de Escritorio (Secretaria-CIPFIE): El cual va a ser utilizado por la Secretaria del CIPFIE para realizar varias tareas como almacenamiento, control, y generacion de informacion. El cliente puede correr en la plataforma Windows.

Usuario Web (Estudiantes/público en general): Los cuales podrán ser utilizados por usuarios de consulta que podrán acceder desde cualquier maquina únicamente con el requisito de que tenga acceso a Internet.

5.4.3 DIAGRAMA DE DESPLIEGUE

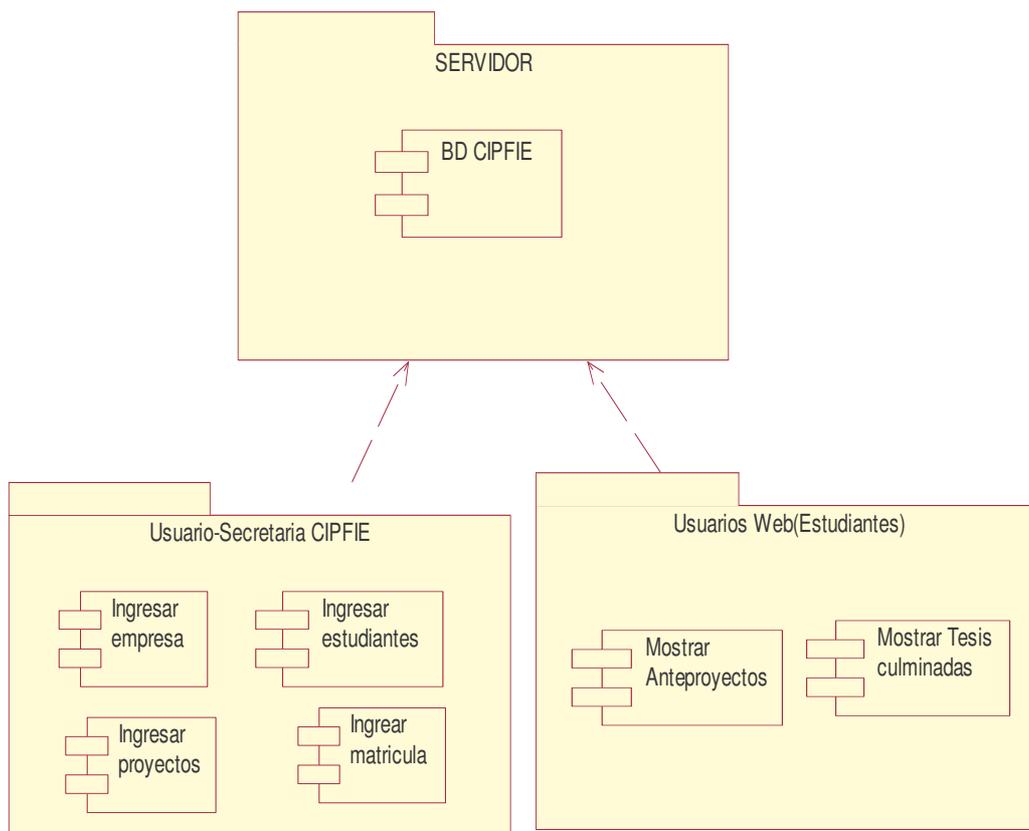


Ilustración 45 Diagrama de Despliegue del Sistema

5.4.4 DIAGRAMA DE COMPONENTES

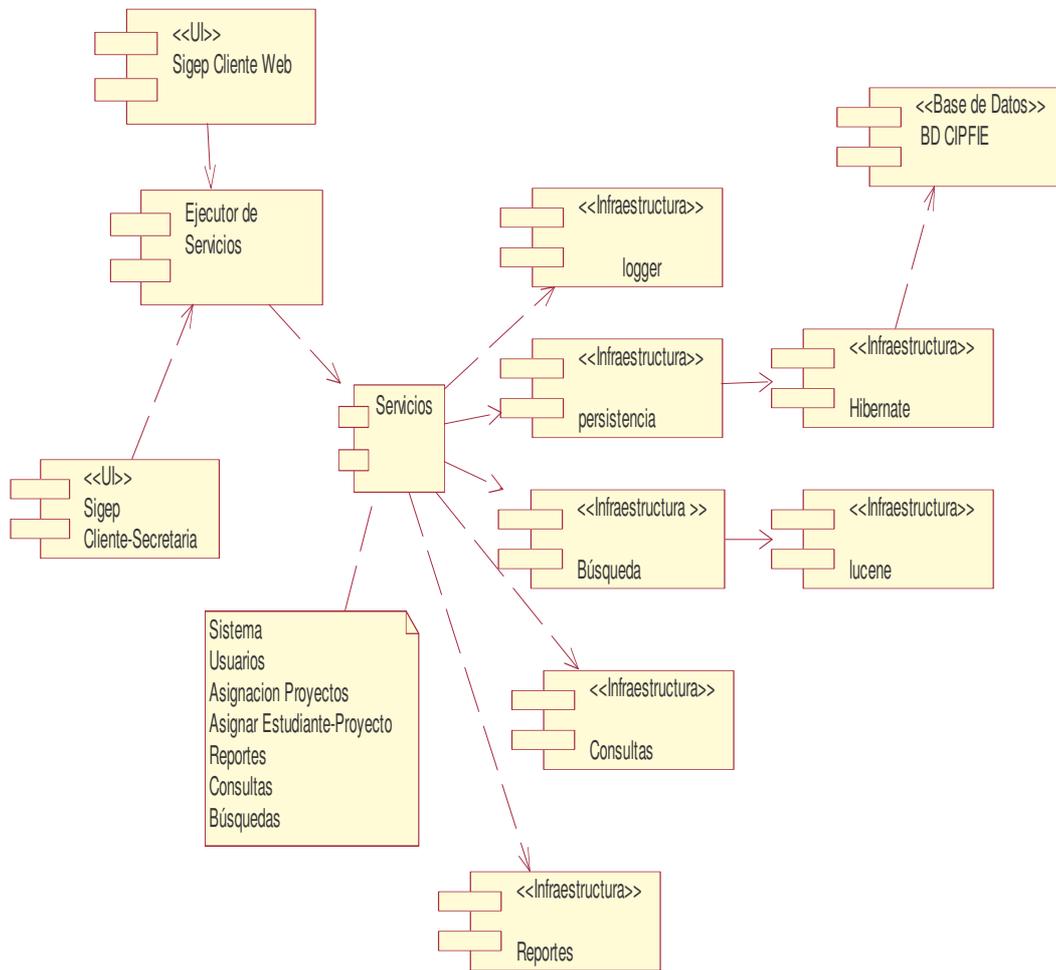


Ilustración 46 Diagrama de Componentes del Sistema

5.5 IMPLEMENTACIÓN

5.5.1 PLATAFORMA

La plataforma a utilizar en el desarrollo del Sistema SiGep es Java, ya que a más de ser un lenguaje de programación, constituye un plataforma completa al momento de desarrollar aplicaciones, por este motivo existen gran cantidad de herramientas.

Java brinda independencia de plataforma, esto significa que un programa escrito en java puede correr en cualquier tipo de hardware con el único requisito de que tenga instalado la máquina virtual de Java (JVM).

Además de lo anterior Java también ofrece un soporte completo para lo que POO (Programación Orientada a Objetos) y por tanto soporta comportamientos de la POO como: herencia, polimorfismo, ecapsulamiento.

5.5.2 HERRAMIENTAS A UTILIZARSE PARA EL DESARROLLO

5.5.2.1. Herramientas para Modelado

Para especificar, analizar, diseñar el sistema Sigep antes de codificarlo utilizaremos la herramienta: **Rational Rouse**, que es una herramienta software para el modelado visual mediante UML de Sistmas Software.

Se eligió esta herramienta ya que ofrece entre otras las siguientes características.

- ✓ Mantiene la consistencia de los modelos del sistema software
- ✓ Chequeo de la sintaxis UML
- ✓ Generación Documentación automáticamente
- ✓ Generación de Código a partir de los Modelos
- ✓ Ingeniería Inversa (crear modelo a partir código)

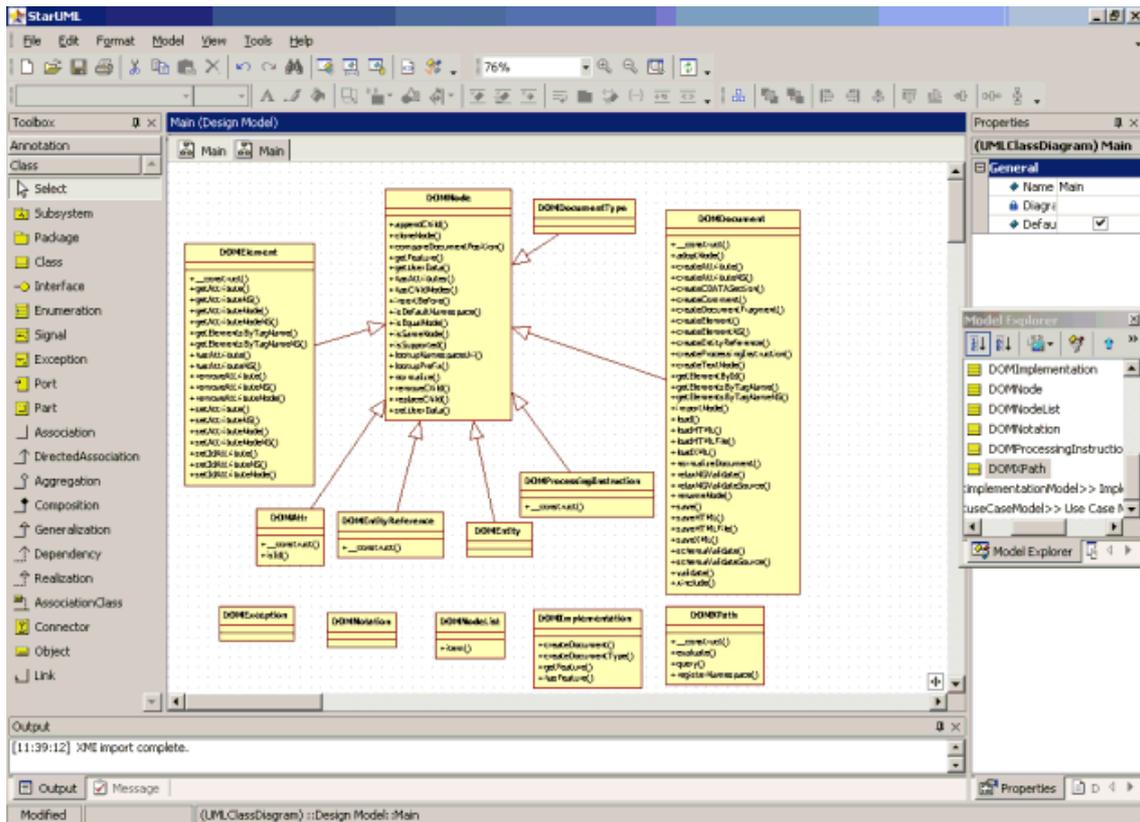


Ilustración 48 StarUML Herramienta para modelado

También se utilizó otra herramienta para la diagramación de la arquitectura del sistema como lo es **Microsoft Visio** es un programa de diagramación para Windows que utiliza gráficos vectoriales para crear diagramas

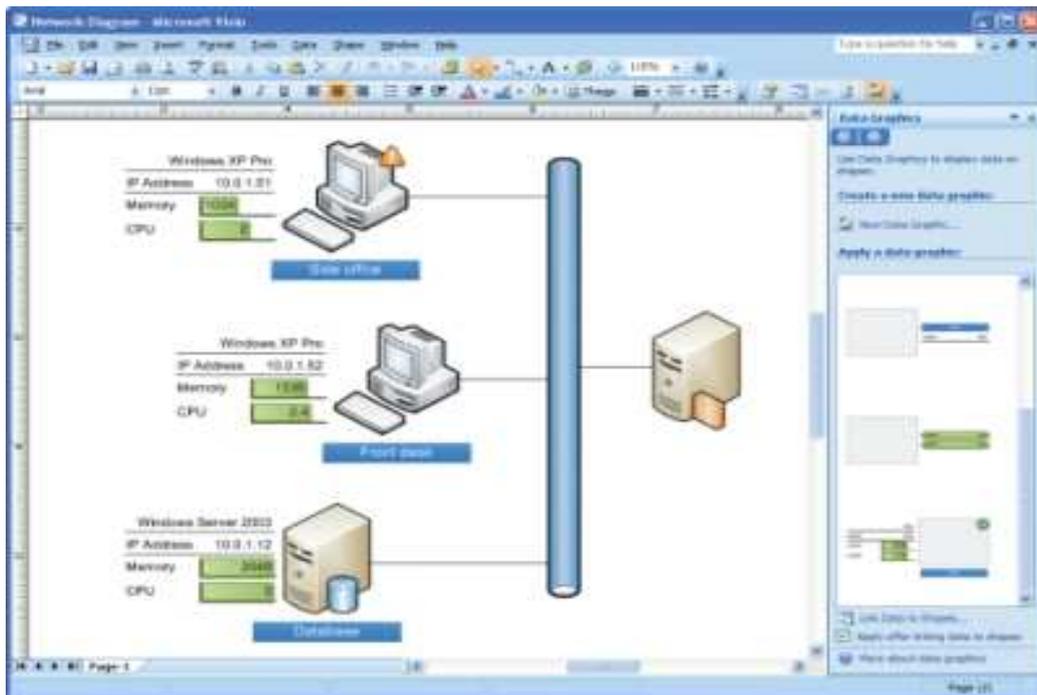


Ilustración 49 Microsoft Visio Herramienta para modelado

5.5.2.2. Herramientas para desarrollo de Interfaz de Usuario

Para el diseño de la página web se utilizó **Dreamweaver** que probablemente sea el mejor editor de Páginas Web. Aunque sea un experto programador de HTML el usuario que lo maneje, siempre se encontrarán en este programa razones para utilizarlo, sobretodo en lo que a productividad se refiere.

Cumple perfectamente el objetivo de diseñar páginas con aspecto profesional, y soporta gran cantidad de tecnologías, además muy fáciles de usar:

- Hojas de estilo y capas
- Javascript para crear efectos e interactividades
- Inserción de archivos multimedia...

Además es un programa que se puede actualizar con componentes, que fabrica tanto Macromedia como otras compañías, para realizar otras acciones más avanzadas.

En resumen, el programa es realmente satisfactorio, incluso el código generado es de buena calidad. La única pega consiste en que al ser tan avanzado, puede resultar un poco difícil su manejo para personas menos experimentadas en el diseño de webs.

Dreamweaver ha evolucionado mucho en su versión 4, que incluye soporte para la creación de páginas dinámicas de servidor en ASP, con acceso a bases de datos (versión Ultradev) y una mayor integración con otras herramientas de Macromedia como Fireworks.



Ilustración 50 Macromedia Dreamweaver programa para desarrollo de páginas web

5.5.2.3. Herramientas para desarrollo de Base de Datos

ProgreSQL es un lenguaje de administración de bases de datos, destacando sus partes, tablas, campos que al ser un excelente lenguaje de soporte para consultas con varios elementos recopilados al nivel web sirve para manejo y total fiabilidad al hacerse de inventarios de negocios.

Se han creado Programas facilitando el uso tanto del ProgreSQL, SQL y Oracle a los mas inexpertos pero con una eficiencia del 100%

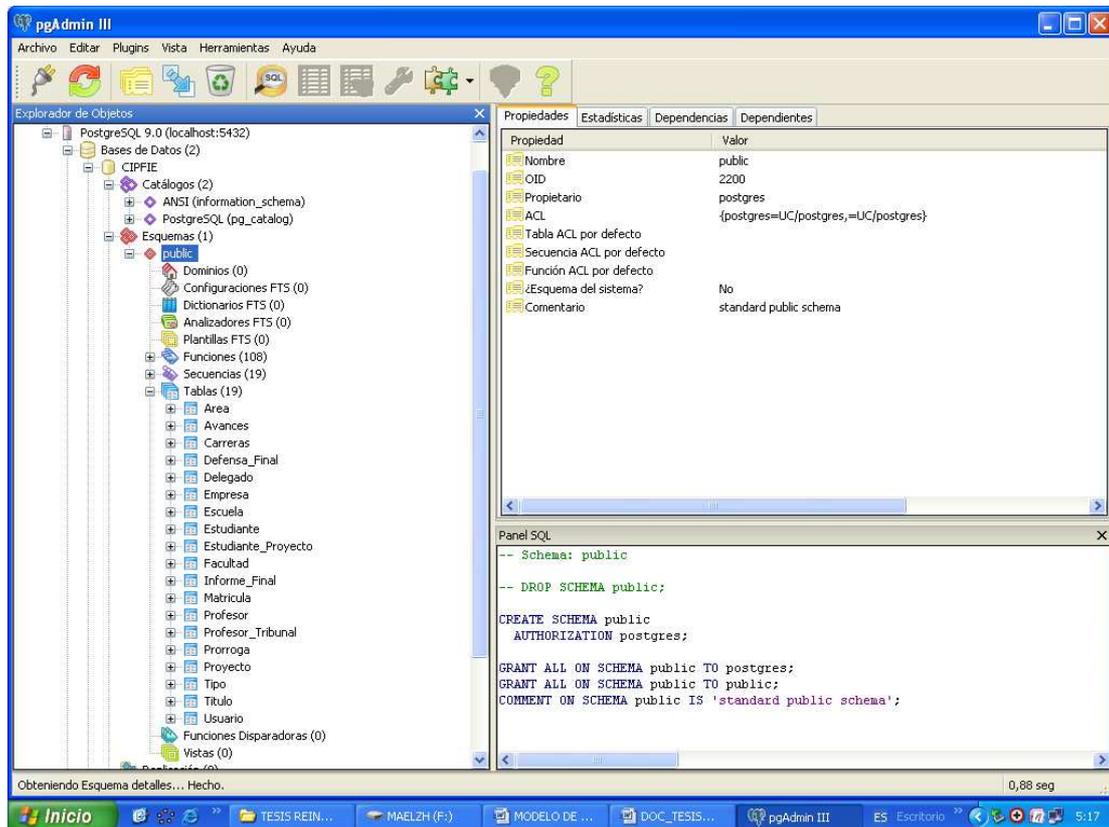


Ilustración 51 ProgreSQL Herramienta para administracion de Base de Datos

5.5.2.4. Herramientas para Programación

Para la programación se utilizó **NetBeans** que es un entorno de desarrollo integrado libre utilizando lenguaje de programación Java.

La plataforma NetBeans permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos. Un módulo es un archivo Java que contiene clases de java escritas para interactuar con las APIs de NetBeans y un archivo especial (manifest file) que lo identifica como módulo. Las aplicaciones

construidas a partir de módulos pueden ser extendidas agregándole nuevos módulos. Debido a que los módulos pueden ser desarrollados independientemente, las aplicaciones basadas en la plataforma NetBeans pueden ser extendidas fácilmente por otros desarrolladores de software.

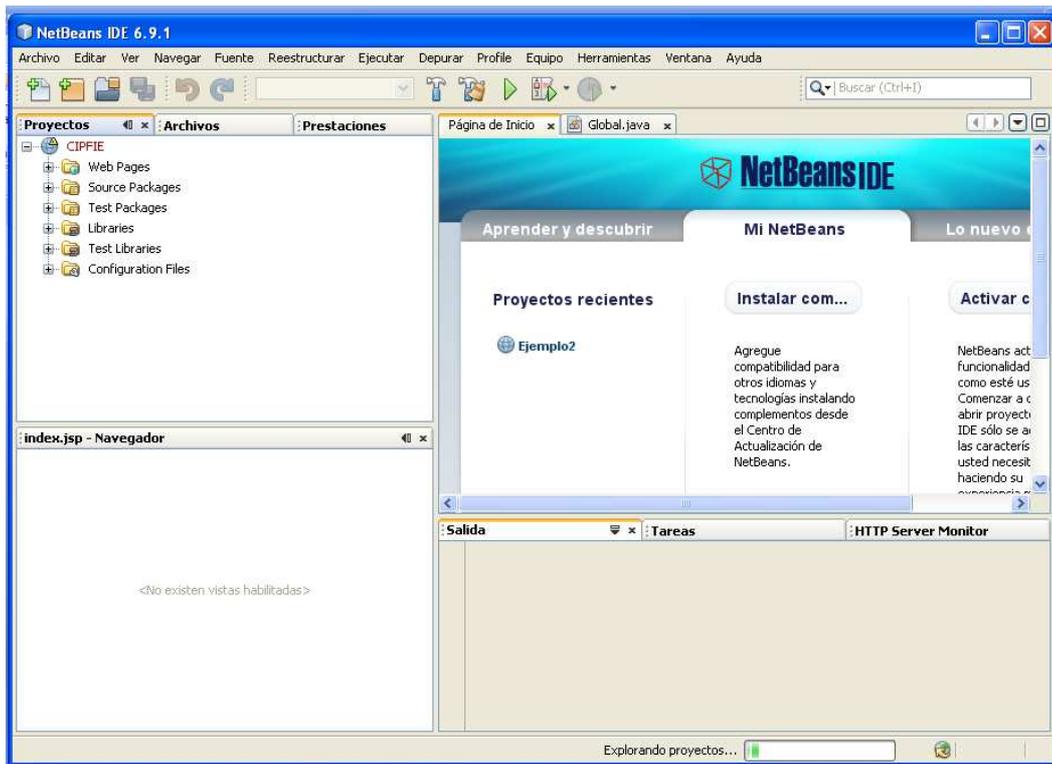


Ilustración 52 Netbeans Herramienta para programación

5.5.2 CONSTRUCCIÓN DEL SISTEMA

5.6 PRUEBAS

5.6.1 OBJETIVO

Verificar que el sistema Sigep, que se ha desarrollado satisface las necesidades y requisitos funcionales.

5.6.2 AMBIENTE

Para los casos de prueba se ha tomado los dos aspectos:

- ✓ Hardware
- ✓ Software

✓ Datos

5.6.2.1 HARDWARE

Para realizar las pruebas con respecto al hardware se ha instalado el sistema Sigep en una máquina con las siguientes características:

EQUIPO	ELEMENTO	CAPACIDAD
Servidor	Procesador	Pentium 4 3.0 Ghz
	Memoria	1 GB
	Disco Duro	80 GB
	Tarjeta de Red	10/100 Mbps
Cliente	Procesador	Intel 1.73 Ghz
	Memoria	256 MB
	Disco Duro	40 GB
	Tarjeta de Red	10/100 Mbps

5.6.2.2 SOFTWARE

Las características del Software utilizado para las pruebas son las siguientes:

EQUIPO	TIPO DE APLICACIÓN	NOMBRE
Servidor	Sistema Operativo	Fedora Core 6
	Servidor de Aplicaciones	Jboss 4.0.2
	Máquina Virtual Java	JVM 1.5
Cliente de Escritorio	Sistema Operativo	Windows XP
	Máquina Virtual java	JVM 1.5
Cliente Web	Sistema Operativo	Windows XP
	Visor Web	Firefox

5.6.2.3 DATOS

Los Datos que se utilizaron fueron Datos ficticios que se tomaron como ejemplo.

5.6.3 USUARIOS PARA LAS PRUEBAS

Los Roles de usuarios que utilizaron el Sistema y relizaron las pruebas son los definidos como Actores del Sistema:

- ✓ Administrador
- ✓ Secretaria-CIPFIE
- ✓ Estudiantes

5.6.4 CASOS DE PRUEBA

Los Casos de prueba que se tomaron para verificar la correcta funcionalidad del Sistema son los definidos en el apartado 5.2.3.

5.6.4.1 CASO DE PRUEBA VALIDAR USUARIO

CASO DE PRUEBA VALIDAR USUARIO	
PRECONDICIONES	
El Administrador deberá estar registrado en el sistema.	
PASOS	
ENTRADA	RESULTADO
1. El Administrador solicita autenticación	2. Muestra la pantalla de validación
3. El Administrador Ingresa su login y password	4. Muestra la pantalla principal de acuerdo al tipo de usuario que se haya autntivado.

Tabla XX Caso de Prueba Validar usuario

5.6.4.2 CASO DE PRUEBA INGRESAR ANTEPROYECTO

CASO DE PRUEBA INGRESAR ANTEPROYECTO	
PRECONDICIONES	
El Administrador deberá estar registrado para poder ingresar nuevos datos.	
PASOS	
ENTRADA	RESULTADO
1. El Administrador llena formulario	2. Valida los datos ingresados
3. El usuario acepta los datos ingresados	4. Guarda la información.

5.6.4.3 CASO DE PRUEBA REGISTRAR MATRÍCULA

CASO DE PRUEBA REGISTRAR MATRÍCULA	
------------------------------------	--

PRECONDICIONES	
El Usuario deberá estar registrado para poder ingresar nuevos datos.	
PASOS	
ENTRADA	RESULTADO
1. El Administrador. envia a buscar el Anteproyecto	2. Muestra la pantalla para con los datos del Proyecto
3. El usuario ingreas datos necesários para registrar matricula.	4. Valida y guarda los datos.

Tabla XXI Caso de Prueba registrar usuario

5.6.4.4 CASO DE PRUEBA BUSCAR

CASO DE PRUEBA BUSCAR	
PRECONDICIONES	
El Administrador deberá estar registrado para poder ingresar nuevos datos.	
PASOS	
ENTRADA	RESULTADO
1. El Aministrador ingresa el parámetro de búsqueda.	2. Valida datos y muestra pantalla con información solicitada.

Tabla XXII Caso de Prueba buscar

5.6.4.5 CASO DE PRUEBA IMPRIMIR ACTAS

CASO DE PRUEBA INGRESAR TÍTULO	
PRECONDICIONES	
El Administrador deberá estar registrado para poder ingresar nuevos datos.	
PASOS	
ENTRADA	RESULTADO
1. El Administrador ingresa delegado	2. Muestra pantalla para ingreso de dato.
3. Solicita impresión.	4. Acta impresa,

Tabla XXIII Caso de Prueba Imprimir Actas

5.6.4.6 CASO DE PRUEBA MOSTRAR LISTADOS

CASO DE PRUEBA INGRESAR USUARIOS	
PRECONDICIONES	
El Administrador deberá estar registrado para poder ingresar nuevos datos.	
PASOS	
ENTRADA	RESULTADO
1. El Administrador ingresa parámetro para mostrar.	2. Valida y muestra listados según parámetro.

Tabla XXIV Caso de Prueba Mostrar Listados

5.6.4.7 CASO DE PRUEBA INGRESAR PROYECTO DE INVESTIGACIÓN

CASO DE PRUEBA INGRESAR PROYECTO DE INVESTIGACIÓN	
PRECONDICIONES	
El Administrador deberá estar registrado para poder ingresar nuevos datos.	
PASOS	

ENTRADA	RESULTADO
1. El Administrador llena formulario	2. Datos validados
3. Acepta datos	4. Guarda información de proyecto.

Tabla XXV Caso de Prueba Ingresar proyecto de investigación

5.6.4.8 CASO DE PRUEBA INGRESAR EVENTO CIENTÍFICO

CASO DE PRUEBA INGRESAR PROYECTO CIENTÍFICO	
PRECONDICIONES	
Se haya ingresado al Sistema con éxito.	
PASOS	
ENTRADA	RESULTADO
1. Llena el formulario de registro de participación en el evento académico científico.	2. Valida los datos ingresados
3. Acepta los datos.	4. Guarda información.

Tabla XXVI Caso de Prueba Ingresar proyecto científico

5.6.4.9 CASO DE PRUEBA INGRESAR PUBLICACIONES

CASO DE PRUEBA REGISTRAR ESTUDIANTES	
PRECONDICIONES	
Se haya ingresado con éxito al sistema.	
PASOS	
ENTRADA	RESULTADO
1. El Administrador llena formulario de registro de Publicación.	2. Valida datos ingresados.
3. Acepta datos	4. Guarda información.

Tabla XXVII 1Caso de Prueba Ingrear publicaciones

5.7 PROPUESTA PARA LA IMPLANTACIÓN

Instalar el sistema Sigep, configurarlo y evaluarlo con el fin de obtener comentarios sobre el desempeño del mismo que permita rectificación y con esto mejorar su rendimiento.

5.7.1 PARTICIPANTES

En la etapa de instalación y configuración del sistema el grupo de desarrollo acompañará a la persona responsable del sistema.

Para dar buen uso al sistema y en caso de cualquier eventualidad, se entrega el manual de usuario a los Usuarios Finales del sistema.

5.7.2 REQUERIMIENTOS

A continuación se especifica los requisitos mínimos del hardware que debe sobre el cual deberá ser instalado el sistema Sigep.

EQUIPO	ELEMENTO	CAPACIDAD
Servidor	Procesador	Pentium 4 3.0 Ghz
	Memoria	1 GB
	Disco Duro	80 GB
	Tarjeta de Red	10/100 Mbps
Cliente	Procesador	Intel 1.73 Ghz
	Memoria	256 MB
	Disco Duro	40 GB
	Tarjeta de Red	10/100 Mbps

En la siguiente tabla se especifica en cambio los requerimientos mínimos de Software en el cual se instalará el sistema y en el cual su desempeño será el óptimo:

EQUIPO	TIPO DE APLICACIÓN	NOMBRE
Servidor	Sistema Operativo	Fedora Core 6
	Servidor de Aplicaciones	Jboss 4.0.2
	Máquina Virtual Java	JVM 1.5
Cliente de Escritorio	Sistema Operativo	Windows XP
	Máquina Virtual java	JVM 1.5
Cliente Web	Sistema Operativo	Windows XP

	Visor Web	Firefox
--	-----------	---------

5.8 COMPROBACIÓN DE LA HIPÓTESIS

Reingeniería de Sistemas Informáticos	Gestión de Proyectos
<p>Análisis de Inventario.- Este proceso consiste en el estudio de la antigüedad, importancia de la aplicación en el negocio y el proceso de mantenimiento actual, entre otros criterios, para estudiar la posible conveniencia de la reingeniería.</p>	<p>Análisis de Inventario.- Realizar un estudio acerca del posible análisis de reingeniería con la posibilidad de corregir algunos procesos ambiguos analizando sus conveniencias que tendría el sistema con la introducción de ciertos procesos.</p>
<p>Reestructuración de documentos.- En este proceso se puede optar por una de tres opciones: Evitar la documentación de los módulo estáticos que no van a sufrir cambios, documentar sólo lo que se va a modificar y documentar toda la información del sistema, si es que este es fundamental para el negocio</p>	<p>Reestructuración de documentos.- En esta fase se debe documentar todos los módulos que han sufrido cambios con el fin de obtener un respaldo teórico de la parte aplicativa.</p>
<p>Ingeniería Inversa.- En este proceso se extraen modelos de alto nivel de abstracción que ayuden a la comprensión de la aplicación para poder modificarla y que sirvan como punto de</p>	<p>Ingeniería Inversa.- En este proceso es necesario extraer los modelos de alto nivel el cual nos servirá para la utilización la reingeniería con el objetivo de que exista un buen funcionamiento del sistema a través del tiempo.</p>
	<p>Reestructuración de código.- En este proceso empieza tanto la reestructuración de la base de datos</p>

<p>partida para el siguiente proceso.</p> <p>Reestructuración de código.- La reestructuración de datos es una actividad de reingeniería a gran escala.</p> <p>Ingeniería hacia Adelante (Directa).- En el motor se insertaría el programa viejo, que lo analizaría, reestructuraría y después regeneraría la forma de exhibir los mejores aspectos de la calidad del software.</p>	<p>como de la interfaz gracias a este proceso el sistema tendrá muchos más beneficios que los tenía anteriormente.</p> <p>Ingeniería hacia Adelante (Directa).- En esta fase entraría el sistema antiguo para ser analizado y empezar a reestructurar todos sus módulos que sean necesarios con el fin de obtener mejoras para su mejor funcionamiento.</p>
<p style="text-align: center;">Metodología Cíclica</p>	<p style="text-align: center;">Gestión de Proyectos de Grado del CIPFIE-ESPOCH</p>
<p>Análisis de Inventario.- Todas las organizaciones de software deberán disponer de un inventario de todas sus aplicaciones.</p> <p>Reestructuración de documentos.- Una documentación escasa es la marca de muchos sistemas de información heredados.</p> <p>Ingeniería Inversa.- Una ingeniería inversa con éxito precede de una o más especificaciones de diseño y fabricación para el producto.</p> <p>Reestructuración de código.- Para llevar a cabo esta actividad, se analiza el código fuente mediante una</p>	<p>Análisis de Inventario.- Una vez culminado este sistema pasara a formar parte del inventario de DESITEL-ESPOCH</p> <p>Reestructuración de Documentos.- Con la culminación de este sistema se realizara una documentación completa debido a que el sistema anterior no estaba siendo utilizado por no cumplir con todos los requerimientos necesarios.</p> <p>Ingeniería Inversa.- en este proceso se realizara un análisis del programa anterior con el fin de obtener el nuevo</p>

<p>herramienta de reestructuración.</p> <p>Reestructuración de datos.- Un programa que posea una estructura de datos débil será difícil de adaptar y de mejorar</p> <p>Ingeniería hacia Adelante (Directa).- La ingeniería directa, que se denomina también renovación o reclamación, no solamente recupera la información de diseño de un software ya existente, sino que, además, utiliza esta información en un esfuerzo por mejorar su calidad global</p>	<p>sistema de alto nivel.</p> <p>Reestructuración de código.- A partir de un análisis del código del sistema anterior escogeremos los módulos validos para nuestro sistema.</p> <p>Reestructuración de datos.- En esta fase entraremos analizar la estructura de la base de datos con el fin de implantar nuevos requerimientos con el fin de mejorar el manejo del sistema actual.</p> <p>Ingeniería hacia Adelante (Directa).- En esta fase nos basaremos en el diseño del software anterior con el fin abstraer las partes más importante que nos ayudaran a establecer una mejora del sistema actual</p>
---	---

Tabla XXVIII Demostración de Hipótesis

CONCLUSIONES

- ✓ Al investigar las diferentes metodologías de Reingeniería de Sistemas Informáticos nos encontramos que existe poca información sobre las metodologías sin embargo hemos logrado investigar las siguientes: Metodología de Reestructuración, Metodología de Análisis de Opciones para Reingeniería (OAR), Metodología de Herradura y Metodología Cíclica.
- ✓ Según nuestro análisis en base a las características nos encontramos que algunas metodologías se centran en el diseño, arquitectura, algunas pretenden modificar la arquitectura global de programa. Otras sola se enfocan en modificar parte de los módulos del sistema, en fin cada metodología tiene su intencionalidad de ahí depende que se elija una u otra la que se adapte mejor a nuestra necesidad.
- ✓ Después de analizar y comparar las diferentes metodologías de Reingeniería de Sistemas Informáticos se puede concluir que una de las mejores metodologías dependiendo el proyecto es la Metodología Cíclica, ya que es una de las más completas que sigue paso a paso y en forma clara.

Metodología Cíclica:

- Análisis de Inventario
- Reestructuración de documentos
- Ingeniería Inversa
- Reestructuración de código
- Reestructuración de datos
- Ingeniería hacia Adelante (Directa)

- ✓ Gracias a la metodología escogida y que cumple con todas las características necesarias se ha obtenido una nueva versión del Sistema "Sigep", con lo cual hemos logrado cumplir con los requerimientos especificados.
- ✓ La reutilización de un Sistema Informático aparece como una alternativa para desarrollar software de manera eficiente, rápida y sobre todo productiva.

RECOMENDACIONES

- ✓ Se recomienda realizar un análisis previo a la toma de decisión sobre si realizar o no, una Reingeniería de un Sistema Informático ya que puede ser que se invierta dinero antes de obtener resultados y al final de la Reingeniería nos demos cuenta que no satisface 100% los requerimientos.
- ✓ Una vez elegido el camino de la reingeniería es importante también buscar la metodología adecuada que se ajuste a nuestro sistema, para obtener un sistema actual de calidad, mucho más eficiente y productivo que el sistema existente.

RESUMEN

El objetivo de la presente investigación fue el analizar las diferentes metodologías para realizar Reingeniería a Sistemas Informáticos, encontrar la que mejor se adapte a las necesidades y aplicarle en la Reingeniería del Sistema Sigep (Sistema de Gestión de proyectos de Grado), del CIFIE-ESPOCH.

Los métodos que se emplearon en el desarrollo de la tesis son el Método Inductivo y deductivo, debido a que se investigo las diferentes metodologías de reingeniería de sistemas informáticos con el fin de ver cuál de las metodologías es la conveniente para el Sistema de Gestión de Proyectos "SIGEP" que fue desarrollada para registrar los Proyectos de Grado, Proyectos de Docentes, Publicaciones y Eventos Científicos.

Para especificar, analizar y diseñar el Sistema Sigep antes de codificarlo se utilizó herramientas como Rational Rouse y StarUML, para la diagramación de la arquitectura se empleó Microsoft Visio, Dreamweaver para el desarrollo de la Interfaz de usuario (Pagina Web), se utilizó PostgreSQL como Lenguaje de Administración de base de Datos y por último utilizamos la Plataforma Netbeans como entorno de desarrollo integrado libre, con el lenguaje de programación Java.

Una vez aplicado la Reingeniería al Sistema de Gestión de Proyectos "SIGEP", se ha logrado una versión nueva del sistema informático que cumple con un 100% lo que es el registro de tesis de los estudiantes, al mismo tiempo cumple con los registros de Proyectos de Investigación, Publicaciones y Eventos de los

diferentes docentes en un 100% logrando así un notable incremento de productividad.

Concluimos que la reutilización de un Sistema Informático aparece como una alternativa para desarrollar software de manera eficiente, rápida y sobre todo productiva.

Se recomienda realizar un análisis previo a la toma de decisión sobre si realizar o no, una Reingeniería de un Sistema Informático ya que puede ser que se invierta dinero antes de obtener resultados y al final de la Reingeniería nos demos cuenta que no satisface 100% los requerimientos.

SUMARY

Computing systems reengineering methodology analysis, practical case, degree project management of Project investigation Centre of Computing and Electronics Faculty, "Escuela Superior Politecnica de Chimborazo" (CIPFIE-ESPOCH).

Inductive and Deductive methods were used in this paper because of the different methodologies of reengineering systems were researched in order to analyze which methodology is the most convenient for the Project management system "Sigep" that was developed to register the Degree Projects, Teachers, publications and scientific events.

Tools as rational and StarUML, were used to specify, analyze and design the "Sigep" System, Microsoft Visio was used for architecture diagramming, Dreamweaver for the development of user interface (Web Page), PostgreSQL, was used as Data basis managing language and Netbeans Platform was used as free integrated development environment, with Java programming.

A new computing system version was reached which registers 100% of students' papers, investigation projects, publications and events of different teachers by reaching a high production.

Degrees (GPG) appear as an alternative to develop software in an efficient, fast and effective way.

It is recommended to carry out an analysis before to make decisions to do or not the computing system reengineering in order to avoid wasting of money and bad results.

GLOSARIO

Abstracción:

Separar las características que interesan para un modelo de las irrelevantes.

Acoplamiento:

Medida de la complejidad de los interfaces de los módulos.

ActiveX:

Tipo de componente distribuido como binario.

Actividad:

Cada una de las partes en las que se divide una fase.

Actividades, diagrama:

Gráfico de UML que modela el comportamiento interno de una clase.

Actividades, diagrama de:

Gráfico UML que muestra el comportamiento y la participación de las clases en el mismo. Útil para flujos de trabajo.

Actor:

Persona o sistema que juega un papel en la interacción con un sistema.

Agregación:

Relación entre dos clases en la que una es parte de la otra.

Alcance:

Visibilidad de un atributo. Puede ser público, protegido o privado.

Almacén:

Es un conjunto de datos estático. Puede ser un archivo o una base de datos.

Almacenamiento de información:

Es una de las actividades o capacidades más importantes que tiene una computadora, ya que a través de esta propiedad el sistema puede recordar la información guardada en la sección o proceso anterior.

Alto nivel, lenguajes:

Son lenguajes con alto nivel de abstracción. Hacen lo mismo que un lenguaje de bajo nivel pero con menos líneas.

Ámbito:

Número de veces que existe el objeto. De instancia: Una por cada instancia de la clase. Archivador: Una única vez, como las variables *static* de Java.

Análisis:

Etapa donde el problema es la comprensión de las necesidades.

Análisis crítico:

Actividad que se realiza desde el principio del proyecto para localizar los problemas de alto riesgo.

Análisis de valores límite:

Técnica de generación de casos de prueba conjunta con particiones de equivalencia que genera casos de prueba cercanos al máximo o mínimo tamaño de un array o al máximo o mínimo número de iteraciones de un bucle.

Asociaciones:

Relación que se establece entre dos o más clases.

Atributo:

Variable de una clase.

Auditoría:

Examen del sistema preferiblemente por parte de una organización no involucrada en el proceso de desarrollo. Tipos: regulares, especiales, sorpresa, de seguimiento, financiera, operacional, de gestión e informática.

Base de Datos:

Es cuando se accede a la información mediante el software que es parte integral del funcionamiento del sistema. Al hablar del conjunto almacenado de datos, se hace referencia a la base de datos, los cuales son utilizados por los sistemas de información permitiendo el respaldo rápido y seguro de la información.

Brainstorming:

Técnica poco estructurada de obtención de requisitos en una reunión.

Caja blanca, pruebas:

Pruebas que tienen en cuenta la estructura lógica del programa.

Caja de cristal, pruebas:

Ver caja blanca.

Caja negra, pruebas:

Pruebas que tienen en cuenta las especificaciones.

Calificación:

Atributo que rebaja la cardinalidad de una parte de la asociación entre dos clases.

Cardinalidad:

Número de entidades que están asociadas con otra(s) en una relación.

Cascada:

Ciclo de vida lineal.

CASE:

(*Computer Aided Software Engineering*) Herramienta para automatizar o facilitar parte o todo el trabajo mecánico (no creativo) de parte o todo el ciclo de vida.

Casos de prueba:

Conjunto de entradas y de salidas esperadas para esas entradas.

Casos de uso:

Método de análisis para una interacción del sistema con el usuario u otro sistema.

Ciclo de vida:

Conjunto de etapas por las que pasa un sistema informático desde su concepción hasta su retirada.

Clase:

Estructura de datos con las operaciones relativas a esos datos.

Clases, diagrama:

Principal tipo de diagrama en UML que muestra las clases y las relaciones que tienen.

Clases de equivalencia:

Agrupaciones de las entradas iguales a efectos de las pruebas.

Cliente-servidor:

Arquitectura de sistemas distribuidos con una parte que funciona como interfaz (cliente) y otra que gestiona recursos y realiza el trabajo (servidor).

Codificación:

Etapas del ciclo de vida en la que se escribe el código fuente.

Código heredado:

(*legacy code*) Código de aplicaciones antiguas posiblemente muy remendado y sin documentación.

Cohesión:

Característica de un módulo. Un módulo tiene cohesión si sus tareas son independientes del resto del sistema pero están relacionadas entre sí.

Colaboración, diagrama:

Indica interacciones entre objetos a través de los enlaces que hay entre ellos. Similar al diagrama de secuencia.

Complejidad ciclomática:

Medida de la complejidad de un programa, igual al número de caminos independientes que tiene.

Componentes:

Software pensado para ser reutilizable. Consta de código y datos.

Componentes, diagrama:

Gráfico de UML que muestra la organización y dependencias entre componentes.

Composición:

Es un tipo de agregación en la que cada componente pertenece a un todo y sólo a ese.

Congruencia:

Un método es congruente si tiene un nombre similar a métodos similares, condiciones, orden de argumentos, valor proporcionado y condiciones de error.

Contexto, diagrama:

Representación del sistema y entidades externas que interactúan con él.

CORBA:

(*Common Object Request Broker Architecture*) Sistema distribuido multiplataforma basado en objetos. Cada uno puede ser cliente y servidor.

CRC:

Mecanismo para representar clases e interacciones entre ellas.

Crisis del software:

La consecuencia de escribir código sin plantear seriamente metodologías para el análisis, diseño y mantenimiento.

CVS:

(*Concurrent Versioning System*) Almacenamiento de ficheros con capacidad de trazar las aportaciones de cada persona y la historia de cada archivo.

Defecto:

El sistema hace algo de forma equivocada. Es la consecuencia de un error y se manifiesta a través de un fallo.

DFD:

(*Diagrama de Flujo de Datos*) Herramienta del diseño estructurado para modelar la transformación o transacción de la información.

Diccionario de datos:

Descripción detallada de los flujos de datos

Diseño:

Etapas del ciclo de vida en la que se divide el sistema en subsistemas y posteriormente se decide cómo serán las estructuras de datos y algoritmos.

Distribución, diagrama:

Gráfico de UML que refleja la organización del hardware.

Encapsulamiento:

Propiedad por la que un objeto o módulo proporciona al exterior las funciones necesarias para su manejo y no los detalles internos.

Entidad-Relación:

Modelo de datos del diseño estructurado.

Entidades externas:

Personas o sistemas que interactúan con el sistema.

Entrada de información:

Es el proceso mediante el cual el Sistema de Información toma los datos que requiere para procesar la información. Las entradas pueden ser manuales o automáticas.

Entrada/Salida, pruebas:

Ver caja negra.

Entrevistas:

Técnica de obtención de requisitos que consiste en hablar con un usuario.

Error:

Equivocación de un desarrollador en cualquiera de las fases. Produce uno o más defectos.

Especificación:

Traducción de los requisitos del análisis a un documento que sirva para empezar el diseño.

Especificación de control:

Muestran como a partir de señales de control se activan o desactivan procesos del diagrama de flujo de control asociado.

Especificación de procesos:

Concretar en pseudocódigo o en algún lenguaje de programación cómo es un módulo que ya no se puede descomponer más.

Espiral, ciclo de vida:

Ciclo de vida con varias iteraciones. Hace énfasis en el control del riesgo. Típico en la orientación a objetos.

Estados, diagrama:

Representación del comportamiento interno de un sistema.

Esteriotipos:

Forma de dar nombre a un elemento que no forma parte del estándar de UML.

Estructurales, pruebas:

Ver pruebas de caja blanca.

Estructuras, diagrama:

Método de descomposición funcional donde el bloque básico es el módulo.

Etapas:

Cada una de las partes de las que se compone el ciclo de vida.

Factor de calidad:

Forma de determinar la calidad de un software. Están divididos en función de criterios de operación, revisión y transición. Cada uno se obtiene a partir de varias métricas.

Fallo:

Manifestación de un defecto.

Flujo de datos:

Representación de datos que se mueven entre procesos o entidades externas.

Flujo de control:

Diagrama que informa sobre cuando y en que orden ocurren los sucesos.

Framework:

Conjunto integrado de componentes que colaboran para proporcionar una arquitectura reutilizable para una familia de aplicaciones.

Fuente, ciclo de vida:

Ciclo de vida para sistemas orientados a objetos.

Funcionales, pruebas:

Ver caja negra.

Generalización:

Ver herencia.

Grafo de flujo:

Representación de un programa usada para el diseño de casos de prueba estructurales.

Herencia:

Mecanismo por el que unas clases mantienen los mismos métodos y propiedades de otras a las cuales amplían.

HIPO, diagramas:

(*Hierarchy Input Process Output*) Diagrama de diseño que muestra entradas, salidas y funciones.

IDL:

(*Interface Definition Language*) Lenguaje formal para expresar interfaces.

Incremental:

Ciclo de vida derivado del de cascada pero con iteraciones para implementar distintas partes del sistema.

Ingeniería inversa:

Análisis de un producto construido para comprender sus principios de diseño.

Indicador:

Medida numérica para comprobar algo de un modo objetivo. Los hay económicos, financieros, de ocupación laboral y de gestión.

Información:

La información es un fenómeno que proporciona significado o sentido a las cosas, e indica mediante códigos y conjuntos de datos, los modelos del pensamiento humano. La información por tanto, procesa y genera el conocimiento humano.

Inspecciones:

Revisión del código sin ejecutarlo. Muy estructurada y realizada por un equipo.

Interfaz:

Conjunto de servicios proporcionados por una clase.

JAD:

(*Joint Application Design*) Técnica muy estructurada de obtención de requisitos en una reunión.

Java Beans:

Componentes escritos en java.

Jackson, diagramas:

Produce una especificación del programa a partir de una especificación de las entradas y las salidas, que deben ser secuenciales.

JRP:

(*Joint Requirements Planning*) Subconjunto del JAD para obtención de requisitos de alto nivel.

Lenguaje estructurado:

Pseudocódigo para escribir las especificaciones de proceso.

Línea base:

Puntos de referencia en la gestión de configuración.

Mantenimiento:

Etapa final del ciclo de vida. Consume la mayor parte de los recursos.

Mantenibilidad:

Cualidad de una aplicación que hace que el mantenimiento sea más fácil.

Metodología:

Modo sistemático de fabricar un producto.

Métricas:

Medidas numéricas sobre alguna cualidad de un programa.

Modelo:

Abstracción de la realidad donde se omite lo no esencial.

Módulo:

Cada una de las partes con entidad propia en las que se divide un sistema.

Modularidad:

Propiedad del software. Un programa es modular si está dividido en partes con tareas definidas.

Multiplicidad:

Número de elementos que están en los lados de una relación.

Navegabilidad:

Posibilidad de moverse de una clase a otra utilizando las relaciones que hay entre ellas.

Normalización:

Algoritmos que convierten una base de datos en otra equivalente pero sin anomalías de inserción, borrado y modificación, más eficiente y menos redundante.

Notas:

Explicaciones de los elementos de notación de UML.

Objeto:

Instancia de una clase.

OCL:

(*Object Constraint Language*) Lenguaje formal para expresar restricciones.

Paquetes:

Conjunto de clases relacionadas entre si.

Patrones:

Prototipos de soluciones para problemas conocidos. Los hay de análisis, arquitectónicos, de diseño, de codificación y de organización.

Persistencia:

Característica de un objeto que se puede escribir en disco para posteriormente ser recuperado.

Polimorfismo:

Capacidad de un conjunto de objetos de responder a un mensaje con el mismo nombre o a un objeto de poder responder a un mismo mensaje con distintos parámetros.

Privado:

Tipo de alcance más restrictivo posible. Una propiedad de este tipo solo puede ser accedida dentro de la clase en la que está declarada.

Proceso:

Subsistemas o funciones en las que se divide el sistema.

Protegido:

Tipo de alcance que restringe el acceso a la clase en la que se ha declarado y a sus descendientes.

Prototipo:

Versión reducida de la aplicación final. Se construye para obtener requisitos o para partiendo de él hacer incrementos hasta el sistema final.

Pruebas:

Batería de test que el sistema tiene que superar para ser considerado operativo.

Público:

Tipo de alcance que permite el acceso a cualquier otra clase.

Reducción de riesgos, cascada con:

Ciclo de vida cascada con iteraciones en el análisis y diseño global.

Reingeniería:

Reconstrucción completa de una aplicación antigua utilizando técnicas modernas.

Requisitos funcionales:

Lo que el sistema hace para el usuario.

Requisitos no funcionales:

Características del sistema (fiabilidad, mantenibilidad, etc). Se tienen en cuenta en la fase de diseño.

Responsabilidades:

Descripción de lo que hace una clase.

Revisión:

Las distintas versiones de un elemento software que van surgiendo.

Salida de información:

La salida es la capacidad de un Sistema de Información para sacar la información procesada o bien datos de entrada al exterior. Es importante aclarar que la salida de un Sistema de Información puede constituir la entrada a otro Sistema de Información o módulo.

Sashimi, ciclo:

Ciclo de vida en cascada con solapamiento entre las fases.

Secuencias, diagrama:

Gráfico de UML que muestra los mensajes intercambiados entre objetos teniendo en cuenta el tiempo.

Sistema:

Conjunto de elementos que cooperan entre si para proporcionar servicios.

Sistemas, ingeniería de:

Herramientas y metodologías para crear sistemas.

Software, ingeniería del:

Herramientas y metodologías para crear software.

Tarea:

Una etapa (p.ej. el diseño) se descompone en tareas. Una tarea puede realizarse a lo largo de varias etapas, por ejemplo la documentación.

Tablas de decisión:

Forma de escribir una especificación de procesos cuando hay muchas variables y en función de su valor se toman decisiones.

Transición de estados:

Diagrama de UML que refleja los cambios que experimenta una clase como si fuera un autómata finito.

UML:

(*Unified modeling language*) Notación *de facto* para el modelado de sistemas orientados a objetos.

V, ciclo de vida en:

Ciclo de vida similar al de cascada pero que toma en consideración el nivel de abstracción de cada una. El análisis valida el mantenimiento y el diseño verifica las pruebas.

Validación:

Comprobación de que el sistema cumple con las especificaciones funcionales y no funcionales.

Variante:

Versión de un elemento software que coexiste con otra con algunas diferencias.

VCL:

(*Visual Component Library*) Componente gráfico.

Verificación:

Comprobación de que el sistema funciona correctamente.

Versión:

Un elemento software en un momento dado.

Warnier, diagramas:

Representación jerárquica de la estructura de los programas y estructuras de datos.

BIBLIOGRAFÍA

1. TESIS DE REINGENIERÍA DE SISTEMAS

<http://bibdigital.epn.edu.ec/bitstream/15000/2115/1/CD-2901.pdf>

2011-04-23

2. REINGENIERÍA DE SOFTWARE

<http://www.buenastareas.com/ensayos/Reingenieria-De-Software/519168.html>

2011-02-12

3. MARCO TEÓRICO

http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/flores_r_mi/capitulo2.pdf

2011-02-12

4. REINGENIERÍA DE SOFTWARE UN CAMINO O EL CAMINO

http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/flores_r_mi/capitulo2.pdf

2011-02-12

5. REINGENIERÍA DE SISTEMAS

http://www.concatel.com/pfw_files/cma/Web_cvteam/Docs/casestudies/es/cs_interfacom.pdf

2011-04-25

6. REINGENIERÍA DE SOFTWARE DE SISTEMAS

<http://www.elrinconcito.com/articulos/Reingenieria/Articulo.htm>

2011-04-15

7. TÉCNICAS Y HERRAMIENTAS DE REINGENIERÍA DE SOFTWARE

<http://www.everyoneweb.es/WA/DataFilesutvmtic/Reingenieria.pdf>

2011-02-12

8. GREDOS, METODOLOGÍA DE REINGENIERÍA

<http://gredos.usal.es/jspui/bitstream/10366/21762/1/DPTOIA-IT-2004-003.pdf>

2011-03-04

9. IISCI, INGENIERÍA INVERSA

[http://www.iiisci.org/journal/CV\\$/risi/pdfs/X581YP.pdf](http://www.iiisci.org/journal/CV$/risi/pdfs/X581YP.pdf)

2011-02-12

10. LACCEI.ORG, HERRAMIENTA ISOXPRT

<http://www.laccei.org/LACCEI2009-Venezuela/p83.pdf>

2011-02-12

11. MONOGRAFÍAS, REINGENIERÍA

<http://www.monografias.com/trabajos/reingenieria/reingenieria.shtml>

2011-04-13

12. REINGENIERÍA DE PROGRAMACIÓN

<http://profesores.fi-b.unam.mx/heriolg/Reipro10.pdf>

2011-02-12

13. REINGENIERÍA PARA SISTEMAS DE EVALUACIÓN TEÓRICA

<http://www.revistaciencias.com/publicaciones/EEAAklAEIFxpOUEqJu.php>

2011-03-18

14. CAPÍTULO 27 – REINGENIERÍA

<http://www.scribd.com/doc/32165548/CAPITULO-27-REINGENIERIA>

2011-02-12

15. REINGENIERÍA

<http://www.slideshare.net/darwinnano/reingenieria-476282>

2011-02-12

16. EVOLUCIÓN Y MANTENIMIENTO DE SOFTWARE

<http:// triana.escet.urjc.es/aspf/MasterURJC-SAPF-Tema1-08-09.pdf>

2011-08-12

17. INTRODUCCIÓN A LA REINGENIERÍA

[http://www.virtual.unal.edu.co/cursos/sedes/manizales/4100010/Lecciones/
Cap1/Reingenieria.htm](http://www.virtual.unal.edu.co/cursos/sedes/manizales/4100010/Lecciones/Cap1/Reingenieria.htm)

2011-02-12

ANEXOS

MANUAL DE USUARIO

Ingresar al Sistema

Para ingresar al sistema el usuario debe ingresar su Login y su Password caso contrario no podrá ingresar al sistema.



Proyecto Estudiante

Ingreso Nuevo Proyecto de Estudiante

Para ingresar un nuevo proyecto del estudiante hacer un clic en **Proyecto Estudiante** de la barra de menú seleccionamos **Proyecto** y escogemos la opción **Ingresar** como nos muestra en la siguiente pantalla.



En la presente nos muestra todos los campos que se debe llenar para ingresar un proyecto, como nos muestra en la siguiente pantalla.

Proyecto Estudiante | Proyecto Investigacion | Eventos | Publicaciones | Docentes | Ayuda

PRESENTACION DE ANTEPROYECTOS

Fecha:	<input type="text"/>	Titulo proyecto:	<input type="text"/>	Estado:		
Area:	<input type="text"/>	Resolución:	<input type="text"/>	En Proceso	Aprobado	Rechazado
Escuela:	SISTEMAS	Integrantes:	<input type="radio"/> 1 <input type="radio"/> 2	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Estudiante 1	
Cédula:	<input type="text"/> enviar
Nombres: <input type="text"/>	
Director:	NOMBRE
Cédula:	<input type="text"/> enviar
Escuela:	Lic. en Diseño Grafico

Tribunal de analisis	
Cédula:	<input type="text"/> enviar
Escuela:	Lic. en Diseño Grafico
Nombres:	DELEGADO 1
Cédula:	<input type="text"/> enviar
Escuela:	Lic. en Diseño Grafico
Nombres:	DELEGADO 2

Para el ingreso de la fecha ingresamos dando clic en el icono que se encuentra alado del campo de la fecha donde nos desplegara un menú de la fecha como nos muestra en la siguiente pantalla.

Fecha:	<input type="text"/>	Titulo	<input type="text"/>																																																																													
Area:	<input type="text"/>	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td colspan="7" style="text-align: center;">Abril, 2012</td> </tr> <tr> <td colspan="7" style="text-align: center;">Hoy</td> </tr> <tr> <td>sem</td> <td>Lun</td> <td>Mar</td> <td>Mié</td> <td>Jue</td> <td>Vie</td> <td>Sáb</td> <td>Dom</td> </tr> <tr> <td>13</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td>1</td> </tr> <tr> <td>14</td> <td>2</td> <td>3</td> <td>4</td> <td>5</td> <td>6</td> <td>7</td> <td>8</td> </tr> <tr> <td>15</td> <td>9</td> <td>10</td> <td>11</td> <td>12</td> <td>13</td> <td>14</td> <td>15</td> </tr> <tr> <td>16</td> <td>16</td> <td>17</td> <td>18</td> <td>19</td> <td>20</td> <td>21</td> <td>22</td> </tr> <tr> <td>17</td> <td>23</td> <td>24</td> <td>25</td> <td>26</td> <td>27</td> <td>28</td> <td>29</td> </tr> <tr> <td>18</td> <td>30</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td colspan="7" style="text-align: center;">Acerca del calendario</td> </tr> </table>		Abril, 2012							Hoy							sem	Lun	Mar	Mié	Jue	Vie	Sáb	Dom	13							1	14	2	3	4	5	6	7	8	15	9	10	11	12	13	14	15	16	16	17	18	19	20	21	22	17	23	24	25	26	27	28	29	18	30							Acerca del calendario						
Abril, 2012																																																																																
Hoy																																																																																
sem	Lun	Mar	Mié	Jue	Vie	Sáb	Dom																																																																									
13							1																																																																									
14	2	3	4	5	6	7	8																																																																									
15	9	10	11	12	13	14	15																																																																									
16	16	17	18	19	20	21	22																																																																									
17	23	24	25	26	27	28	29																																																																									
18	30																																																																															
Acerca del calendario																																																																																
Escuela:	SISTEMAS																																																																															
Estudiante 1																																																																																
Cédula:	<input type="text"/>																																																																															
Nombres: <input type="text"/>																																																																																

Escoger el número de integrantes que realizaran el proyecto en este si escogemos dos integrantes nos aparecerá los dos campos para llenar a los dos integrantes del proyecto como nos muestra en la siguiente pantalla.

Escuela:	SISTEMAS	Integrantes:	<input type="radio"/> 1 <input checked="" type="radio"/> 2			
----------	----------	--------------	--	--	--	--

Estudiante 1		Estudiante 2	
Cédula:	<input type="text"/> enviar	Cédula:	<input type="text"/> enviar
Nombres: <input type="text"/>		Nombres: <input type="text"/>	

Para ingresar los integrantes solo es necesario poner el numero de cedula y presionar sobre el acceso directo de enviar en donde automáticamente se pondrá el nombre completo del estudiante en el campo nombres.

Estudiante 1	
Cédula:	060311880-3 enviar
Nombres:	

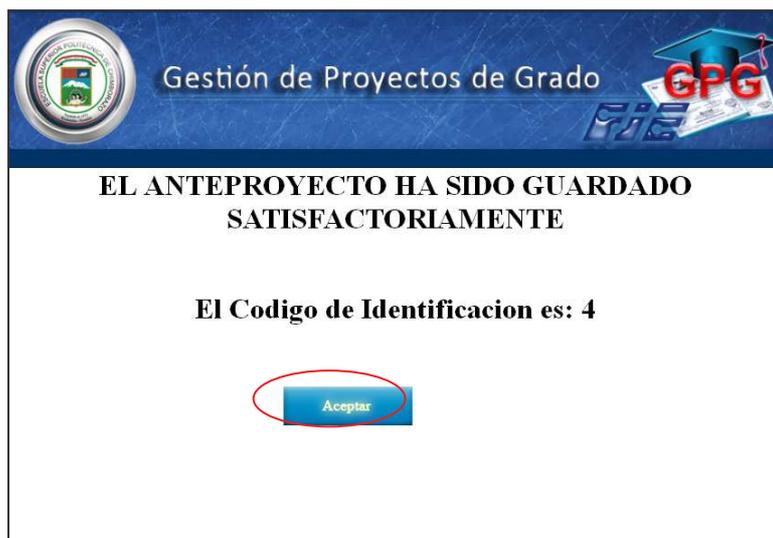
Para el ingreso del Director y del tribunal es necesario poner el numero de cedula y escoger a que escuela pertenece una vez ingresado el numero de cedula y escogido a que escuela pertenece damos clic en el acceso directo enviar y automaticamente se ingresar el nombre completo del docente.

Director	NOMBRE	Cédula:	060311880-6 enviar	Escuela:	Ing. en Sistemas
Tribunal de analisis					
Cédula:	060321880-9 enviar	Nombres	DELEGADO 1		
Escuela:	Ing. en Sistemas				
Cédula:	060254826-6 enviar	Nombres	DELEGADO 2		
Escuela:	Ing. en Sistemas				

Para culminar con el ingreso de un nuevo proyecto y dejar almacenado damos clic en el botón **aceptar** como nos muestra en la siguiente pantalla.



Una vez presionado aceptar para guardar el nuevo nos aparecerá la siguiente pantalla en donde nos indicara el código del proyecto y para retornar a la pantalla principal damos clic en aceptar como nos muestra en la siguiente pantalla



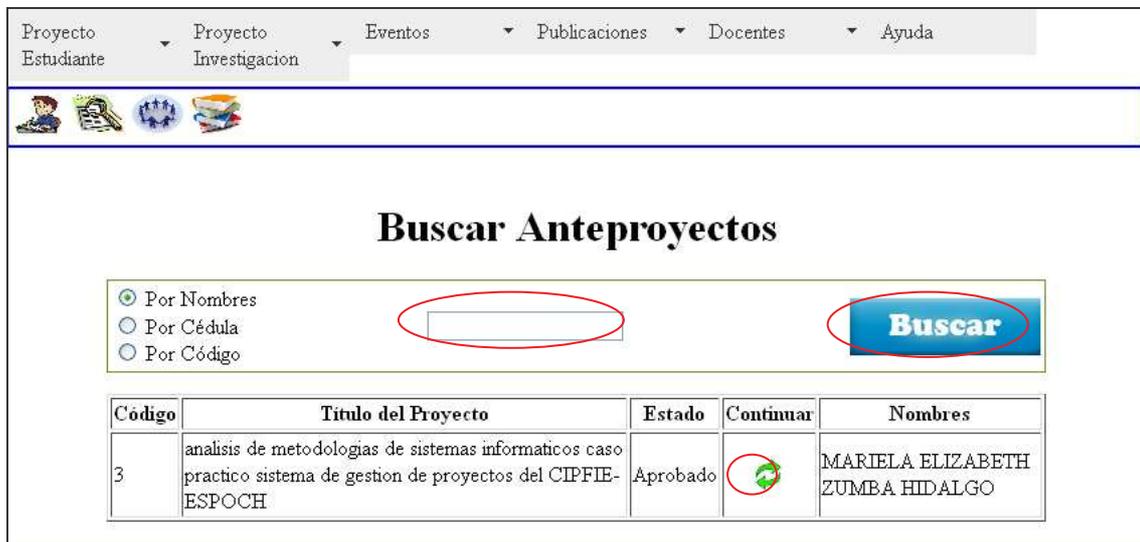
Actualizar un Proyecto

Para actualizar un proyecto presionamos sobre **proyecto estudiante** seleccionamos **Proyecto** y escogemos la opción **Actualizar** como nos muestra en la siguiente pantalla.



The screenshot shows a web interface with a navigation menu on the left. The 'Proyecto Estudiante' menu is open, and the 'Actualizar' option is highlighted. The main content area is titled 'Buscar Anteproyectos' and contains a search form with input fields for 'Nombres', 'Cédula', and 'Código', and a 'Buscar' button. Below the search form is a table with columns: Código, Título del Proyecto, Estado, Área, and Nombres.

En la presente pantalla el usuario podrá ingresar actualizar el proyecto por medio del nombre, cedula o código del integrante y presionamos buscar en donde me aparecerá la información necesaria para ver si es correcto el proyecto que se está buscando para actualizar presionamos sobre el icono continuar.



The screenshot shows the 'Buscar Anteproyectos' page with search results. The search form has radio buttons for 'Por Nombres', 'Por Cédula', and 'Por Código'. The search results table is as follows:

Código	Título del Proyecto	Estado	Continuar	Nombres
3	analisis de metodologias de sistemas informaticos caso practico sistema de gestion de proyectos del CIPFIE-ESPOCH	Aprobado		MARIELA ELIZABETH ZUMBA HIDALGO

Una vez realizado los cambios correspondientes presionamos aceptar para que quede almacenado los cambios como nos muestra en la siguiente pantalla

Proyecto Estudiante	Proyecto Investigacion	Eventos	Publicaciones	Docentes	Ayuda
---------------------	------------------------	---------	---------------	----------	-------

ACTUALIZAR ANTEPROYECTO

Fecha Aprobacion:	2012-04-09 00:00:00.0	Titulo proyecto:	analisis de metodologias de sistemas informaticos	Estado:	<input type="radio"/> En Proceso <input checked="" type="radio"/> Aprobado <input type="radio"/> Rechazado
Area:	Desarrollo				
Escuela:	sistemas				

Integrantes:

Estudiantes	Cédula:	060311880-3	Nombres:	RAUL HUMBERTO CUZCO NARANJO
	Cédula:	060371660-6	Nombres:	MARIELA ELIZABETH ZUMBA HIDALGO

Tribunal de direccion

Director	Cédula:	111111111-1	Nombres:	Inf. Raúl Rosero
Miembro	Cédula:	222222222-2	Nombres:	Ing. Landy Ruiz
Miembro	Cédula:	333333333-3	Nombres:	Dr. Geovanny Vallejo

Matricular Proyecto

Para ingresar a matricular un proyecto presionamos sobre **proyecto estudiante** seleccionamos **Matricular Proyecto** como nos muestra en la siguiente pantalla.

Proyecto Estudiante	Proyecto Investigacion	Eventos	Publicaciones	Docentes	Ayuda
---------------------	------------------------	---------	---------------	----------	-------

Buscar Anteproyectos para Matricular

Nombres	<input type="text"/>	<input type="button" value="Buscar"/>
Cédula	<input type="text"/>	

go	Titulo del Proyecto	Estado	Continuar	Nombres
----	---------------------	--------	-----------	---------

Para poder ingresar la matricula del proyecto por medio del nombre, cedula del integrante y presionamos buscar en donde me aparecerá la información necesaria para ver si es correcto el proyecto que se está buscando para matricular presionamos sobre el icono continuar.

Proyecto Estudiante | Proyecto Investigacion | Eventos | Publicaciones | Docentes | Ayuda

Buscar Anteproyectos para Matricular

Por Nombres

 Por Cédula

Código	Título del Proyecto	Estado	Continuar	Nombres
3	analisis de metodologias de sistemas informaticos caso practico sistema de gestion de proyectos del CIPFIE-ESPOCH	Aprobado		RAUL HUMBERTO CUZCO NARANJO
4		null		

Para ingresar la matricula del estudiante el usuario podrá cambiar el miembro de la tesis dando clic sobre las opciones que nos muestra a lado de los nombres de los miembros, después del escoger el miembro ingresamos la matricula de la tesis y para culminar el ingreso de la matricula presionamos sobre el botón aceptar como nos muestra en la siguiente pantalla.

Proyecto Estudiante | Proyecto Investigacion | Eventos | Publicaciones | Docentes | Ayuda

PROYECTO

Fecha Aprobacion: 2012-04-09 | Titulo proyecto: analisis de metodologias de sistemas informaticos caso practico sistema de gestion de proyectos del CIPFIE-ESPOCH | Estado: En Proceso Aprobado Rechazado

Area: Desarrollo

Integrantes:

Estudiante		Estudiante	
Cédula:	060311880-3	Nombres:	RAUL HUMBERTO CUZCO NARANJO
Cédula:	060371660-6	Nombres:	MARIELA ELIZABETH ZUMBA HIDALGO

Tribunal de dirección

Director	Cédula:	Nombres:
	111111111-1	Inf. Raúl Rosero
Miembro	Cédula:	Nombres:
	222222222-2	Ing. Landy Ruiz
Miembro	Cédula:	Nombres:
	333333333-3	Dr. Geovanny Vallejo

Resolución: CD-FIE-020-2011 | Fecha Inicio: 2012-04-10 00:00:00.0 | Fecha Final: 2013-04-10 00:00:00.0
 Resolución: CD-FIE-020-2012 | Fecha Inicio: 2012-04-30 00:00:00.0 | Fecha Final: 2012-10-30 00:00:00.0

Prórroga

Resolución	Fecha Inicio	Fecha Fin

Tesis Escrita

Para ingresar a Tesis Escrita de un proyecto presionamos sobre **proyecto estudiante** seleccionamos **Tesis Escrita** como nos muestra en la siguiente pantalla.

Proyecto Estudiante | Proyecto | Eventos | Publicaciones | Docentes | Ayuda
 Proyecto | Investigacion

Proyecto | Buscar Proyecto | Matricular Proyecto | Tesis Escrita | Tesis Oral | Listados

Buscar Anteproyectos para Matricular

Nombres **Buscar**
 Cédula

Código	Titulo del Proyecto	Estado	Continuar	Nombres

Para poder imprimir las Actas de la tesis Escrita del proyecto se lo puede realizar la búsqueda por medio del nombre, cedula del integrante y presionamos buscar en donde me aparecerá la información necesaria para ver si es correcto el proyecto que se está buscando para imprimir actas presionamos sobre el icono continuar.

Proyecto Estudiante | Proyecto | Eventos | Publicaciones | Docentes | Ayuda
 Proyecto | Investigacion

Buscar Proyectos para Defensa Escrita

Por Nombres **Buscar**
 Por Cédula

Código	Titulo del Proyecto	Estado	Continuar	Nombres
3	analisis de metodologias de sistemas informaticos caso practico sistema de gestion de proyectos del CIPFIE-ESPOCH	Aprobado		RAUL HUMBERTO CUZCO NARANJO
4		null		

Para imprimir las actas presionamos sobre el acceso directo Acta Escrita y a la vez podemos ingresar el delegado de la tesis si no se desea ingresar el miembro de la tesis podemos presionar el botón cancelar para no ingresar el miembro como nos muestra en la pantalla.

Proyecto Estudiante | Proyecto Investigacion | Eventos | Publicaciones | Docentes | Ayuda

TESIS ESCRITA

Fecha Aprobacion:	2012-04-09	Titulo proyecto:	analisis de metodologias de sistemas informaticos caso practico sistema de gestion de proyectos del CIPFIE-ESPOCH
Area:	Desarrollo		

Integrantes:

Estudiante		Estudiante	
Cédula:	060311880-3	Nombres:	RAUL HUMBERTO CUZCO NARANJO
Cédula:	060371660-6	Nombres:	MARIELA ELIZABETH ZUMBA HIDALGO

Tribunal de dirección

	Cédula:	Nombres:	
Director	111111111-1	Inf. Raúl Rosero	Acta Escrita
Miembro	333333333-3	Dr. Geovanny Vallejo	Acta Escrita
Miembro	444444444-4	Ing. Eduardo Villa	Acta Escrita

Asignar Delegado del Decano:

Delegado	Cédula:	Escuela:	Nombre	Delegado
		Ing. Diseño Grafico		

Aceptar
Cancelar

Tesis Oral

Para ingresar a Tesis Oral de un proyecto presionamos sobre **proyecto estudiante** seleccionamos **Tesis Oral** como nos muestra en la siguiente pantalla.

Proyecto Estudiante | Proyecto Investigacion | Eventos | Publicaciones | Docentes | Ayuda

Proyecto | Buscar Proyecto | Matricular Proyecto | Tesis Escrita | **Tesis Oral** | Listados

Buscar Proyectos para Defensa Escrita

Nombres: Buscar

Cédula:

go	Titulo del Proyecto	Estado	Continuar	Nombres
----	---------------------	--------	-----------	---------

Para poder imprimir las Actas de la tesis Oral del proyecto se lo puede realizar la búsqueda por medio del nombre, cedula del integrante y presionamos buscar en donde me aparecerá la información necesaria para ver si es correcto el proyecto que se está buscando para imprimir actas presionamos sobre el icono continuar.

Proyecto Estudiante Proyecto Investigacion Eventos Publicaciones Docentes Ayuda

Buscar Proyectos para Defensa Oral

Por Nombres Por Cédula

Buscar

Código	Título del Proyecto	Estado	Continuar	Nombres
3	analisis de metodologias de sistemas informaticos caso practico sistema de gestion de proyectos del CIPFIE-ESPOCH	Aprobado		RAUL HUMBERTO CUZCO NARANJO
4		null		

Para imprimir las actas presionamos sobre el acceso directo Acta Escrita y a la vez podemos ingresar el delegado de la tesis si no se desea ingresar el miembro de la tesis podemos presionar el botón cancelar para no ingresar el miembro como nos muestra en la pantalla.

Proyecto Estudiante Proyecto Investigacion Eventos Publicaciones Docentes Ayuda

TESIS ORAL

Fecha Aprobacion: 2012-04-09 Titulo proyecto: analisis de metodologias de sistemas informaticos caso practico sistema de gestion de proyectos del CIPFIE-ESPOCH

Area: Desarrollo

Integrantes:

Estudiante		Estudiante	
Cédula:	060311880-3	Nombres:	RAUL HUMBERTO CUZCO NARANJO
Cédula:	060371660-6	Nombres:	MARIELA ELIZABETH ZUMBA HIDALGO

Tribunal de dirección

Director	Cédula:	111111111-1	Nombres:	Inf. Raúl Rosero	Acta Escrita
Miembro	Cédula:	333333333-3	Nombres:	Dr. Geovanny Vallejo	Acta Escrita
Delegado	Cédula:	444444444-4	Nombres:	Ing. Eduardo Villa	Acta Escrita

Aceptar
Cancelar

Listado de Proyectos

Para ingresar a Listados de todas las tesis presionamos sobre **proyecto estudiante** seleccionamos **Listados** como nos muestra en la siguiente pantalla.



Proyecto Investigación

Ingreso Nuevo Proyecto de Investigación

Para ingresar un nuevo proyecto del Investigación hacer un clic en **Proyecto Investigación** de la barra de menú seleccionamos **Proyecto** y escogemos la opción **Ingresar** como nos muestra en la siguiente pantalla.



En la presente nos muestra todos los campos que se debe llenar para ingresar un proyecto, como nos muestra en la siguiente pantalla.

PROYECTOS DOCENTES

Datos Generales:							
Fecha:	<input type="text"/>	Evolución:	<input type="text"/>				
Título:	<input type="text"/>						
Director Proyecto:	<input type="text"/>	Subdirector Proyecto:	<input type="text"/>				
Área:	<input type="text"/>		Estado del Proyecto:				
Efecto de Impacto:	Área de Investigación:	Lineas de Investigación:	Formulado:	En Ejecución:	Ejecutado:		
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>		
<input type="text"/>	<input type="text"/>	<input type="text"/>					
Financiamiento							
Institucional:	<input type="radio"/>	Personal:	<input type="radio"/>	Ambo:	<input type="radio"/>	Otro:	<input type="radio"/>
Integrantes							
Docentes:	<input type="text"/>	Estudiantes:	<input type="text"/>				
Detalle Formal del Proyecto							
Introducción							
<div style="border: 1px solid black; width: 100%; height: 100%;"></div>							
Objetivos:							
<div style="border: 1px solid black; width: 100%; height: 100%;"></div>							
Justificación							
<div style="border: 1px solid black; width: 100%; height: 100%;"></div>							
Presupuesto							
Presupuesto ejecutado				Presupuesto Aprobado			
Talento Humano:	<input type="text"/>			<div style="border: 1px solid black; width: 100%; height: 100%;"></div>			
Viajes Técnicos:	<input type="text"/>						
Equipos:	<input type="text"/>						
Recursos Bibliográficos y Software:	<input type="text"/>						
Materiales y Suministros:	<input type="text"/>						
Transferencia y Escudados:	<input type="text"/>						
Subcontratos y Servicios:	<input type="text"/>						
Total:	<input type="text"/>						Total
Resultados Obtenidos:							
<div style="border: 1px solid black; width: 100%; height: 100%;"></div>							
<input type="button" value="Aceptar"/> <input type="button" value="Cancelar"/>							

Actualizar un Proyecto de Investigación

Para actualizar un proyecto de Investigación presionamos sobre **proyecto Investigación** seleccionamos **Proyecto** y escogemos la opción **Actualizar** como nos muestra en la siguiente pantalla.

Proyecto Estudiante	Proyecto Investigacion	Eventos	Publicaciones	Docentes	Ayuda
  	Proyecto	Ingresar			
	Buscar Proyecto	Actualizar			
	Listados				

Buscar Anteproyectos para Matricular

Por Nombres

 Por Cédula

Código	Título del Proyecto	Estado	Continuar	Nombres
--------	---------------------	--------	-----------	---------

En la presente pantalla el usuario podrá ingresar actualizar el proyecto por medio del nombre, cedula o código del integrante y presionamos buscar en donde me aparecerá la información necesaria para ver si es correcto el proyecto que se está buscando para actualizar presionamos sobre el icono continuar.

Proyecto Estudiante	Proyecto Investigacion	Eventos	Publicaciones	Docentes	Ayuda
   					

Buscar Proyectos Docente

Por Nombres

 Por Codigo

 Entre Fechas

Código	Nombre del Proyecto	Estado	Siguiente	Director
--------	---------------------	--------	-----------	----------

Una vez realizado los cambios correspondientes presionamos aceptar para que queden almacenados los cambios como nos muestra en la siguiente pantalla

PROYECTOS DOCENTES

Datos Generales:							
Fecha:	2012-04-02	Resolución:	CD-111-2012				
Título:	proyecto medio ambiente						
Director Proyecto	Ing Raul Rosero			Subdirector Proyecto	Ing. Geovanny Vallejo		
Area:			Estado del Proyecto:				
Sector de impacto	Area de Investigación	Lineas de Investigación	Formulado:	En Ejecución	Ejecutado		
Desarrollo humano y so	Ciencias Exactas y Nat	Desarrollo del riego ter	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>		
Financiamiento							
Institucional	<input checked="" type="radio"/>	Personal	<input type="radio"/>	Ambos	<input type="radio"/>	Otros	<input type="radio"/>
Integrantes							
Docentes	1			Estudiantes	1		
Escuela:	Ing. Diseño Grafico	Cédula:	060345880-3	enviar	Cédula:	060344888-3	enviar
Nombre	Cargo	Título	Dedicación	Nombre	Cargo	Título	
Detalle Formal del Proyecto							
Introducción							

Listado de Proyectos de Investigación

Para ver los Listados de todos los proyectos presionamos sobre **proyecto Investigación** seleccionamos **Listados** como nos muestra en la siguiente pantalla.

Proyecto	Proyecto	Eventos	Publicaciones	Docentes	Ayuda
Estudiante	Investigacion				
	Proyecto				
	Buscar Proyecto				
	Listados	Proyectos Formulados			
		Proyectos en Ejecucion			
		Proyectos Ejecutados			
<input type="radio"/> Por Código:					
<input checked="" type="radio"/> Por Título del Proyecto:					
Código	Nombre	Director	Subdirector	Actualizar	

Eventos

Ingreso Nuevo Evento

Para ingresar un nuevo Evento hacer un clic en **Eventos** de la barra de menú seleccionamos **Eventos** y escogemos la opción **Ingresar** como nos muestra en la siguiente pantalla.

Proyecto Estudiante	Proyecto Investigación	Eventos	Publicaciones	Docentes	Ayuda
		Evento Buscar Eventos Listados	Ingresar Actualizar		

Buscar Proyectos Docentes

Por Código:
 Por Título del Proyecto:

Código	Nombre del Proyecto	Director	Subdirector	Actualizar
--------	---------------------	----------	-------------	------------

En la presente nos muestra todos los campos que se debe llenar para ingresar un proyecto, como nos muestra en la siguiente pantalla.

Proyecto Estudiante	Proyecto Investigación	Eventos	Publicaciones	Docentes	Ayuda

PARTICIPACION EN EVENTOS ACADEMICOS CIENTIFICOS

Datos Generales:					
Fecha:	<input type="text"/>	Resolución:	<input type="text"/>		
Tema:	<input type="text"/>				
Nombre Docente:	<input type="text"/>				
Area:	<input type="text"/>				
Sector de impacto	<input type="text" value="Desarrollo humano y social"/>				Fecha Final: <input type="text"/>
Lugar:	Duración:	Fecha Inicio:	U. Responsable: <input type="text"/>		
<input type="text"/>	<input type="text"/>	<input type="text"/>			
Tipo de Participación					
Ponente:	<input type="radio"/>	Asistente:	<input type="radio"/>		
<input type="button" value="Aceptar"/>			<input type="button" value="Cancelar"/>		

Actualizar un Evento

Para actualizar un **Evento** presionamos sobre Eventos seleccionamos Eventos y escogemos la opción **Actualizar** como nos muestra en la siguiente pantalla.

Proyecto Estudiante	Proyecto Investigación	Eventos	Publicaciones	Docentes	Ayuda
	Proyecto	Ingresar			
	Buscar Proyecto	Actualizar			
	Listados				

Buscar Anteproyectos para Matricular

Por Nombres

 Por Cédula

Código	Título del Proyecto	Estado	Continuar	Nombres
--------	---------------------	--------	-----------	---------

En la presente pantalla el usuario podrá ingresar actualizar el proyecto por medio del nombre, entre fechas presionamos buscar en donde me aparecerá la información necesaria para ver si es correcto el Evento que se está buscando para actualizar presionamos sobre el icono continuar.

Proyecto Estudiante	Proyecto Investigación	Eventos	Publicaciones	Docentes	Ayuda

Buscar Eventos

Por Nombres

 Entre Fechas

Código	Nombre del Evento	Docente Encargado	Area	Siguiente
--------	-------------------	-------------------	------	-----------

Una vez realizado los cambios correspondientes presionamos aceptar para que queden almacenados los cambios como nos muestra en la siguiente pantalla.

Proyecto Estudiante	Proyecto Investigación	Eventos	Publicaciones	Docentes	Ayuda
---------------------	------------------------	---------	---------------	----------	-------

PARTICIPACION EN EVENTOS ACADEMICOS CIENTIFICOS

Datos Generales:					
Fecha:	2012-04-02	Resolución:	CD-111-2012		
Tema:	Seminarios				
Nombre Docente:	Ing. Landy Ruiz				
Area:					
Sector de impacto	Fomento agropecuario y d. productivo		Fecha Final:	U. Responsable:	
Lugar:	Duración:	Fecha Inicio:	2012-04-30	Ing. Raul Rosero	
Riobamba	6 meses	2012-04-02			
Tipo de Participación					
Ponente:	<input checked="" type="radio"/>	Asistente:	<input type="radio"/>		
Aceptar			Cancelar		

Listado de Eventos

Para ver el Listado de todos los Eventos presionamos sobre Eventos seleccionamos **Listados** como nos muestra en la siguiente pantalla.

Proyecto Estudiante	Proyecto Investigación	Eventos	Publicaciones	Docentes	Ayuda
---------------------	------------------------	---------	---------------	----------	-------

Evento		
Buscar Eventos		
Listados	<table border="1"> <tr> <td>Todos Eventos</td> </tr> </table>	Todos Eventos
Todos Eventos		

Buscar Anteproyectos para Matricular

<input checked="" type="radio"/> Por Nombres	<input type="text"/>	Buscar
<input type="radio"/> Por Cédula		

Código	Título del Proyecto	Estado	Continuar	Nombres
--------	---------------------	--------	-----------	---------

Publicaciones

Ingreso Nueva Publicación

Para ingresar una nueva Publicación hacer un clic en Publicaciones de la barra de menú seleccionamos Publicación y escogemos la opción **Ingresar** como nos muestra en la siguiente pantalla.

Proyecto Estudiante	Proyecto Investigacion	Eventos	Publicaciones	Docentes	Ayuda
			Publicación	Ingresar	
			Buscar Publicaciones	Actualizar	
			Listados		

Buscar Anteproyectos para Matricular

Por Nombres
 Por Cédula

Código	Título del Proyecto	Estado	Continuar	Nombres
--------	---------------------	--------	-----------	---------

En la presente nos muestra todos los campos que se debe llenar para ingresar un proyecto, como nos muestra en la siguiente pantalla.

Proyecto Estudiante	Proyecto Investigacion	Eventos	Publicaciones	Docentes	Ayuda
---------------------	------------------------	---------	---------------	----------	-------

PUBLICACIONES

Datos Generales:					
Fecha	<input type="text"/>	Resolución:	<input type="text"/>		
Nombre Publicación	<input type="text"/>	Docente Encargado	<input type="text"/>		
Area:	<input type="text"/>	Lugar:	<input type="text"/>		
Tipo Publicación	<input type="text"/>				
Libros	<input type="radio"/>	Textos Basicos	<input type="radio"/>	Articulos Cientificos	<input type="radio"/>
			Revistas	<input type="text"/>	
			Indexadas	<input type="radio"/>	No Indexadas <input type="radio"/>
<input type="button" value="Aceptar"/>			<input type="button" value="Cancelar"/>		

Actualizar una Publicación

Para actualizar una Publicación presionamos sobre Publicaciones seleccionamos Publicación y escogemos la opción **Actualizar** como nos muestra en la siguiente pantalla.

Proyecto Estudiante	Proyecto Investigación	Eventos	Publicaciones	Docentes	Ayuda
			Publicación	Ingresar	
			Buscar Publicaciones	Actualizar	
			Listados		

PUBLICACIONES

Datos Generales:

Fecha	<input type="text"/>	Resolución:	<input type="text"/>
Nombre Publicación	<input type="text"/>	Docente Encargado	<input type="text"/>
Area:	<input type="text"/>	Lugar:	<input type="text"/>
Tipo Publicación			
Libros	<input type="radio"/>	Textos Basicos	<input type="radio"/>
		Articulos Cientificos	<input type="radio"/>
		Revistas	<input type="text"/>
		Indexadas	<input type="radio"/> No Indexadas <input type="radio"/>

En la presente pantalla el usuario podrá ingresar actualizar el proyecto por medio del nombre, entre fechas presionamos buscar en donde me aparecerá la información necesaria para ver si es correcto el Evento que se está buscando para actualizar presionamos sobre el icono continuar.

Proyecto Estudiante	Proyecto Investigación	Eventos	Publicaciones	Docentes	Ayuda
---------------------	------------------------	---------	---------------	----------	-------

Buscar Publicaciones

Por Nombres
 Entre Fechas

Código	Título	Lugar	Docente encargado	Continuar	Fecha
--------	--------	-------	-------------------	-----------	-------

Una vez realizado los cambios correspondientes presionamos aceptar para que queden almacenados los cambios como nos muestra en la siguiente pantalla

Proyecto Estudiante	Proyecto Investigación	Eventos	Publicaciones	Docentes	Ayuda
---------------------	------------------------	---------	---------------	----------	-------



PUBLICACIONES

Datos Generales:					
Fecha	2012-04-24 00:00:00.0	Resolución:	CD-111-2012		
Nombre Publicación	Redes Inalambricas	Docente Encargado	Ing. Raul		
Area:	Redes	Lugar:	Piobamba		
Tipo Publicación					
Libros	<input checked="" type="radio"/>	Textos Basicos	<input type="radio"/>	Articulos Cientificos	<input type="radio"/>
			Revistas		
			Indexadas	<input type="radio"/>	No Indexadas <input type="radio"/>
Aceptar			Cancelar		

Listado de Publicaciones

Para ver el Listado de todas las Publicaciones presionamos sobre Publicaciones seleccionamos **Listados** como nos muestra en la siguiente pantalla.

Proyecto Estudiante	Proyecto Investigación	Eventos	Publicaciones	Docentes	Ayuda
---------------------	------------------------	---------	---------------	----------	-------



PUBLI

Datos Generales:					
Fecha	2012-04-24 00:00:00.0	Resolución:	CD		
Nombre Publicación	Redes Inalambricas	Docente Encargado	In		
Area:	Redes	Lugar:	Piobamba		
Tipo Publicación					
Libros	<input checked="" type="radio"/>	Textos Basicos	<input type="radio"/>	Articulos Cientificos	<input type="radio"/>
			Revistas		
			Indexadas	<input type="radio"/>	No Indexadas <input type="radio"/>
Aceptar			Cancelar		

- Publicación
- Buscar Publicaciones
- Listados
- Publicaciones de Libros
- Publicaciones Textos Basicos
- Publicaciones Articulos Cientificos

Docentes

Para ver un Reporte de todas las tesis dirigidas por un docente, hacer un clic en Docentes de la barra de menú seleccionamos Reportes y escogemos la opción que vamos a Mostrar cómo nos muestra en la siguiente pantalla.



The screenshot shows a navigation menu with the following items: Proyecto Estudiante, Proyecto Investigacion, Eventos, Publicaciones, Docentes, and Ayuda. The 'Docentes' menu is open, showing a sub-menu with 'Reportes' selected. Below the menu is a search section titled 'Buscar Anteproyectos para Matricular'. It includes radio buttons for 'Por Nombres' (selected) and 'Por Cédula', a search input field, and a blue 'Buscar' button. Below the search section is a table with five columns: Código, Título del Proyecto, Estado, Continuar, and Nombres.

Ayuda

Para ver la ayuda del sistema damos un clic sobre ayuda de la barra de menú como nos muestra en la siguiente pantalla.



The screenshot shows the same navigation menu as the previous image, but with 'Ayuda' selected in the main menu. The 'Ayuda' sub-menu is open, showing a blue button labeled 'Ayuda'. The rest of the page content, including the search section and the table, remains the same as in the previous screenshot.