

**APLICACIÓN WEB PARA GENERACION DE INFORMES  
ACADEMICOS (WAFAP)**

**ARLEY GALINDO FORERO  
DIANA MILENA LOPEZ FONSECA**

**CORPORACION UNIVERSITARIA MINUTO DE DIOS  
TECNOLOGIA EN INFORMATICA  
SOACHA  
2012**

**APLICACIÓN WEB PARA GENERACION DE INFORMES  
ACADEMICOS (WAFAP)**

**ARLEY GALINDO FORERO  
DIANA MILENA LOPEZ FONSECA**

**TRABAJO DE GRADO PARA OPTAR EL TITULO DE TECNOLOGO  
EN INFORMATICA**

**TUTOR  
VIOLETA SUAREZ**



**CORPORACION UNIVERSITARIA MINUTO DE DIOS  
TECNOLOGIA EN INFORMATICA  
SOACHA  
2012**

Nota de aceptación:

---

---

---

---

---

---

---

---

---

---

---

Firma del presidente del jurado

---

Firma del jurado

---

Firma del jurado

## AGRADECIMIENTOS

*ANTES QUE NADA AGRADEZCO A DIOS POR PERMITIRME VIVIR Y QUE CADA DÍA ME LLENE DE CONOCIMIENTOS, DOY GRACIAS A ÉL POR DARMÉ LA FORTALEZA DE LUCHAR DÍA A DÍA Y POR ESTAR SIEMPRE CONMIGO EN CADA PASO QUE DOY.*

*DOY GRACIAS A MI FAMILIA POR SER ESE APOYO INCONDICIONAL EN CADA MOMENTO, EN CADA LUGAR, EN MIS SUEÑOS Y POR LLENARME CADA DÍA DE SU INMENSO AMOR Y VALOR PARA SEGUIR ADELANTE.*

*DIANA MILENA LOPEZ FONSECA*

*Mis agradecimientos son para todas aquellas personas que estuvieron conmigo en toda mi formación profesional, agradezco a Dios por todos los conocimientos adquiridos, por la fortaleza que me da para siempre estar firme con mis propósitos y por no permitirme caer además doy gracias a mi familia por todo su apoyo.*

*Arley Galindo Forero*

## CONTENIDO

### INTRODUCCIÓN

<b>1. TITULO .....</b>	<b>13</b>
<b>APLICACIÓN WEB PARA GENERACION DE INFORMES ACADEMICOS (WAFAP) .....</b>	<b>13</b>
<b>2. TEMA.....</b>	<b>13</b>
<b>3. METODOLOGIA.....</b>	<b>13</b>
<b>4. FASE DE INICIO .....</b>	<b>13</b>
4.1 PLANTEAMIENTO DEL PROBLEMA .....	13
<b>4.2 ALCANCES Y DELIMITACIONES .....</b>	<b>14</b>
4.2.1 ALCANCES .....	14
4.2.2 DELIMITACIONES .....	14
<b>4.3 OBJETIVOS .....</b>	<b>15</b>
4.3.1 GENERAL .....	15
4.3.2 ESPECÍFICOS .....	15
<b>4.4 JUSTIFICACION .....</b>	<b>16</b>
<b>4.5 MARCO DE REFERENCIA .....</b>	<b>17</b>
4.5.1 MARCO HISTÓRICO .....	17
4.5.2 MARCO TEÓRICO.....	18
<b>4.6 HERRAMIENTAS DE DESARROLLO .....</b>	<b>19</b>
<b>4.7 CRONOGRAMA DE ACTIVIDADES.....</b>	<b>19</b>
<b>5. FASE DE ELABORACION.....</b>	<b>19</b>
5.1 DIAGRAMAS DE CASOS DE USO .....	17
5.2 DIAGRAMAS DE SECUENCIA .....	18
5.3 DICCIONARIO DE DATOS .....	24
<b>6. FASE DE CONSTRUCCION .....</b>	<b>26</b>
6.1 DIAGRAMA DE CLASES .....	26
6.2 CONSTRUCCION DE LA BASE DE DATOS.....	18
<b>5. FASE DE TRANSICION .....</b>	<b>19</b>

<b>CONCLUSIONES.....</b>	<b>37</b>
<b>8. BIBLIOGRAFIA E INFOGRAFIA .....</b>	<b>38</b>
<b>9. ANEXOS.....</b>	<b>39</b>

## LISTA DE TABLAS

	Pág.
Tabla 1. Diccionario de datos	24

## LISTA DE ILUSTRACIONES

Ilustración 1. Diagrama de casos de uso general	19
Ilustración 2. Diagrama de caso de uso ingresar al sistema.	20
Ilustracion 3. Diagrama de caso de uso registrar usuarios	20
Ilustración 4. Diagrama de caso de uso ingreso de áreas y asignaturas.	21
Ilustración 5. Diagrama de casos de uso asignación de docentes.	21
Ilustración 6. Diagrama de casos de uso para cargue de notas.	22
Ilustración 7. Diagrama de casos de uso generación de boletines.	22
Ilustración 8. Diagrama de secuencia para ingresar al sistema.	23
Ilustración 9. Diagrama de secuencia para registrar usuarios.	23
Ilustración 10. Diagrama de secuencia para generación de boletines .	24
Ilustracion 11. Diagrama de clases.	26
Ilustracion 12. Base de datos.	33

**LISTA DE ANEXOS**

Pág.

ANEXO I. CRONOGRAMA DE ACTIVIDADES. 39

ANEXO II MANUAL DE USUARIO WAFAP

ANEXO III MANUAL TECNICO

## GLOSARIO

**AJAX:** (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas o RIA (*Rich Internet Applications*). Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargarlas, lo que significa aumentar la interactividad, velocidad y usabilidad en las aplicaciones.

**CSS:** Es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML o XML (y por extensión en XHTML).

**JAVA:** Es un lenguaje de programación orientado a objetos, desarrollado por Sun Microsystems.

**JQUERY:** Es una biblioteca de JavaScript, creada inicialmente por John Resig, que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web.

**JSP:** Es una tecnología Java que permite generar contenido dinámico para web, en forma de documentos HTML, XML o de otro tipo.

**SQL:** El lenguaje de consulta estructurado (SQL) es un lenguaje de base de datos normalizado, utilizado por los diferentes motores de bases de datos para realizar determinadas operaciones sobre los datos o sobre la estructura de los mismos.

**TOMCAT:** También llamado Jakarta Tomcat o simplemente Tomcat, funciona como un contenedor de servlets desarrollado bajo el proyecto Jakarta en la Apache Software Foundation. Tomcat implementa las especificaciones de los servlets y de JavaServer Pages (JSP) de Sun Microsystems.

## RESUMEN

Encuestas realizadas en el municipio de Soacha demostraron que hay muchas instituciones educativas que no manejan sus procesos digitalmente y acuden a entidades ajenas para el desarrollo de los mismos como es el caso de la generación de informes académicos (boletines, certificados, paz y salvos entre otros).

Es por ello que con este documento presentamos la propuesta para implementar un aplicación WEB que permita la generación de informes académicos y que le pertenezca a la institución permitiéndole tener control de sus procesos y reducir gastos. La aplicación en su primera versión será implementada en el colegio Santa Rosa de Lima para la determinación de tiempos y ajustes de gastos y así poderla implementar en más colegios.

La aplicación inicialmente contará con los módulos de ingreso de usuarios, ingreso de asignaturas, ingreso de calificaciones y el de generación de boletines se tendrá en cuenta los perfiles y roles para la correspondiente asignación de usuarios; esta será desarrollada en JSP (Java Server Pages) y utilizaremos AJAX entre otras herramientas de desarrollo, contará con una base de datos confiable, estable y segura que se encontrará montada en un servidor LINUX.

Es necesario contar con un servidor-hosting para el alojamiento de la página web en donde será cargado el aplicativo, para cumplir con todo esto es necesario tener un cronograma de actividades para llevar un control del desarrollo de la página y la aplicación.

## **ABSTRACT**

Surveys in Soacha showed that there are many schools that do not manage their digital processes and turn to outside entities for the development of the same as in the case of academic reporting (newsletters, certificates, and other peace and sound).

That is why this paper we present the proposal to implement a web application that allows academic and reporting that belongs to the institution allowing you to have control of their processes and reduce costs. The first version application will be implemented in the St. Rose of Lima school in determining time and cost adjustments so that it can be deployed and more schools.

The application initially had modules user entry, entry of subjects, grades and income generation newsletters will be considered for roles profiles and the corresponding allocation of users, this will be developed in JSP (Java Server Pages) and use AJAX and other development tools, have a database. Reliable, stable and sure they are mounted on a LINUX server.

You need to have a server-hosting for the accommodation of the website where the application will be loaded to accomplish all that is necessary to have a schedule of activities to track the development of the site and the application.

## INTRODUCCION

En el mundo el manejo de la información y el control de procesos en forma sistematizada es de gran importancia, Existen muchos programas y aplicaciones que se encargan de esto pero a pesar de ello no están siendo cubiertos ciertos sectores.

Desde épocas remotas las instituciones educativas han realizado sus procesos manualmente y mediante archivos, pero en el siglo XXI ya es necesario que todos los colegios tomen conciencia de manejar su información y registros digitalmente. A pesar de que existen demasiados sistemas no han sido implementados ¿a que se deberá esto?. Los sistemas se han desarrollado pero no han tenido el seguimiento correspondiente en cuanto a la implementación y no se han evaluado los factores económicos o tal vez no se pensó realmente en las necesidades de los usuarios.

Un estudio en distintos colegios de Soacha comprueba que son pocas las instituciones que tienen implementadas tecnologías de la información para la ejecución de los procesos. Lo cual demuestra que es factible la implementación de sistemas que controlen estos procesos.

## **1. TITULO**

### **APLICACIÓN WEB PARA GENERACION DE INFORMES ACADEMICOS (WAFAP)**

## **2. TEMA**

Aplicación web que se encargue de la generación de informes académicos en alguna institución educativa del municipio de Soacha utilizando diversos módulos en donde se incluye el del perfil de usuarios.

## **3. METODOLOGIA**

La metodología para el desarrollo del sistema WAFAP es RUP (Rational Unified Process) la cual es un producto del proceso de ingeniería de software que proporcionara a este sistema un enfoque disciplinado para asignar funciones, El objetivo es asegurar el desarrollo de una aplicación web de alta calidad que resuelva las necesidades del colegio SANTA ROSA DE LIMA.

## **4. FASE DE INICIO**

### **4.1 PLANTEAMIENTO DEL PROBLEMA**

#### **4.1.1 DESCRIPCION DEL PROBLEMA**

En muchos colegios del municipio de Soacha no existe un control sistematizado de los informes académicos, por lo tanto deben acudir a personas externas a la institución para realizar los mismos, lo cual genera mas demora en los procesos y gastos para esta; a pesar de que en el mercado ya existen algunas aplicaciones que se encargan de estos procesos, no se han implementado en los colegios del municipio por falta de recursos y falta de conocimiento de la existencia de los mismos; Para ello el sistema WAFAP será desarrollado de tal

manera que el usuario se acople a su funcionamiento y se agilicen los procesos.

La aplicación será implementada en el Colegio Santa Rosa de Lima, ya que esta es una de las instituciones privadas del municipio que necesita este sistema para agilizar sus procesos y reducir costos además darle reconocimiento mediante la página WEB en la que será montada la aplicación.

#### **4.1.2 FORMULACIÓN DEL PROBLEMA**

¿Cómo se maneja la generación de informes académicos en los colegios privados del municipio de Soacha y qué beneficios se obtienen al contar con un software para la generación de estos?

### **4.2 ALCANCES Y DELIMITACIONES**

#### **4.2.1 ALCANCES**

El sistema WAFAP en su primera versión contará con el módulo de registro de usuarios, el módulo de asignación de materias, el módulo de registro de calificaciones y el módulo de generación de boletines, esto incluye el control de usuarios y la correspondiente asignación de roles.

#### **4.2.2 DELIMITACIONES**

El software será implementado en el Colegio Santa Rosa de Lima en el municipio de Soacha. Este es un colegio privado que necesita un sistema que agilice sus procesos y así mismo reducir gastos y obtener beneficios como el manejo digital de sus informes académicos.

## **4.3 OBJETIVOS**

### **4.3.1 GENERAL**

Diseñar y desarrollar un sistema que controle y genere informes académicos del colegio "SANTA ROSA DE LIMA" del municipio de Soacha mediante la implementación de un aplicativo web para optimizar el proceso y reducir costos.

### **4.3.2 ESPECÍFICOS**

- Implementar el manejo de perfiles para controlar el acceso a los usuarios que puedan estar vinculados a la institución.
- Crear un modulo que permita a la institución generar internamente sus boetines haciendo de esto un proceso mas rápido y con la misma eficiencia.
- Elaborar un manual de usuario digital que permita conocer el funcionamiento del sistema.
- Diseñar una base de datos confiable, estable y segura.

#### **4.4 JUSTIFICACION**

El sistema WAFAP en su primera version contara con el modulo de registro de usuarios , el modulo de asignacion de materias , el modulo de registro de calificaciones y el modulo de generacion de boletines, esto incluye el control de usuarios y la correspondiente asignacion de roles.

A partir de la necesidades que se presentan en el colegio SANTA ROSA DE LIMA se desarrollara una aplicación WEB que contara con los módulos de registro de usuarios, asignación de materias , registro de calificaciones , y el modulo de generación de boletines, esta aplicación será desarrollada en busca de optimizar procesos académicos y reducir costos sobre el manejo de la institución en la parte académica puesto que ellos acuden a personas ajenas a la institución para generar sus boletines y el manejo de la información es manual. Por esta razón se busca implementar este sistema que llevara un control sistematizado de estos procesos y le permitirá a la institución tener total dominio sobre estos.

El sistema contara con un logueo en donde se hará la correspondiente validación de los usuarios, los cuales cumplen un rol dentro del colegio y basados en esto se le asignaran los privilegios.

## **4.5 MARCO DE REFERENCIA**

### **4.5.1 MARCO HISTÓRICO**

#### **Antecedentes de la institución**

La institución educativa "Santa Rosa de Lima" del municipio de Soacha fue fundada en 1976. Iniciando sus labores desde 1979 ofreciendo un aporte pedagógico, educativo, formativo y social, en una zona de rápido desarrollo poblacional del municipio de Soacha brindando educación en el área de preescolar, 1°, 2° y 3° de básica primaria. Hoy el día la institución cuenta con grados desde preescolar hasta 11° y para llevar un control académico eficiente de 401 estudiantes que estudian actualmente en el, mas los estudiantes nuevos que se integren a esta comunidad educativa es necesario implementar tecnologías de la información para tal fin. Actualmente el colegio "Santa rosa de lima" necesita de entidades externas para el procesamiento y la generación de los informes académicos de los estudiantes, enviando los reportes académicos en medios digitales (Excel) y como respuesta a esto las empresas encargadas de esta labor procesan esta información y como resultado la entrega final del producto son los informes académicos en material físico lo cual hace que la información sea más difícil de manipular.

#### **Antecedentes del proyecto**

En muchos colegios de Colombia antiguamente manejaban la mayor parte de su información en formatos físicos, pero con la llegada de las nuevas tecnologías fueron convirtiendo esta información en formatos digitales con la ventaja de que el acceso a la información fuese mas fácil, además de que no ocuparían espacio en el medio laboral y así mismo no habrían preocupaciones por deterioro de dichos materiales. Pero esto no era suficiente ya que el concepto de "seguridad de la información" no era tenida muy en cuenta y los sistemas

eran muy vulnerables a perder su información por virus o ataques que se realizaran a través de la red.

Con las tecnologías de la información que han sido implementadas en el país, se ha trabajado bastante porque los procesos sean ejecutados con precisión y los resultados sean los correctos disminuyendo cada día su margen de error. Actualmente se han desarrollado sistemas para colegios que abarcan módulos para la sistematización de informes académicos implementando además de esto muchos módulos para una gran cantidad de procesos que pueda tener cualquier colegio lo cual hace que la magnitud de estos sistemas sea muy amplia y requieran de medios físicos (hardware) con altas capacidades de almacenamiento y procesamiento por lo cual son pocos los colegios que cuenten con los recursos tecnológicos para adquirirlos. Como solución a esto será desarrollado un sistema que se familiarice con las necesidades de un colegio en específico.

#### **4.5.2 MARCO TEÓRICO**

La tecnología en la actualidad es de gran importancia, es por ello que la mayoría de las entidades llevan control de sus procesos digitalmente y hacen uso de la tecnología en la que la sociedad ya esta sumergida, entre estas entidades encontramos los colegios, los cuales manejan muchos procesos financieros y academicos y encuentran ideal llevarlos sistemáticamente. Porque les permite tener orden sobre toda la información manejada además les reduce costos y agilizan sus procesos. Pero se debe tener en cuenta que no todos los colegios manejan su información digitalmente, aun existen instituciones que llevan el control de su registros manualmente , lo cual no es grave pero lo ideal seria llevarlos digitalmente ya que esto permite tener muchas ventajas en cuanto a la organización y los tiempos de trabajo, también le permite al colegio tener total dominio sobre su información y reducir gastos.

A pesar de que existen muchas aplicaciones creadas con el fin de optimizar procesos academicos, éstas no han sido implementadas y muchos colegios no las utilizan por la seguridad de su información o por los recursos que exigen estas aplicaciones.

Tener un control sistematizado de la información tiene muchas ventajas tanto organizacionales como económicas y a pesar de que ya existen aplicaciones que ofrecen ciertos servicios a las instituciones educativas para el control de sus procesos es necesario buscar y contar con el enfoque de la institución para que esta se sienta mas segura y obtenga el software como propio.

## **4.6 HERRAMIENTAS DE DESARROLLO**

La aplicación se desarrollara bajo el lenguaje de programación JSP ya que este es un lenguaje orientado a objetos y es fuerte porque contiene su propio compilador y permite que la segunda carga o las siguientes versiones sean mucho más rápidas también será utilizada la librería jquery que permite simplificar código javascript y crear animaciones en los DOM de HTML.

Utilizaremos la técnica de desarrollo Ajax ya que esta realiza peticiones asincrónicas al servidor y agiliza las consultas y las cargas además es ideal para desarrollar aplicaciones interactivas.

El uso de las hojas de estilo es muy importante para el diseño de la aplicación y de la página web, en cuanto a la implementación de la aplicación se necesita un servidor\_hosting que contenga servidores APACHE TOMCAT y MYSQL es este caso será implementado bajo plataforma LINUX ya que es un sistema gratuito y es más seguro en cuanto a virus.

## **4.7 CRONOGRAMA DE ACTIVIDADES**

Para el desarrollo de la aplicación web y el diseño de la página es necesario definir tiempos y actividades, por ello hemos utilizado la popular herramienta gráfica (DIAGRAMA DE GANTT), la cual nos ha permitido establecer esto.

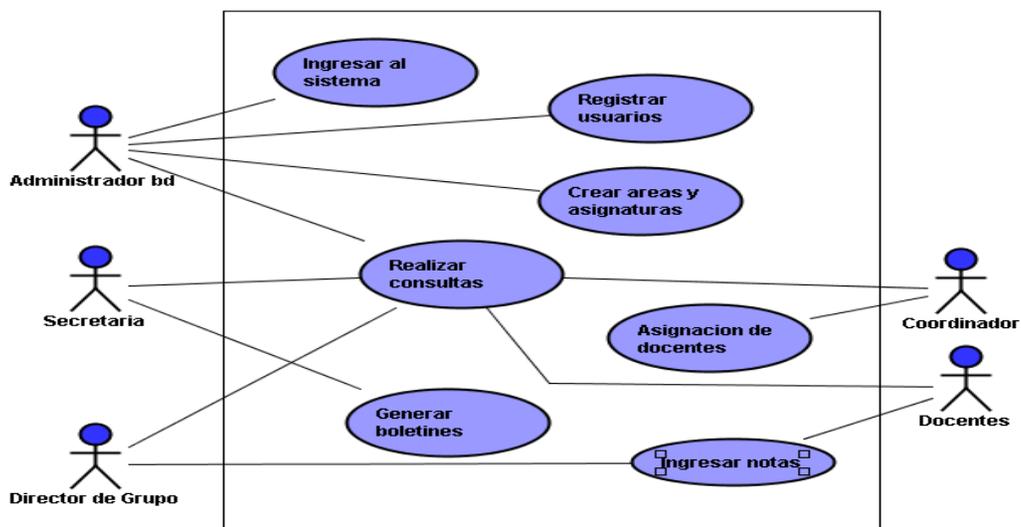
**Ver anexo 1.**

## 5. FASE DE ELABORACION

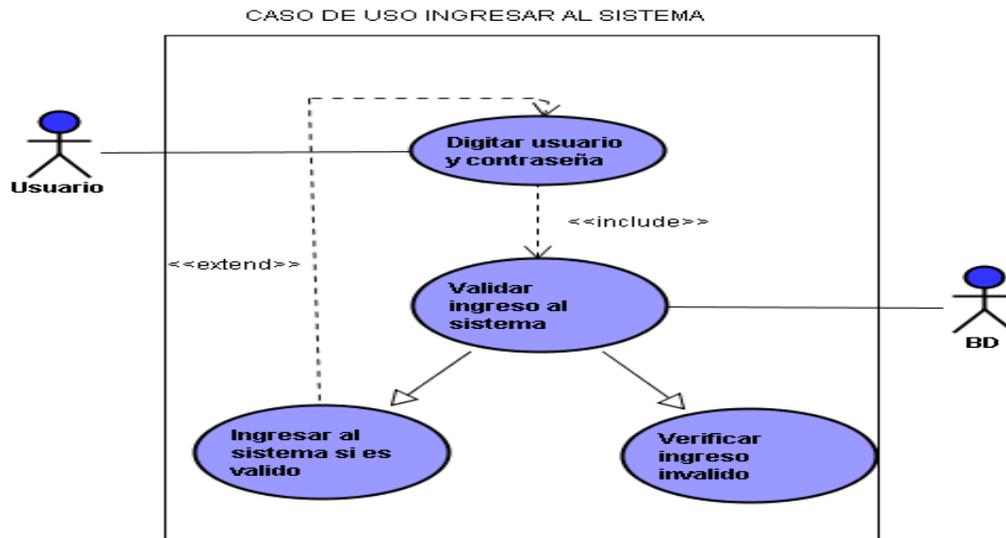
Esta fase de la metodología RUP permite establecer la estructura base para la arquitectura del sistema WAFAP y para ello tenemos los diagramas de casos de uso, los diagramas secuenciales y el diagrama de clases tambien se tendra en cuenta el diccionario de datos, el cual nos permite diseñar nuestro modelo entidad relación.

### 5.1 DIAGRAMAS DE CASOS DE USO

Ilustracion 1. Diagrama de casos de uso general



Ilustracion 2. Diagrama de caso de uso ingresar al sistema.



Ilustracion 3. Diagrama de caso de uso registrar usuarios

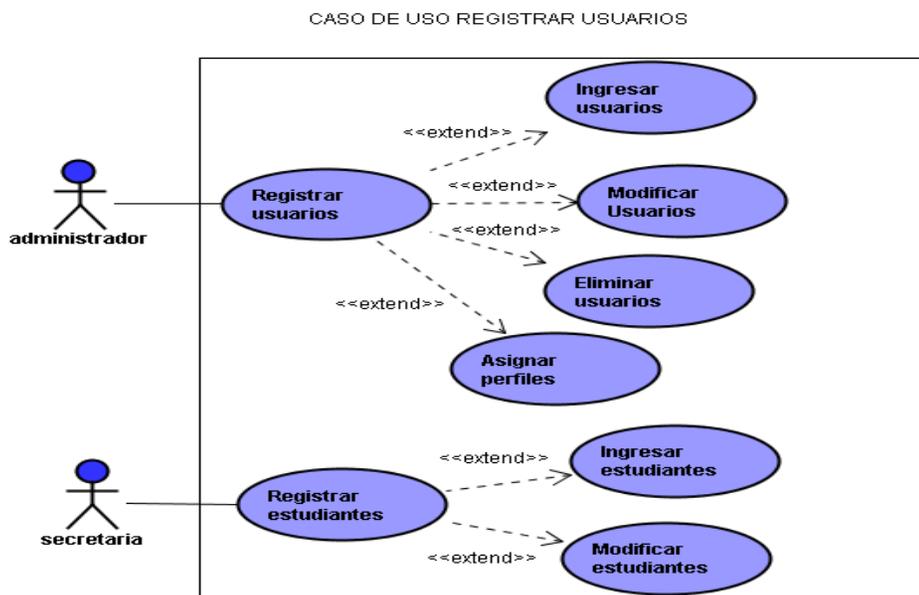


Ilustración 4. Diagrama de caso de uso ingreso de áreas y asignaturas.

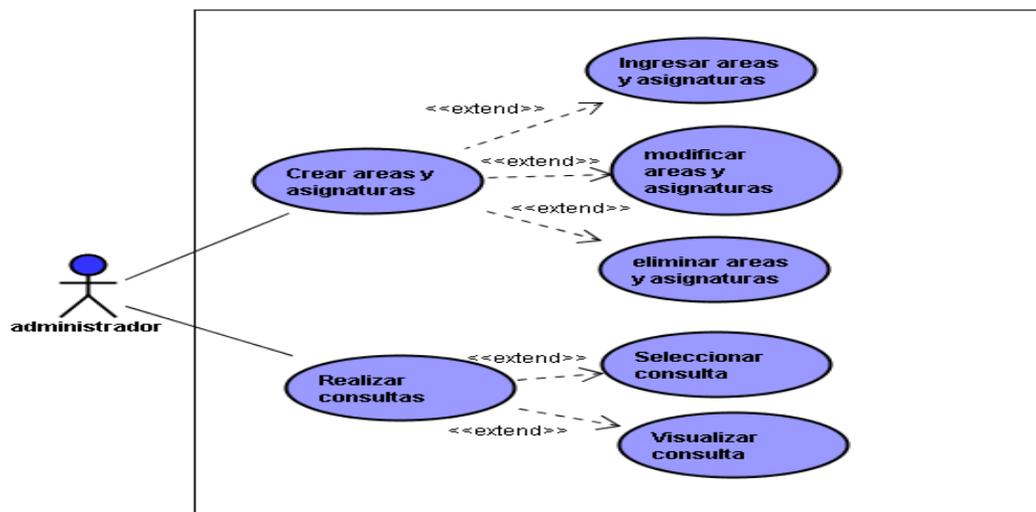


Ilustración 5. Diagrama de casos de uso asignación de docentes.

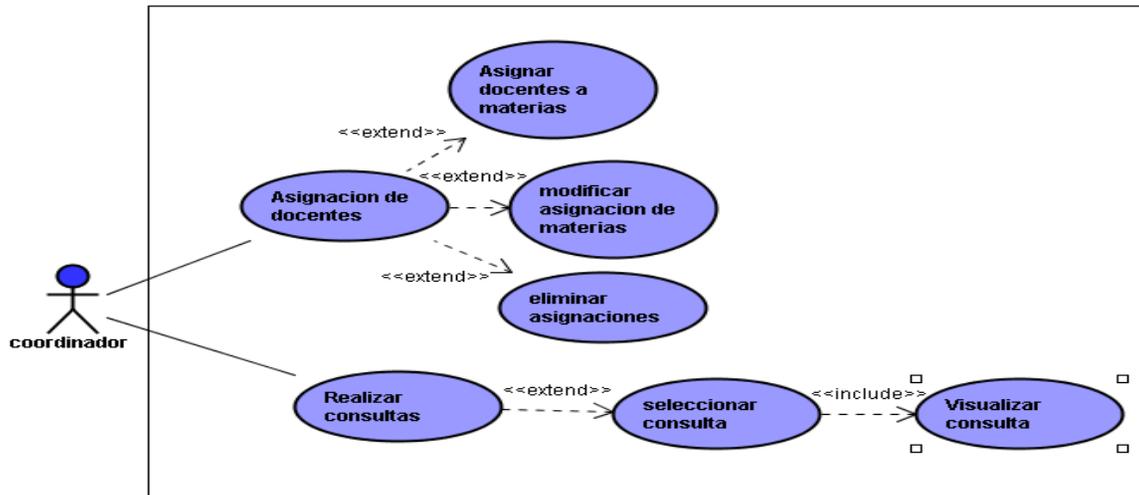


Ilustración 6. Diagrama de casos de uso para cargue de notas

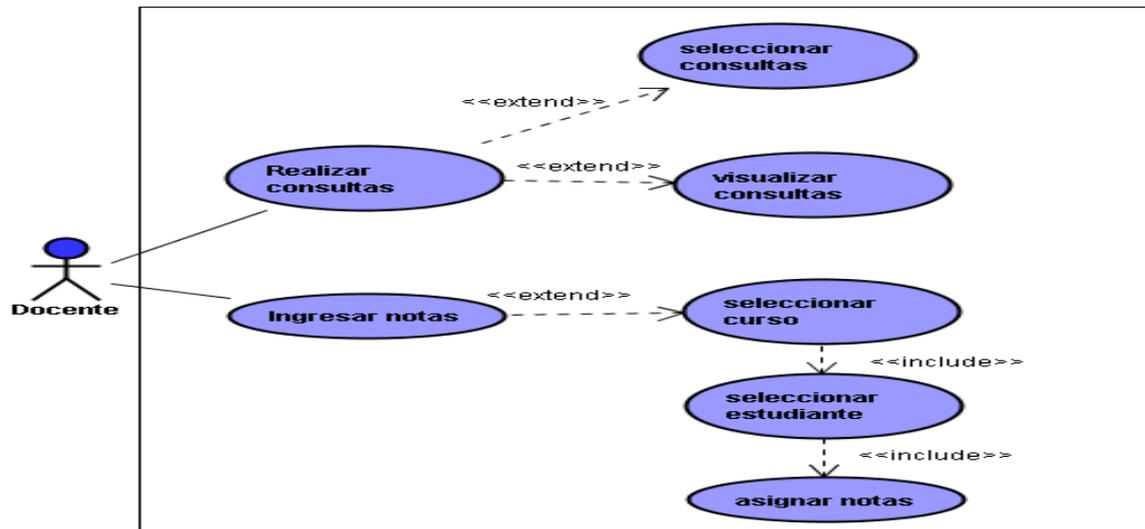
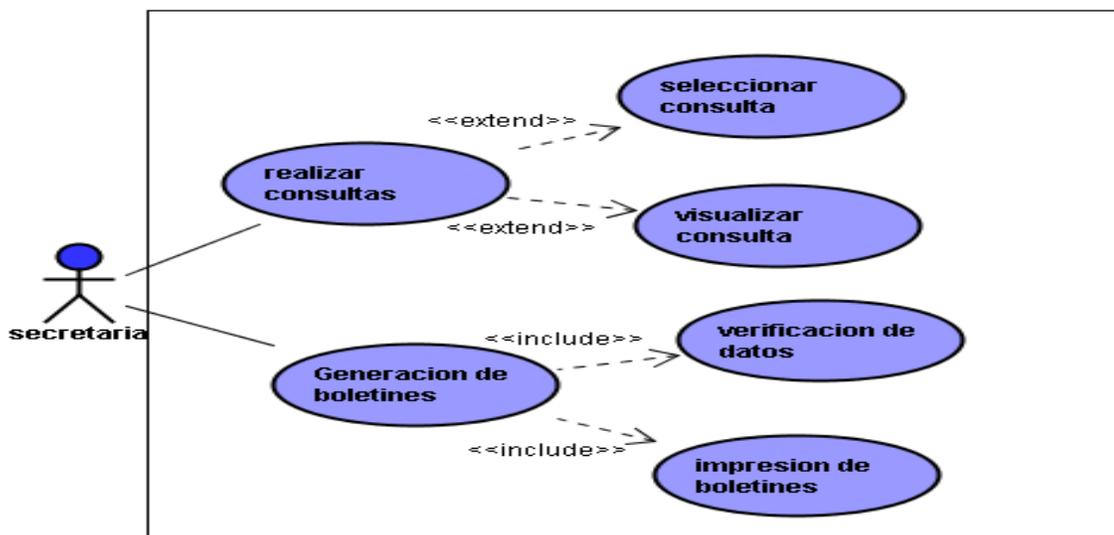


Ilustración 7. Diagrama de casos de uso generación de boletines.



## 5.2 DIAGRAMAS DE SECUENCIA

Ilustración 8. Diagrama de secuencia para ingresar al sistema

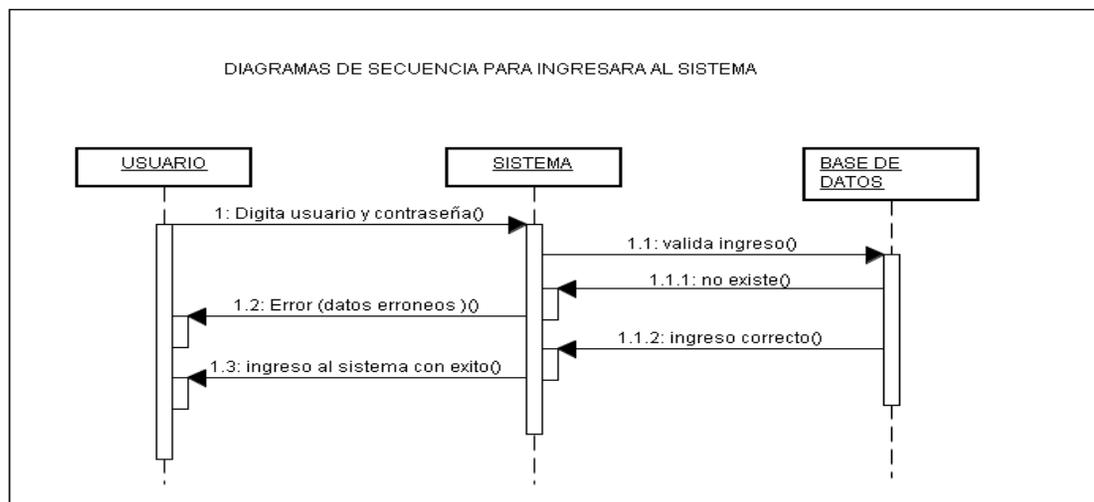


Ilustración 9. Diagrama de secuencia para registrar usuarios

DIAGRAMA DE SECUENCIA PARA REGISTRAR USUARIOS AL SISTEMA

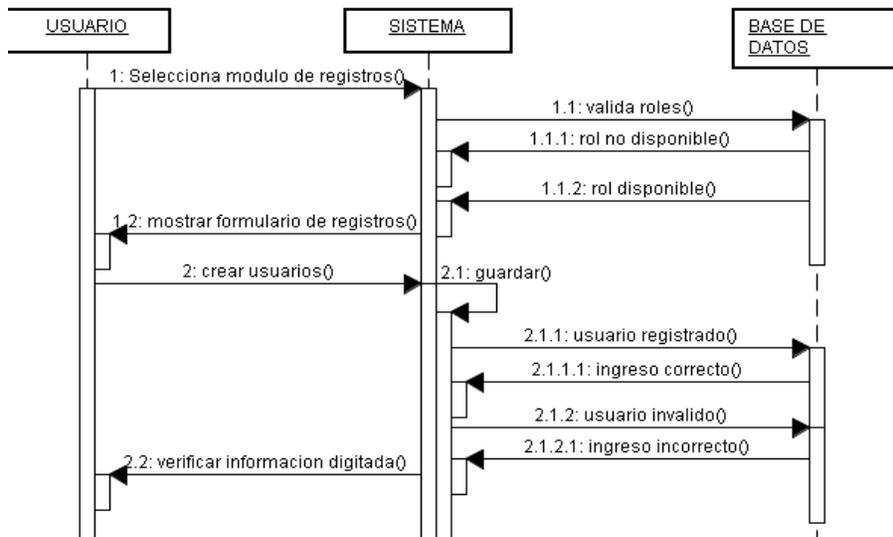
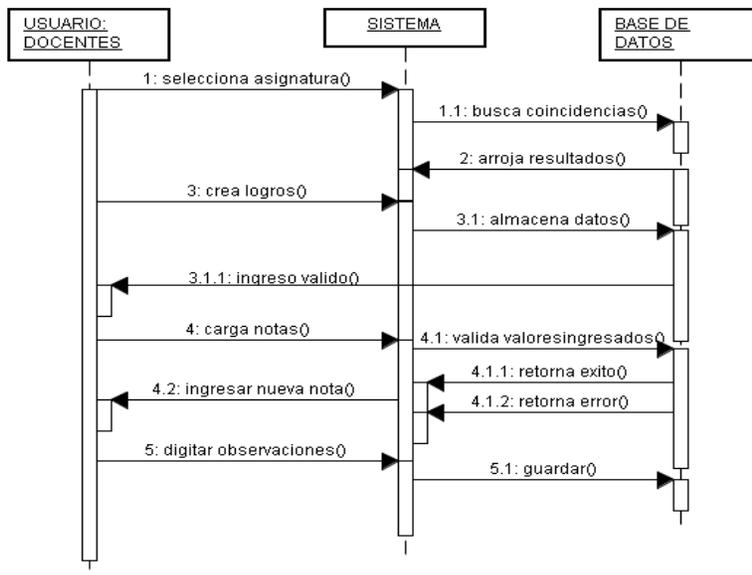


Ilustración 10. Diagrama de secuencia para generación de boletines



### 5.3 DICCIONARIO DE DATOS

Tabla 1. Diccionario de datos

<b>USUARIOS</b>
Cod_user
Usr_nomb
Usr_apel1
Usr_apel2
Usr_pass
Tipo_docu
Lugar_exp
Fecha_exp
Dirección
Teléfono
Celular
Lugar_nacim
Fecha_nacim
Email
Cod_tipo
Ind_matricula
Usr_insert
Fecha_insert
Usr_update
Fecha_update
Cod_estado

<b>MATRICULAS</b>	<b>NOTAS</b>	<b>BOLETINES</b>
Year	Cod_boletin	Periodo
Cod_user	Cod_logro	Year
Cod_curso	Nota	Cod_user
		Cod_curso
		Observaciones
<b>CURSOS</b>	<b>LOGROS</b>	<b>ASIGNATURAS</b>
Cod_curso	Cod_logro	Cod_asig
Desc_curso	Desc_logro	Cod_materia
Rango	Cod_asig	Cod_curso
Usr_insert	Periodo	Cod_year
Fecha_insert	Usr_insert	Cod_user
Usr_update	Fecha_insert	Usr_insert
Fecha_update	Usr_update	Fecha_insert
Ind_estado	Fecha_update	Usr_update
	Ind_estado	Fecha_update

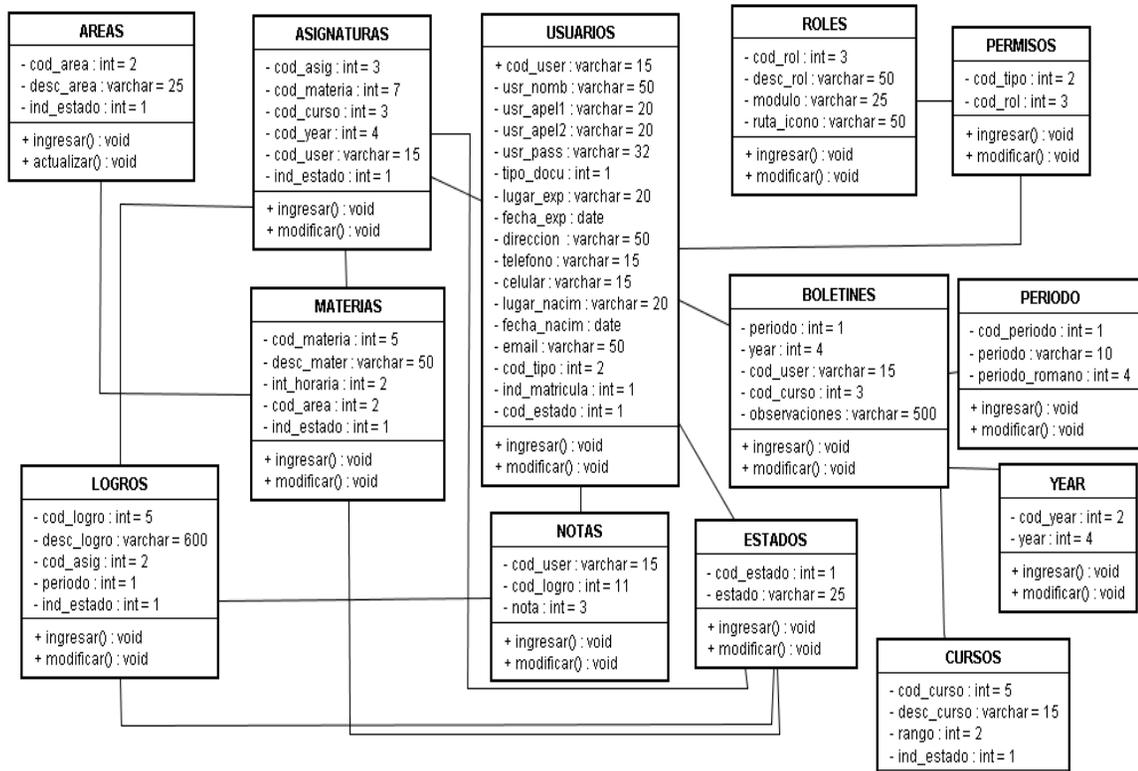
		Ind_estado
<b>AREAS</b>	<b>PERIODO</b>	<b>MATERIAS</b>
Cod_area	Cod_periodo	Cod_materia
Desc_area	Periodo	Desc_mater
Usr_insert	Periodo_romano	Int_horaria
Fecha_insert	Usr_insert	Cod_area
Usr_update	Fecha_insert	Usr_insert
Fecha_update	Usr_update	Fecha_insert
Ind_estado	Fecha_update	Usr_update
		Fecha_update
		Ind_estado
<b>ESTADOS</b>	<b>TIPO_DOCUMENTO</b>	<b>TIPO_USUARIO</b>
Cod_estado	Tipo_docu	Cod_tipo
Estado	Desc_docu	Nom_tipo
		Cod_estado
<b>PERMISOS</b>	<b>ROLES</b>	<b>YEAR</b>
Cod_tipo	Cod_rol	Cod_year
Cod_rol	Desc_rol	Year
Usr_insert	Modulo	<b>USER_YEAR</b>
Fecha_insert	Ruta_icono	Cod_year
Usr_update	Usr_insert	Cod_user
Fecha_update	Fecha_insert	<b>YEAR_CURSO</b>
	Usr_update	Cod_year
	Fecha_update	Cod_curso
		<b>CURSO_USER</b>
		Cod_curso
		Cod_user

## 6. FASE DE CONSTRUCCION

En esta fase se tendrá en cuenta el diagrama de clases junto con el modelo entidad relación para la construcción de la base de datos llegando hasta la cuarta forma normal lo que nos permite estandarizar la base de datos (ARDI) y darle mayor seguridad a la información.

## 6.1 DIAGRAMA DE CLASES

Ilustración 11. Diagrama de clases



## 6.2 CONSTRUCCION DE LA BASE DE DATOS

### 1. Tabla áreas

```

CREATE TABLE `areas` (
  `cod_area` int(2) NOT NULL AUTO_INCREMENT,
  `desc_area` varchar(25) NOT NULL,
  `usr_insert` varchar(50) NOT NULL,
  `fecha_insert` datetime NOT NULL,
  `usr_update` varchar(50) NOT NULL,

```

```

`fecha_update` datetime NOT NULL,
`ind_estado` int(1) NOT NULL,
PRIMARY KEY (`cod_area`),
KEY `desc_area` (`desc_area`)
) ENGINE=InnoDB AUTO_INCREMENT=3 DEFAULT CHARSET=latin1.

```

## 2. Tabla de asignaturas

```

CREATE TABLE `asignaturas` (
  `cod_asig` int(3) NOT NULL AUTO_INCREMENT,
  `cod_materia` int(7) NOT NULL,
  `cod_curso` int(5) NOT NULL,
  `cod_year` int(2) NOT NULL,
  `cod_user` varchar(15) NOT NULL,
  `usr_insert` varchar(50) NOT NULL,
  `fecha_insert` datetime NOT NULL,
  `usr_update` varchar(50) NOT NULL,
  `fecha_update` datetime NOT NULL,
  `ind_estado` int(1) NOT NULL,
  PRIMARY KEY (`cod_asig`),
  KEY `nom_asig` (`cod_materia`),
  KEY `cod_curso` (`cod_curso`),
  KEY `cod_year` (`cod_year`),
  KEY `cod_user` (`cod_user`),
  KEY `ind_estado` (`ind_estado`),
  CONSTRAINT `asignaturas_ibfk_5` FOREIGN KEY (`ind_estado`) REFERENCES
`estados` (`cod_estado`),
  CONSTRAINT `asignaturas_ibfk_1` FOREIGN KEY (`cod_materia`) REFERENCES
`materias` (`cod_materia`),
  CONSTRAINT `asignaturas_ibfk_2` FOREIGN KEY (`cod_curso`) REFERENCES
`cursos` (`cod_curso`),
  CONSTRAINT `asignaturas_ibfk_3` FOREIGN KEY (`cod_year`) REFERENCES
`year` (`cod_year`),
  CONSTRAINT `asignaturas_ibfk_4` FOREIGN KEY (`cod_user`) REFERENCES
`usuarios` (`cod_user`)
) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=latin1

```

## 3. Tabla de Boletines

```

CREATE TABLE `boletines` (
  `cod_periodo` int(1) NOT NULL,
  `cod_year` int(2) NOT NULL,
  `cod_user` varchar(15) NOT NULL,
  `cod_curso` int(5) NOT NULL,
  `observaciones` varchar(500) NOT NULL,
  PRIMARY KEY (`cod_periodo`,`cod_year`,`cod_user`,`cod_curso`),
  KEY `cod_year` (`cod_year`),

```

```

KEY `cod_user` (`cod_user`),
KEY `cod_curso` (`cod_curso`),
CONSTRAINT `boletines_ibfk_4` FOREIGN KEY (`cod_curso`) REFERENCES
`cursos` (`cod_curso`),
CONSTRAINT `boletines_ibfk_1` FOREIGN KEY (`cod_periodo`) REFERENCES
`periodo` (`cod_periodo`),
CONSTRAINT `boletines_ibfk_2` FOREIGN KEY (`cod_year`) REFERENCES `year`
(`cod_year`),
CONSTRAINT `boletines_ibfk_3` FOREIGN KEY (`cod_user`) REFERENCES
`usuarios` (`cod_user`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1

```

#### 4. Tabla de contacto

```

CREATE TABLE `contacto` (
  `cod_contacto` int(11) NOT NULL AUTO_INCREMENT,
  `nombres` varchar(50) NOT NULL,
  `email` varchar(50) NOT NULL,
  `comentarios` varchar(600) NOT NULL,
  PRIMARY KEY (`cod_contacto`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1

```

#### 5. Tabla de cursos

```

CREATE TABLE `cursos` (
  `cod_curso` int(5) NOT NULL,
  `desc_curso` varchar(15) NOT NULL,
  `rango` int(2) NOT NULL,
  `usr_insert` varchar(50) NOT NULL,
  `fecha_insert` datetime NOT NULL,
  `usr_update` varchar(50) NOT NULL,
  `fecha_update` datetime NOT NULL,
  `ind_estado` int(1) NOT NULL,
  PRIMARY KEY (`cod_curso`),
  KEY `desc_curso` (`desc_curso`),
  KEY `ind_estado` (`ind_estado`),
  CONSTRAINT `cursos_ibfk_2` FOREIGN KEY (`ind_estado`) REFERENCES `estados`
(`cod_estado`),
  CONSTRAINT `cursos_ibfk_1` FOREIGN KEY (`ind_estado`) REFERENCES `estados`
(`cod_estado`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1

```

#### 6. Tabla de curso\_user

```

CREATE TABLE `curso_user` (
  `cod_curso` int(5) NOT NULL,
  `cod_user` varchar(15) NOT NULL,
  PRIMARY KEY (`cod_curso`,`cod_user`)

```

) ENGINE=InnoDB DEFAULT CHARSET=latin1

#### 7. Tabla de estados

```
CREATE TABLE `estados` (
  `cod_estado` int(1) NOT NULL,
  `estado` varchar(25) NOT NULL,
  PRIMARY KEY (`cod_estado`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1
```

#### 8. Tabla de logros

```
CREATE TABLE `logros` (
  `cod_logro` int(5) NOT NULL AUTO_INCREMENT,
  `desc_logro` varchar(600) DEFAULT NULL,
  `cod_asig` int(3) DEFAULT NULL,
  `cod_periodo` int(1) NOT NULL,
  `usr_insert` varchar(50) NOT NULL,
  `fecha_insert` datetime NOT NULL,
  `usr_update` varchar(50) NOT NULL,
  `fecha_update` datetime NOT NULL,
  `ind_estado` int(1) NOT NULL,
  PRIMARY KEY (`cod_logro`),
  KEY `periodo` (`cod_periodo`),
  KEY `cod_asig` (`cod_asig`),
  KEY `ind_estado` (`ind_estado`),
  CONSTRAINT `logros_ibfk_3` FOREIGN KEY (`ind_estado`) REFERENCES `estados`
  (`cod_estado`),
  CONSTRAINT `logros_ibfk_1` FOREIGN KEY (`cod_asig`) REFERENCES `asignaturas`
  (`cod_asig`),
  CONSTRAINT `logros_ibfk_2` FOREIGN KEY (`cod_periodo`) REFERENCES `periodo`
  (`cod_periodo`)
) ENGINE=InnoDB AUTO_INCREMENT=9 DEFAULT CHARSET=latin1
```

#### 9. Tabla materias

```
CREATE TABLE `materias` (
  `cod_materia` int(7) NOT NULL AUTO_INCREMENT,
  `desc_mater` varchar(50) NOT NULL,
  `int_horaria` int(2) NOT NULL,
  `cod_area` int(2) NOT NULL,
  `usr_insert` varchar(50) NOT NULL,
  `fecha_insert` datetime NOT NULL,
  `usr_update` varchar(50) NOT NULL,
  `fecha_update` datetime NOT NULL,
  `ind_estado` int(1) NOT NULL,
  PRIMARY KEY (`cod_materia`),
```

```

KEY `cod_area` (`cod_area`),
KEY `ind_estado` (`ind_estado`),
CONSTRAINT `materias_ibfk_2` FOREIGN KEY (`ind_estado`) REFERENCES `estados`
(`cod_estado`),
CONSTRAINT `materias_ibfk_1` FOREIGN KEY (`cod_area`) REFERENCES `areas`
(`cod_area`)
) ENGINE=InnoDB AUTO_INCREMENT=14 DEFAULT CHARSET=latin1

```

#### 10.Tabla de notas

```

CREATE TABLE `notas` (
`cod_user` varchar(15) NOT NULL,
`cod_logro` int(11) NOT NULL,
`nota` int(3) NOT NULL,
PRIMARY KEY (`cod_user`,`cod_logro`),
KEY `cod_logro` (`cod_logro`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1

```

#### 11.Tabla de periodo

```

CREATE TABLE `periodo` (
`cod_periodo` int(1) NOT NULL,
`periodo` varchar(10) NOT NULL,
`periodo_romano` char(4) NOT NULL,
`usr_insert` varchar(50) NOT NULL,
`fecha_insert` datetime NOT NULL,
`usr_update` varchar(50) NOT NULL,
`fecha_update` datetime NOT NULL,
PRIMARY KEY (`cod_periodo`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1}

```

#### 12.Tabla de permisos

```

CREATE TABLE `permisos` (
`cod_tipo` int(2) NOT NULL,
`cod_rol` int(3) NOT NULL,
`usr_insert` varchar(50) NOT NULL,
`fecha_insert` datetime NOT NULL,
`usr_update` varchar(50) NOT NULL,
`fecha_update` datetime NOT NULL,
PRIMARY KEY (`cod_tipo`,`cod_rol`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1

```

## 13. Tabla de roles

```
CREATE TABLE `roles` (
  `cod_rol` int(3) NOT NULL,
  `desc_rol` varchar(50) NOT NULL,
  `modulo` varchar(25) NOT NULL,
  `ruta_icono` varchar(50) NOT NULL,
  `fecha_insert` datetime NOT NULL,
  `usr_insert` varchar(20) NOT NULL,
  `fecha_update` datetime NOT NULL,
  `usr_update` varchar(20) NOT NULL,
  PRIMARY KEY (`cod_rol`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1
```

## 14. Tabla de tipo\_documento.

```
CREATE TABLE `tipo_documento` (
  `tipo_docu` int(1) NOT NULL,
  `desc_docu` varchar(25) NOT NULL,
  PRIMARY KEY (`tipo_docu`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1
```

## 15. Tabla de tipo\_usuario

```
CREATE TABLE `tipo_usuario` (
  `cod_tipo` int(2) NOT NULL,
  `nom_tipo` varchar(40) NOT NULL,
  `cod_estado` int(1) NOT NULL,
  PRIMARY KEY (`cod_tipo`),
  KEY `cod_estado` (`cod_estado`),
  CONSTRAINT `tipo_usuario_ibfk_1` FOREIGN KEY (`cod_estado`) REFERENCES `estados`
  (`cod_estado`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1.
```

## 16. Tabla User\_year

```
CREATE TABLE `user_year` (
  `cod_year` int(4) NOT NULL,
  `cod_user` varchar(15) NOT NULL,
  PRIMARY KEY (`cod_year`,`cod_user`),
  CONSTRAINT `user_year_ibfk_1` FOREIGN KEY (`cod_year`) REFERENCES `year`
  (`cod_year`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1
```

## 17. Tabla Usuarios

```
CREATE TABLE `usuarios` (
  `cod_user` varchar(15) NOT NULL,
```

```

`usr_nomb` varchar(50) NOT NULL,
`usr_apel1` varchar(20) NOT NULL,
`usr_apel2` varchar(20) NOT NULL,
`usr_pass` varchar(32) NOT NULL COMMENT 'contraseña encriptada con md5 para inicio de
sesión',
`tipo_docu` int(1) NOT NULL,
`lugar_exp` varchar(20) NOT NULL,
`fecha_exp` date NOT NULL,
`direccion` varchar(50) NOT NULL,
`telefono` varchar(15) NOT NULL,
`celular` varchar(15) NOT NULL,
`lugar_nacim` varchar(20) NOT NULL,
`fecha_nacim` date NOT NULL,
`email` varchar(50) NOT NULL,
`cod_tipo` int(2) NOT NULL,
`ind_matricula` int(1) DEFAULT '0',
`usr_insert` varchar(50) NOT NULL,
`fecha_insert` datetime NOT NULL,
`usr_update` varchar(50) NOT NULL,
`fecha_update` datetime NOT NULL,
`cod_estado` int(1) NOT NULL,
PRIMARY KEY (`cod_user`),
KEY `cod_tipo` (`cod_tipo`),
KEY `tipo_docu` (`tipo_docu`),
KEY `cod_estado` (`cod_estado`),
CONSTRAINT `usuarios_ibfk_1` FOREIGN KEY (`cod_tipo`) REFERENCES `tipo_usuario`
(`cod_tipo`),
CONSTRAINT `usuarios_ibfk_2` FOREIGN KEY (`tipo_docu`) REFERENCES
`tipo_documento` (`tipo_docu`),
CONSTRAINT `usuarios_ibfk_3` FOREIGN KEY (`cod_estado`) REFERENCES `estados`
(`cod_estado`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1

```

#### 18. Tabla de año

```

CREATE TABLE `year` (
`cod_year` int(2) NOT NULL,
`year` int(4) NOT NULL,
PRIMARY KEY (`cod_year`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1

```

#### 19. Tabla de año curso

```

CREATE TABLE `year_curso` (
`cod_year` int(4) NOT NULL,
`cod_curso` int(5) NOT NULL,
PRIMARY KEY (`cod_year`,`cod_curso`)

```

) ENGINE=InnoDB DEFAULT CHARSET=latin

## Ilustracion 12. Base de datos

<b>ardi_areas</b> cod_area : int(2) desc_area : varchar(25) usr_insert : varchar(50) fecha_insert : datetime usr_update : varchar(50) fecha_update : datetime ind_estado : int(1)	<b>ardi_roles</b> cod_rol : int(3) desc_rol : varchar(50) modulo : varchar(25) ruta_icono : varchar(50) fecha_insert : datetime usr_insert : varchar(20) fecha_update : datetime usr_update : varchar(20)	<b>ardi_usuarios</b> cod_user : varchar(15) usr_nomb : varchar(50) usr_apel1 : varchar(20) usr_apel2 : varchar(20) usr_pass : varchar(32) tipo_docu : int(1) lugar_exp : varchar(20) fecha_exp : date direccion : varchar(50) telefono : varchar(15) celular : varchar(15) lugar_nacim : varchar(20) fecha_nacim : date email : varchar(50) cod_tipo : int(2) ind_matricula : int(1) usr_insert : varchar(50) fecha_insert : datetime usr_update : varchar(50) fecha_update : datetime cod_estado : int(1)	<b>ardi_tipo_documento</b> tipo_docu : int(1) desc_docu : varchar(25)
<b>ardi_logros</b> cod_logro : int(5) desc_logro : varchar(600) cod_asig : int(3) cod_perodo : int(1) usr_insert : varchar(50) fecha_insert : datetime usr_update : varchar(50) fecha_update : datetime ind_estado : int(1)	<b>ardi_permisos</b> cod_tipo : int(2) cod_rol : int(3) usr_insert : varchar(50) fecha_insert : datetime usr_update : varchar(50) fecha_update : datetime	<b>ardi_tipos_usuario</b> cod_tipo : int(2) nom_tipo : varchar(40) cod_estado : int(1)	<b>ardi_estados</b> cod_estado : int(1) estado : varchar(25)
<b>ardi_materias</b> cod_materia : int(7) desc_mater : varchar(50) int_horaria : int(2) cod_area : int(2) usr_insert : varchar(50) fecha_insert : datetime usr_update : varchar(50) fecha_update : datetime ind_estado : int(1)	<b>ardi_cursos</b> cod_curso : int(5) desc_curso : varchar(15) rango : int(2) usr_insert : varchar(50) fecha_insert : datetime usr_update : varchar(50) fecha_update : datetime ind_estado : int(1)	<b>ardi_notas</b> cod_user : varchar(15) cod_logro : int(11) nota : int(3)	<b>ardi_contacto</b> cod_contacto : int(11) nombres : varchar(50) email : varchar(50) comentarios : varchar(600)
<b>ardi_asignaturas</b> cod_asig : int(3) cod_materia : int(7) cod_curso : int(5) cod_year : int(2) cod_user : varchar(15) usr_insert : varchar(50) fecha_insert : datetime usr_update : varchar(50) fecha_update : datetime ind_estado : int(1)	<b>ardi_periodo</b> cod_perodo : int(1) periodo : varchar(10) periodo_romano : char(4) usr_insert : varchar(50) fecha_insert : datetime usr_update : varchar(50) fecha_update : datetime	<b>ardi_boletines</b> cod_perodo : int(1) cod_year : int(2) cod_user : varchar(15) cod_curso : int(5) observaciones : varchar(500)	<b>ardi_year</b> cod_year : int(2) year : int(4)
		<b>ardi_user_year</b> cod_year : int(4) cod_user : varchar(15)	<b>ardi_year_curso</b> cod_year : int(4) cod_curso : int(5)
		<b>ardi_curso_user</b> cod_curso : int(5) cod_user : varchar(15)	

## **7. FASE DE CIERRE O TRANSICION**

En esta fase se entregara el producto para que sea implementado y hacer las pruebas correspondientes también se hara entrega del manual de usuario y el manual técnico al cliente.

MANUAL DE USUARIO WAFAP

MANUAL TECNICO

## CONCLUSIONES

El desarrollo de la aplicación WAFAP ha sido un reto bastante grande ya que no solo se buscaba presentar una propuesta de proyecto para optar el título de tecnólogos en informática se busco obtener mas conocimientos, para ello decidimos desarrollar la aplicacion en JSP y utilizar otras tecnologías lo cual fue parte de nuestro valor agregado porque no se tenia ningún conocimiento para desarrollar con este lenguaje pero lo importante es que se logro.

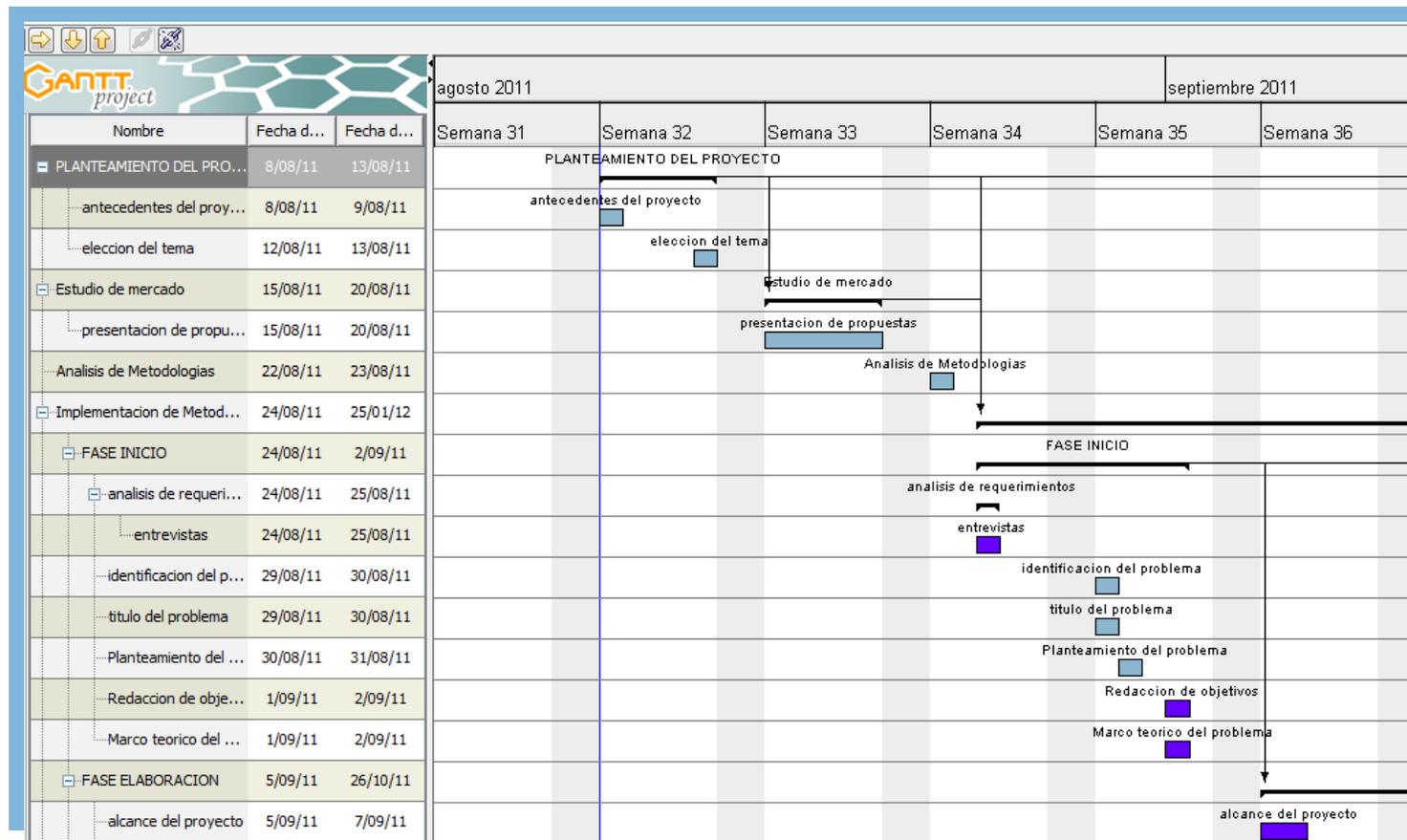
Tambien es muy importante resaltar que el municipio de Soacha tiene muy poco énfasis sobre la parte tecnológica y es por ello que podrían desarrollarse muchos proyectos que beneficien el desarrollo tecnológico del municipio y asi generar mas empleo para los tecnólogos en informática u otros programas y es importante resaltar que los colegios es una de las entidades que necesitan bastante del uso de herramientas tecnológicas por muchas causas entre ellas el manejo de sus procesos academicos digitalmente, que fue en lo que nosotros nos enfocamos para el desarrollo del aplicativo.

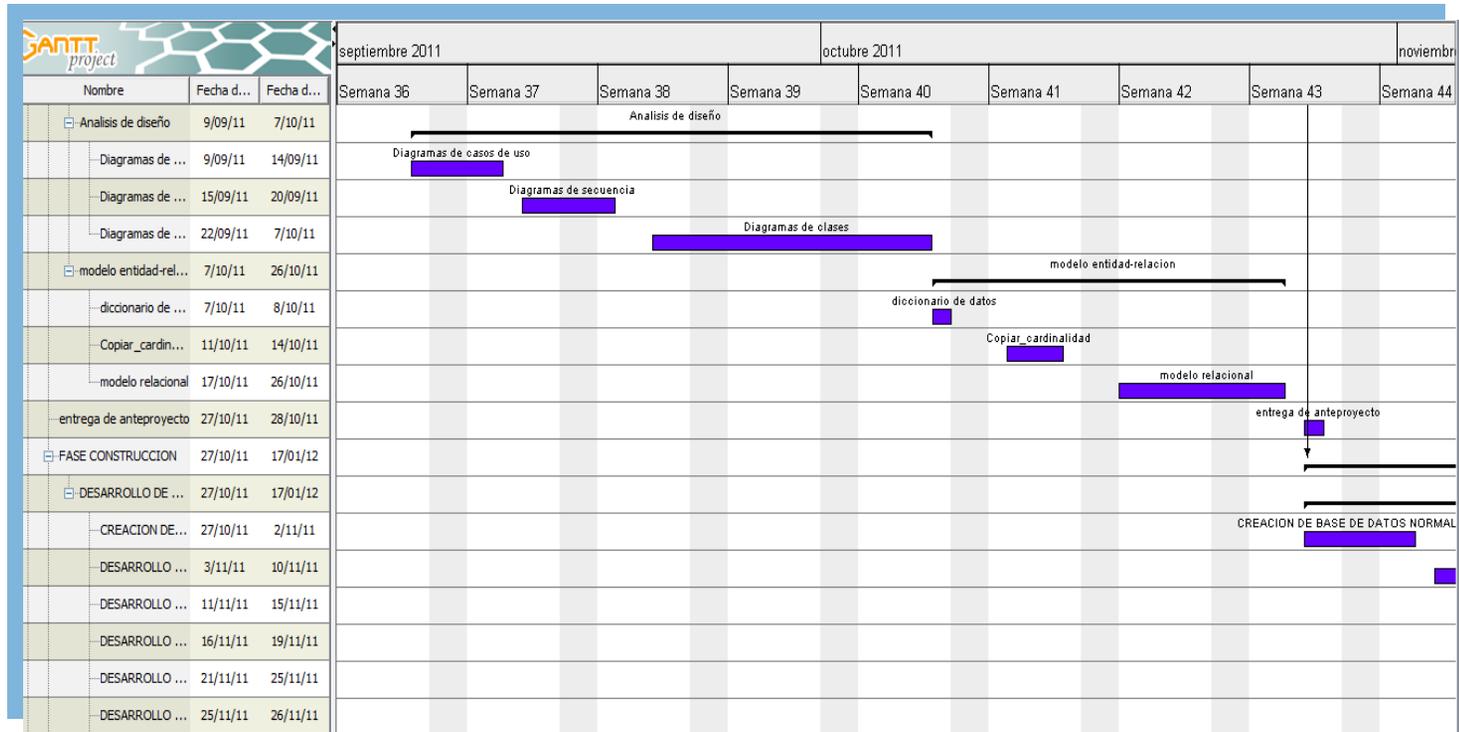
## 8. BIBLIOGRAFIA E INFOGRAFIA

- [http://dis.um.es/~lopezquesada/documentos/IES\\_0607/DFSI/curso/UT11/cookiesJSP.pdf](http://dis.um.es/~lopezquesada/documentos/IES_0607/DFSI/curso/UT11/cookiesJSP.pdf)
- [http://xweb1.ousli.org/material/d\\_intro\\_servlets\\_jsp.pdf](http://xweb1.ousli.org/material/d_intro_servlets_jsp.pdf)
- <http://www.todoexpertos.com/categorias/tecnologia-e-internet/desarrollo-de-sitios-web/jsp/respuestas/82463/pasar-datos-de-un-jsp-a-otro>.
- <http://sliceoflinux.com/2011/04/28/instalar-ubuntu-11-04-paso-a-paso/>
- <http://www.ubuntu.com/download/ubuntu/download>
- <http://www.acatos.es/2010/02/25/instalar-tomcat-6-en-ubuntu/>
- <http://ingmmurillo.blogspot.com/2010/12/como-instalar-apache-tomcat-6-en-ubuntu.html>
- [https://forja.rediris.es/docman/view.php/282/493/instalacion\\_tomcat.pdf](https://forja.rediris.es/docman/view.php/282/493/instalacion_tomcat.pdf)
- <http://jcalderon.wordpress.com/2008/01/04/instalacion-y-configuracion-de-apache-tomcat-6-0-en-windows-xp/>
- <http://www.guia-ubuntu.org/index.php?title=MySQL>

## 9. ANEXOS

### ANEXO I. CRONOGRAMA DE ACTIVIDADES.





SANTT project			noviembre 2011					diciembre 2011				
Nombre	Fecha d...	Fecha d...	Semana 44	Semana 45	Semana 46	Semana 47	Semana 48	Semana 49	Semana 50	Semana 51	Semana 52	
DESARROLLO DE ...	27/10/11	17/01/12										
CREACION DE...	27/10/11	2/11/11										
DESARROLLO ...	3/11/11	10/11/11										
DESARROLLO ...	11/11/11	15/11/11										
DESARROLLO ...	16/11/11	19/11/11										
DESARROLLO ...	21/11/11	25/11/11										
DESARROLLO ...	25/11/11	26/11/11										
INTERFAZ GR...	28/11/11	3/12/11										
REDACCION D...	14/12/11	17/12/11										
REDACCION D...	21/12/11	23/12/11										
PRUEBAS SIS...	16/01/12	17/01/12										
FASE DE TRANSICION	17/01/12	25/01/12										
LEGALIZACION DE...	17/01/12	18/01/12										
CAPACTTACION (...)	23/01/12	25/01/12										
ENTREGA DE LA A...	17/01/12	18/01/12										
PRUEBAS POST DE...	23/01/12	24/01/12										
ENTREGA DE TESIS DE GR...	17/01/12	18/01/12										

DATOS NORMALIZADA											
DESARROLLO DE LOGUEO											
DESARROLLO DE MODULO DE REGISTRO DE USUARIOS Y VALIDACION											
DESARROLLO DE MODULO D ASIGNATURAS											
DESARROLLO DE MODULO GENERACION DE DOCUMENTOS											
DESARROLLO DE MODULO BOLETINES											
INTERFAZ GRAFICA											
REDACCION DE MANUAL DE USUARIO											
REDACCION DE TECHNICAL REFERENCES											



