



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

APLICACIÓN EN INTERNET PARA LA AUTOMATIZACIÓN
DE LA ADMINISTRACIÓN DEL TRÁMITE DE SERVICIO
SOCIAL DE LA FACULTAD DE INGENIERÍA

TESIS PROFESIONAL

QUE PARA OBTENER EL TÍTULO DE:
INGENIERO EN COMPUTACIÓN

PRESENTAN: *Madalope*
GALICIA GONZÁLEZ GPE. MIRIAM
VÁZQUEZ CHÁVEZ BRUNO DAVID

DIRECTOR DE TESIS: ING. ALEJANDRO VELÁZQUEZ MENA



CD. UNIVERSITARIA, MÉXICO, D. F.

Enero 2005

m. 344581



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

AGRADECIMIENTOS

A Dios:

Por darme la vida y la dicha de sentir y disfrutar cada una de las cosas que me rodean.

A mi madre, Socorrito:

Porque siempre lo ha dado todo por mí. Por sus esfuerzos y desvelos. Por ser padre y madre al mismo tiempo. Por todo su amor.

A mis abuelos Carlitos y Cuquita:

Por el amor que sus ojos y sus brazos me expresan todos los días. Por ser también, mis padres.

A mis tíos, Carlos y Martha:

Por escucharme y apoyarme en todos los momentos importantes para mí. Por su cariño y comprensión. Por confiar en mí.

A Alejandro V. Mena:

Por el tiempo dedicado a este trabajo. Por su apoyo dentro y fuera de la tesis.

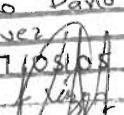
A Bruno:

Por el esfuerzo para culminar este trabajo. Por su amistad.

A mis amigos:

Por compartir conmigo un pequeño trozo de sus vidas. Por todas y cada una de nuestras valiosas conversaciones.

A la UNAM, por inyectarme toda esa magia que sentimos todos aquellos que nos enorgullecemos de pertenecer a ella.

Autorizo a la Dirección General de Bibliotecas de la UNAM a difundir en formato electrónico e impreso el contenido de mi trabajo recepcional.
NOMBRE: Bruno David Vázquez Chávez
FECHA: 27/05/05
FIRMA: 

Gpe. Miriam Galicia González

DEDICATORIAS

A mi madre Socorrito:

Porque con nada podré devolverle todo lo que ha hecho por mí.

A mis abuelos Carlitos y Cuquita, a mis tíos Carlos y Martha:

Por ser piezas claves en mi formación personal. Por la ayuda que siempre me han dado a lo largo de la vida.

A mi hermano César:

Por ser mi compañero en el tren de la vida. Por confiar en mí.

A Omar, Edgar, Cassandra y Lucy:

Por ser mis hermanos menores, y por permitirme apoyarlos en su formación.

*Para todo aquel, cuya humildad le permita
consultar este trabajo y gastar en ello un poco de
su valioso tiempo.*

Gpe. Miriam Galicia González

AGRADECIMIENTOS

A mis padres:

Gracias por enseñarme, con el ejemplo, la responsabilidad y el respeto hacia el trabajo. Por su cariño, apoyo y esfuerzo para formar un ingeniero. Para ambos, muchas gracias.

A mis hermanos:

Gracias por brindarme su compañía y apoyo en todo momento.

A Alejandro Velázquez Mena:

Gracias por ser más que un asesor y por tu invaluable apoyo para la terminación de este trabajo.

A Miriam:

Gracias por tu comprensión, esfuerzo y dedicación durante el desarrollo de este trabajo. Por tu confianza y amistad.

A Isis:

Gracias por tu cariño, comprensión y apoyo en todo momento.

A Dios:

Por permitirme vivir y por darme las fuerzas para seguir cumpliendo mis metas.

Bruno David Vázquez Chávez

DEDICATORIAS

A mis padres, por brindarme la posibilidad de estudiar y darme todo el apoyo para formarme como ingeniero en esta máxima casa de estudios, a la que nunca terminaré de agradecerle la invaluable enseñanza y experiencias que deja en mi vida.

Bruno David Vázquez Chávez

ÍNDICE

CONVENCIONES TIPOGRÁFICAS	1
INTRODUCCIÓN	2
CAPÍTULO 1 ESTUDIO PRELIMINAR	
1.1 Antecedentes	4
1.2 Definición del Problema	5
1.3 Necesidades del Cliente	5
1.3.1 Cuestionario Fase I	6
1.3.2 Cuestionario Fase II	6
1.4 Identificación de los Sistemas Cooperativos	7
1.5 Recursos disponibles y recursos que se obtendrán	8
1.6 Estudio de Factibilidad	8
1.6.1 Factibilidad Técnica	8
1.6.2 Factibilidad Económica	8
1.6.3 Factibilidad Operacional	9
CAPÍTULO 2 CONCEPTOS BÁSICOS	
2.1 Ingeniería de Software	11
2.1.1 El Ciclo de Vida del Software	11
2.1.2 Definición y Objetivos de la Ingeniería de Software	11
2.1.3 El Modelo de Madurez de Capacidades: CMM (Capability Maturity Model)	12
2.1.3.1 Arquitectura del CMM	13
2.2 Tipos de Metodologías	14
2.2.1 Definición y Objetivos de la Metodología	14
2.2.2 Clasificación de las Metodologías	15
2.2.2.1 Metodologías Estructuradas	15
2.2.2.2 Metodologías Orientadas a Objetos	17
2.3 La Unificación de las Metodologías Orientadas a Objetos en un Proceso de Desarrollo y un Lenguaje de Modelado	20
2.3.1 El Proceso Unificado de Desarrollo de Software	20
2.3.2 El Lenguaje Unificado de Modelado (Unified Modeling Language, UML)	21
2.4 Tipos de Arquitectura	22
2.4.1 Arquitectura Lógica de Sistemas	22
2.4.1.1 Modelo de Dos Capas	22
2.4.1.2 Modelo de Tres Capas	23
2.4.1.3 Modelo de N Capas	23
2.4.1.4 Beneficios para Utilizar Arquitectura de Tres o Más Capas	23
2.4.2 Arquitectura Física de Sistemas	24
CAPÍTULO 3 EVALUACIÓN TECNOLÓGICA	
3.1 Herramientas de Programación en Tecnología Web	25
3.1.1 Common Gateway Interface (CGI)	25
3.1.2 Practical Extracting and Reporting Language (PERL)	26
3.1.3 Servlets	27
3.1.4 Hypertext Preprocessor (PHP)	28
3.1.5 Active Server Pages (ASP)	30
3.1.6 Java Server Pages (JSP)	30
3.1.7 JavaScript	31
3.1.8 Applets	32
3.2 Tecnologías de Componentes para Aplicaciones Web	32
3.2.1 Enterprise Java Beans (EJB)	32

3.2.2	COM, DCOM, COM+	33
3.3	Navegadores de Internet	33
3.3.1	Netscape.....	34
3.3.2	Mozilla.....	34
3.3.3	Internet Explorer.....	34
3.3.4	Opera	35
3.3.5	Konqueror.....	35
3.3.6	Lynx.....	35
3.3.7	Estándares de Web y HTTP.....	36
3.4	Servidor Web y Servidor de Aplicación	36
3.4.1	Apache.....	37
3.4.2	AOL Server.....	37
3.4.3	Internet Information Server (IIS).....	37
3.4.4	Servlet Engine.....	38
3.4.5	Java Application Servers	39
3.5	Manejadores de Bases de Datos.....	39
3.5.1	Oracle.....	40
3.5.2	Sybase Server.....	41
3.5.3	Informix	42
3.5.4	SQL Server	43
3.6	Requerimientos de Hardware y Software	44
3.6.1	Requerimientos de Software.....	44
3.6.2	Requerimientos de Hardware.....	45

CAPÍTULO 4 ANÁLISIS Y DISEÑO DEL SISTEMA

4.1	Análisis del Sistema	47
4.1.1	Análisis de Requerimientos	47
4.1.2	Identificación de los Procesos del Sistema	49
4.1.3	Diagrama de Flujo de los Procesos del Sistema.....	51
4.1.4	Diagrama General de Casos de Uso.....	52
4.1.5	Descripción de los Casos de Uso Esenciales	53
4.1.6	Análisis del Dominio y el Modelo Conceptual.....	54
4.1.7	Diagrama de Flujo del Sistema.....	55
4.1.8	Bitácora de Riesgo.....	64
4.1.9	Lista de Personas	64
4.2	Diseño del Sistema	66
4.2.1	Análisis de Requerimientos del Diseño	66
4.2.2	Secuencia de pantallas	81
4.2.2.1	Flujo de pantallas comunes para los diferentes usuarios (Alumno, Coordinador de SS de la División de Ingeniería, Jefe de SS de la FI)	81
4.2.2.2	Flujo de pantallas para el Alumno	90
4.2.2.3	Flujo de Pantallas para el Coordinador de SS de la División de Ingeniería.....	97
4.2.2.4	Flujo de Pantallas para el Jefe de SS de la FI	118
4.2.3	Diagrama de Clases	140
4.2.4	Diagramas de Secuencias.....	141
4.2.5	Diagrama E/R	163
4.2.6	Diccionario de Datos	164
4.2.6.1	Tablas de Trabajo	164
4.2.6.2	Catálogos	168
4.2.7	Lista de Personas	176

CAPÍTULO 5	CONSTRUCCIÓN DEL SISTEMA	
5.1	Análisis del Tiempo	177
5.1.1	Asignación de Tiempos para la Instalación del Ambiente de Desarrollo.....	177
5.1.2	Asignación de Tiempos para la Construcción de la Base de Datos.....	177
5.1.3	Asignación de Tiempos para la Construcción de la Aplicación.....	177
5.1.4	Asignación de Tiempos para el Diseño Gráfico de la Aplicación.....	177
5.2	Fases de Construcción.....	178
5.2.1	Front-End o Capa de Presentación	178
5.2.2	Middleware o Capa de la Lógica del Negocio	184
5.2.3	Back-End o Capa de Almacenamiento.....	190
5.2.4	Paquetes Descriptivos de la Arquitectura.....	194
5.3	Herramientas de Diseño y Programación Utilizadas en la Construcción del Sistema de Administración de Servicio Social.....	195
5.3.1	Herramientas de Diseño	195
5.3.2	Herramientas de Programación	196
5.4	Estándares de Programación	197
CAPÍTULO 6	PRUEBAS DEL SISTEMA	
6.1	Pruebas Realizadas por el Equipo de Trabajo.....	198
6.1.1	Pruebas de Software.....	198
6.1.2	Pruebas de Hardware.....	206
6.2	Pruebas Realizadas por el Usuario.....	207
6.3	Corrección de Programas	210
CAPÍTULO 7	LIBERACIÓN Y MANTENIMIENTO DEL SISTEMA	
7.1	Instalación del Sistema de Administración de Servicio Social.....	213
7.1.1	Servidor de Base de Datos SYBASE	213
7.1.2	Servicio Tomcat	215
7.2	Mantenimiento del Sistema.....	217
7.2.1	Tipos de Mantenimiento.....	218
CONCLUSIONES	219
APÉNDICE A	MINUTAS DE LAS REUNIONES CON EL USUARIO	
A.1	Minuta Fase I de Levantamiento de Información.....	221
A.2	Minuta Fase II de Levantamiento de Información.....	224
A.3	Minuta de la Reunión Celebrada el 9 de Marzo de 2004.....	228
A.4	Minuta de la Reunión Celebrada el 2 de Abril de 2004.....	229
A.5	Minuta de la Reunión Celebrada el 19 de Mayo de 2004.....	231
APÉNDICE B	VISIÓN GENERAL DE UML	
B.1	UML (Unified Modeling Language).....	233
B.2	Casos de Uso.....	233
B.3	Diagramas de Actividades	235
B.4	Diagramas de Clases	236
B.5	Diagramas de Secuencias.....	237
B.6	Diagramas de Componentes.....	238
B.7	Diagramas de Distribución o Despliegue.....	241
GLOSARIO	243
BIBLIOGRAFÍA	250

CONVENCIONES TIPOGRÁFICAS

Con el objetivo de brindar al lector un apoyo bibliográfico e informativo en la lectura del presente trabajo de tesis, se siguieron las siguientes convenciones:

- ◆ Se utiliza la fuente *arial itálica* para señalar la primera ocurrencia de aquellos términos que pueden ser consultados en el glosario que se encuentra al final de la tesis.
- ◆ Los párrafos que al final presentan referencias identificadas como [n], indican que la oración o párrafo fue extraído del libro cuya referencia bibliográfica se encuentra al final de esta tesis. Cabe señalar que n es un valor que indica el número de la referencia bibliográfica.

Introducción



INTRODUCCIÓN

Los sistemas de información han sufrido cambios que buscan asegurar que se alcancen resultados que satisfagan cualquier tipo de necesidad mediante el desarrollo de sistemas de calidad que utilicen tecnología de vanguardia. Los sistemas de información representan una herramienta indispensable en la actualidad para el desarrollo de todas las actividades de cualquier organización, empresa, comercio o escuela, ya que, sin lugar a dudas optimizan enormemente el control de la información y los procesos que se realizan en éstas incrementando de este modo su eficiencia y competitividad.

Otro de los adelantos tecnológicos que sin lugar a dudas desató un gran avance en el área de la informática, es el desarrollo de la red de información más importante que existe: Internet. Desde sus inicios la red Internet ha estado fuertemente ligada con el ambiente académico ya que la mayor parte de los avances, en cuanto al uso y generación de nuevos recursos, han sido primero presentados como proyectos de investigación que después son exportados con gran éxito al ambiente empresarial. Lo anterior es la base de lo que ahora constituye esta completa red de información a la cual podemos acceder desde cualquier parte del mundo.

Actualmente, Internet se ha convertido en el medio más accesible para la publicación de los sistemas de información que manejan grandes acervos de información, comercio electrónico, correo electrónico, foros de discusión, entretenimiento y tecnología, solo por mencionar algunos. Todos estos sistemas contribuyen cada día más al progreso de empresas, instituciones, universidades y personas individuales al proporcionarles grandes herramientas para la comunicación entre ellos y para la automatización de sus procesos.

Actualmente los procesos para el registro, la realización y liberación del Servicio Social (SS) en la Facultad de Ingeniería, requieren de actividades que consumen tiempo y recursos los cuales pueden disminuirse y que, de ser así, simplificarían de gran forma las actividades de los alumnos y del personal que se ve involucrado en las actividades propias de este proceso.

Cuando el alumno empieza a investigar qué proyectos existen para realizar su SS tiene la necesidad de buscar carteles o anuncios en la Facultad de Ingeniería, o bien, debe visitar las Escuelas y Dependencias en las cuales tiene la posibilidad de realizar el Servicio Social para conocer los proyectos actuales en los cuales puede realizar esta importante actividad. Posteriormente, cuando el alumno encuentra un proyecto que le interesa y satisface sus necesidades, comienza su proceso de registro del SS para el cual debe realizar algunas visitas a la Facultad o Dependencia donde desarrollará sus actividades de SS y a la Facultad de Ingeniería para cumplir con todos los trámites propios del proceso de registro. Una vez que se hizo el registro, el alumno debe realizar sus actividades en el lugar, proyecto y horario establecidos en el registro, entregando bimestralmente un reporte de actividades anexo a un Formato de Reporte de Servicio Social que requiere la Unidad de Servicio Social de la Facultad de Ingeniería (USS). Al término del SS, debe solicitarse la liberación del mismo en la USS y recoger la carta de liberación en las oficinas de Servicios Escolares de la Facultad.

Como puede notarse, todas estas actividades requieren de tiempo tanto de los trabajadores que controlan estos trámites, como de los alumnos que los realizan y tienen que desplazarse de un lugar a otro para recoger o entregar, según sea el caso, la documentación necesaria para cada uno de estos procesos.

Por lo mencionado anteriormente, se decidió realizar un Sistema de Información por Internet, el cual, realizará en la medida de lo posible, la automatización de los procesos de Servicio Social de la Facultad de Ingeniería, con lo cual, se pretende colaborar con su desarrollo tecnológico, así como facilitar los trámites para que sean más sencillos y rápidos para todos los involucrados.

En el Capítulo 1 se presenta el Estudio Preliminar del proyecto, los antecedentes, la definición del problema, la identificación de los sistemas involucrados, así como la definición de los recursos disponibles y los que se obtendrán, la factibilidad técnica y los beneficios que ofrece la aplicación.

En el Capítulo 2 se estudian los conceptos relacionados con la Ingeniería de Software, tales como las metodologías de desarrollo y el ciclo de vida del software.

Como Capítulo 3, se realiza la evaluación tecnológica de las diferentes herramientas de desarrollo, así como de los sistemas operativos, bases de datos y navegadores de Internet. Con esta evaluación y con las condiciones de la FI se determinan las herramientas y plataforma a utilizar en la construcción del sistema.

El Capítulo 4 presenta el análisis y diseño de la aplicación. Este capítulo contiene las especificaciones funcionales, el análisis del dominio del problema, el diseño de la base de datos, y el diseño de los componentes de software.

La construcción del sistema se describe en el Capítulo 5, en el cual se muestra el análisis del tiempo empleado en la construcción del sistema, así como el desglose de los componentes que conforman la aplicación.

El Capítulo 6 concentra los resultados de las diferentes pruebas realizadas tanto por el equipo de trabajo de la presente tesis como por los usuarios del sistema, así como las correcciones realizadas al sistema en cada tipo de prueba.

En el Capítulo 7 se documentan los pasos a seguir para la instalación de la aplicación, así como también se definen los tipos de mantenimiento aplicables al sistema.

Posteriormente se presentan las Conclusiones obtenidas del desarrollo de este trabajo.

Como documentación adicional del seguimiento del proyecto, se presenta el Apéndice A, en el cual se encuentran las minutas de las reuniones realizadas con los usuarios del sistema.

Con el fin de brindar una base breve pero concisa de uno de los temas más importantes en la elaboración de la presente tesis, se presenta el Apéndice B, el cual contiene una descripción básica del UML y sus principales diagramas.

Adicionalmente, se muestra el Glosario de términos, cuyo objetivo es brindar al lector una fuente de información básica, en la cual, puede disipar dudas sobre los términos técnicos empleados en la presente tesis.

Finalmente se presentan las Referencias Bibliográficas y Electrónicas utilizadas como fuentes de información.

Estudio Preliminar

CAPÍTULO 1

ESTUDIO PRELIMINAR

1.1 Antecedentes

Los procesos de registro y realización de Servicio Social (SS) en la Facultad de Ingeniería (FI) tienen muchas actividades que consumen tiempo y recursos que posiblemente pueden disminuirse, esto es, el alumno debe visitar las escuelas y dependencias en las cuales tiene la posibilidad de realizar el SS para conocer los proyectos en los cuales puede participar y posteriormente acudir a la Coordinación de Servicio Social de la División de Ingeniería (CSSDI) correspondiente a su carrera para realizar los trámites subsecuentes que constituyen el ejercicio del SS desde su registro hasta su liberación. Además, la Coordinación de Servicio Social de la Facultad de Ingeniería (CSSFI) recibe la información que cada una de las Divisiones de Ingeniería le envía, lo cual obliga a la captura, clasificación y administración manual de la documentación de los alumnos. Todo lo anterior, origina actividades que requieren de tiempo que puede ahorrarse si los alumnos y administrativos tienen la posibilidad de realizar algunas de estas actividades con ayuda de una aplicación que les permita conocer el avance o estado del proceso de SS de forma automatizada.

Por otro lado, tanto la CSSFI como la CSSDI, realizan la administración de los documentos de una manera tradicional, ya que se archivan en carpetas y/o cajones de tal manera que la búsqueda de documentos resulta ser un proceso poco eficiente ya que requiere de mucho tiempo, con la aplicación, se pretende que ambas coordinaciones realicen la administración de documentos de manera electrónica con el objetivo de facilitar esta tarea.

Debido al crecimiento y desarrollo de los sistemas de información, los procesos administrativos han mejorado enormemente tanto en las empresas, escuelas e industrias, lo cual las hace más competitivas y eficientes al tener la posibilidad de contar con una administración automatizada haciendo uso de todas las herramientas que la Ingeniería de la Computación puede brindar.

La eficiencia, rapidez y vanguardia en procesos administrativos deben ser las principales características que el Departamento de Administración Escolar de la FI (DAEFI) debe tener, por lo cual es importante desarrollar aplicaciones informáticas que ayuden a mejorar dichos procesos administrativos para facilitar el mejor desempeño de la planta laboral, así como impulsar el desarrollo tecnológico de la FI.

Como parte de la transformación y automatización de procesos que el DAEFI debe experimentar, el desarrollo de este sistema pretende colaborar con la automatización de las tareas que se desempeñan tanto por los alumnos como por las unidades administrativas correspondientes al brindarles una aplicación en ambiente *Web* que tiene como objetivo mejorar, en la medida de lo posible, las funciones realizadas por cada una de las personas que intervienen en la realización y liberación del SS de los alumnos.

1.2 Definición del Problema

Los requerimientos del sistema estuvieron determinados por el Coordinador de SS de la Facultad de Ingeniería y los diferentes Coordinadores de SS de cada División de Ingeniería de la Facultad. Los requerimientos de los coordinadores, surgieron a partir de entrevistas y visitas. Se entrevistó al Coordinador de SS de la Facultad de Ingeniería, al Coordinador de la División de Ingeniería Civil Topográfica y Geodésica así como al Coordinador de SS de la División de Ingeniería Mecánica e Industrial.

El principal problema se ve encaminado a la centralización de la información referente a los trámites de SS de los alumnos de las diferentes Divisiones de Ingeniería de la Facultad, ya que cada división cuenta con sistemas aislados que les permiten llevar un control de los alumnos que están realizando el SS, así como un sistema central que lo administra el Departamento de Administración Escolar, en el cual se captura de nuevo la información del alumno que le envía cada división. Dicha información llega al DAEFI por medio de mensajería interna, lo que hace que el trámite sea un poco más lento para cada una de sus fases.

Otra problemática, es la gran cantidad de consumo de papel para la emisión de los diferentes formatos que se utilizan para llevar a cabo el alta, entrega de reportes y terminación del SS, lo cual es en su totalidad un gasto realizado por la FI.

Ante esta problemática se llegó a la conclusión de la necesidad de un sistema en el que se mantenga concentrada la información de los alumnos, así como llevar el control de cada alumno que realiza su SS, lo que permite dar seguimiento y control al trámite de los alumnos durante el ejercicio del SS. Este sistema reemplazará en gran medida el consumo de papel para la FI ya que por medio del sistema el alumno deberá llenar los formatos necesarios e imprimirlos en el caso que sea necesario.

El sistema a desarrollar fue denominado “Sistema de Administración del Servicio Social de la Facultad de Ingeniería” (SASS).

1.3 Necesidades del Cliente

Con el propósito de identificar las necesidades del cliente, se tuvieron una serie de entrevistas con los coordinadores de SS, en las cuales se formularon una serie de cuestionarios que permitieron conocer paso a paso este trámite. Estos cuestionarios nos ayudaron en gran medida a descubrir los requerimientos de los coordinadores, así como para planear la solución más adecuada que cubriera sus necesidades.

La elaboración de dichos cuestionarios se realizó por fases, las cuales se fueron dando conforme íbamos conociendo más a fondo los procesos.

El cuestionario que tomamos como primera fase nos ayudó a conocer de manera general en qué consiste el trámite de SS, quienes intervienen y si existe algún sistema de cómputo que les ayude actualmente en el control de los procesos. Para cumplir este objetivo, utilizamos el cuestionario denominado como Cuestionario Fase I el cual se presenta a continuación.

1.3.1 Cuestionario Fase I

1. ¿Cuál es el procedimiento que debe seguir un alumno de la FI para efectuar el SS?
2. ¿Qué requisitos debe cumplir un alumno para registrar el SS?
3. ¿Qué requisitos debe cumplir un alumno para terminar el SS?
4. ¿Cuáles son las fuentes de información donde los alumnos pueden consultar los programas y dependencias en las cuales pueden realizar el SS?
5. ¿Cuáles son las autoridades responsables de controlar los trámites del SS?
6. ¿Cuál es el periodo máximo para que un alumno realice el SS?
7. ¿Existe algún tipo de incentivo otorgado a los alumnos por la práctica del SS?
8. ¿Quién establece los horarios para la prestación del SS?
9. ¿Existe algún sistema que lleve el control de los alumnos que están realizando el SS en la FI, y si existe, mencione cuáles son los procesos dentro del trámite de SS que abarca dicho sistema?

Para responder estas preguntas, se elaboró una minuta a la que denominamos Minuta Fase I, la cual puede ser consultada en el Apéndice A de este trabajo.

Con base en la minuta que surgió como resultado del Cuestionario Fase I se procedió a elaborar un segundo cuestionario al que denominamos Cuestionario Fase II, el cual nos permitió dar un seguimiento sobre los módulos que comprende el Sistema de Administración de Servicio Social (SASS) actual, así como comprender la manera en que se registran los programas de SS a partir del Sistema de Administración de Servicio Social desarrollado por la Dirección General de Administración Escolar (DGAE). El Cuestionario Fase II se presenta a continuación.

1.3.2 Cuestionario Fase II

1. ¿Cuáles son y en qué consisten los módulos que comprende el Sistema de Administración de Servicio Social actual?
2. ¿Cómo puede obtenerse la información de los programas de SS que se encuentran dados de alta?
3. ¿Quién autoriza al alumno para poder iniciar el SS?
4. ¿En dónde deben entregarse los reportes bimestrales que los alumnos realizan sobre su SS?
5. ¿Quién decide si el reporte bimestral es correcto o no?
6. ¿Cuál es el tiempo máximo que puede pasar para que un alumno entregue su reporte bimestral?
7. ¿Quién decide si el alumno culminó de manera satisfactoria el SS?
8. ¿Quién entrega la carta de liberación al alumno?
9. ¿Cuánto tiempo debe pasar para que el alumno pueda recoger su carta de liberación?

Al igual que con el primer cuestionario, se elaboró una minuta a la que denominamos Minuta Fase II, la cual contiene las respuestas a estas preguntas. Esta minuta puede consultarse en el Apéndice A de esta tesis.

Dadas las necesidades presentadas, y ya que en los procesos del trámite de SS existe la interacción entre alumnos y coordinadores, se decidió desarrollar la aplicación en un ambiente de Internet, y así permitirle tanto al alumno como al coordinador darle seguimiento a los procesos

que les compete a cada uno de ellos sin tener la necesidad de acudir personalmente a la FI sólo para conocer su estatus en los procesos de SS.

1.4 Identificación de los Sistemas Cooperativos

Un aspecto importante en el estudio preliminar, es saber si el sistema que se va a construir interactuará o recibirá información de algún sistema ya existente, en este caso el Sistema de Administración de Servicio Social (SASS) recibirá datos provenientes del Sistema de Administración de Servicio Social (SIAS) de la DGAE.

La base de datos del sistema SIAS servirá como fuente de información para los catálogos de Institución, Dependencia, Área, División, Carrera, Coordinador, Responsable, Programa, Actividad y Carrera_Programa que serán utilizados en el SASS.

En el SIAS existirá un proceso que descargará la información de dichos catálogos en archivos de texto, los cuales serán transferidos posteriormente al servidor del SASS. Una vez depositados los archivos se cargarán a la base de datos del SASS.

El Diagrama de Sistemas Cooperativos es el que a continuación se muestra:

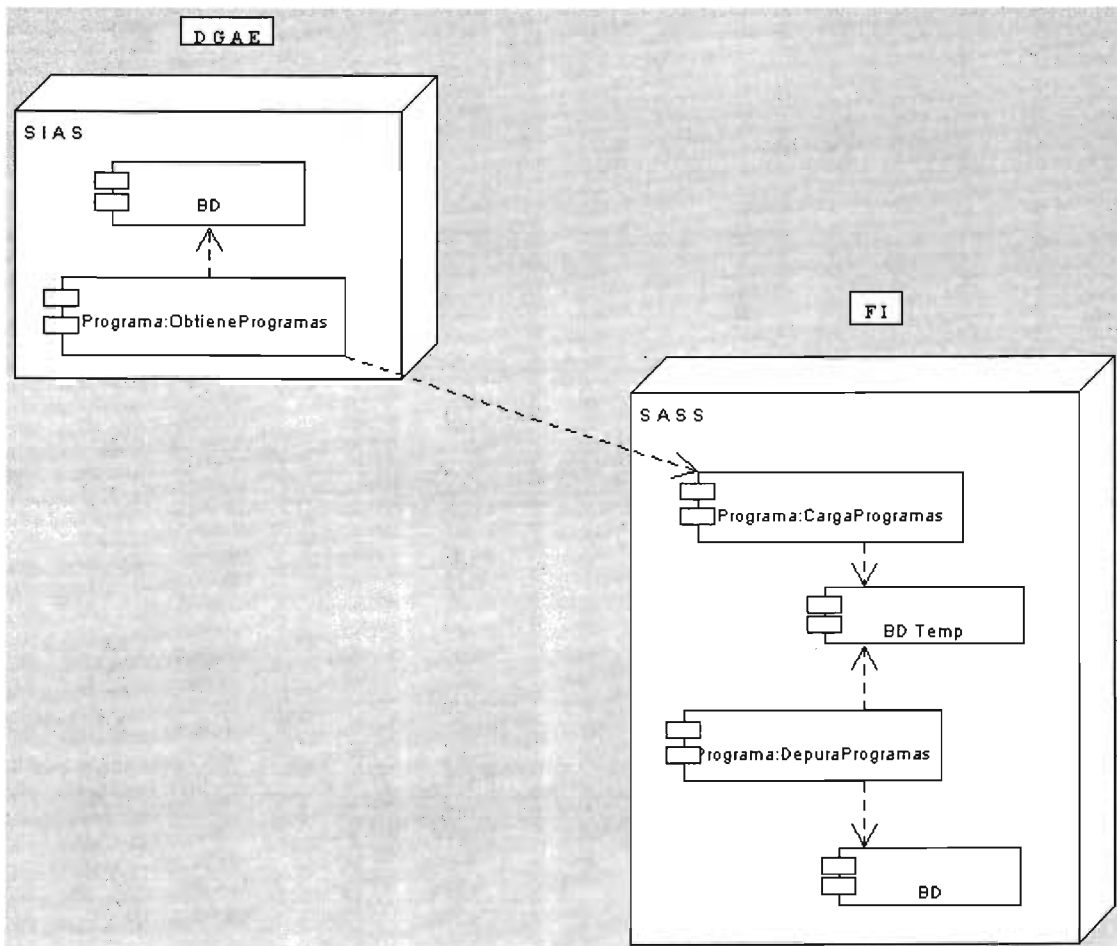


Figura 1.1 Diagrama de Sistemas Cooperativos.

1.5 Recursos disponibles y recursos que se obtendrán

Dados los requerimientos del sistema, de una aplicación que se utilice a través de Internet tanto para los alumnos de la FI como para las autoridades responsables de este proceso, se requerirá de una infraestructura de red interna de la FI que permita dar este servicio.

Ya que el sistema que se propuso es la actualización de un servicio para el Departamento de Servicios Escolares de la Facultad de Ingeniería y gracias a que esta ya cuenta con la infraestructura necesaria para la implementación de este tipo de aplicaciones y por decisión del Departamento de Servicios Escolares de la Facultad de Ingeniería, se decidió utilizar la infraestructura existente, la cual se encuentra regulada por la Unidad de Servicios Académicos (USECAD).

Debido a que la USECAD es la entidad encargada de proveer las facilidades para montar este sistema, el personal correspondiente de esta unidad establece contacto con el administrador de la red para asignación de espacio en disco, tanto en un servidor que provee el servicio de base de datos como un servidor que proporciona el servicio de Web.

Las divisiones de Ingeniería de la FI ya cuentan con la infraestructura de red para acceder a Internet, pero algunas unidades de SS de cada división no cuentan con tarjetas de red en sus máquinas para acceder a Internet, por lo que para el sistema, será necesario agregar las tarjetas de red a las computadoras de las personas responsables del trámite.

1.6 Estudio de Factibilidad

1.6.1 Factibilidad Técnica

La infraestructura tecnológica con la que cuenta la USECAD, otorga la facilidad para implantar el sistema, al contar tanto con los servidores y configuraciones de los mismos, así como contar con la infraestructura de red disponible para acceder al servicio desde Internet.

Las tarjetas de red necesarias para las coordinaciones de SS fueron evaluadas por gente de la misma División de Ingeniería correspondiente para decidir el modelo y características técnicas de dichas tarjetas.

1.6.2 Factibilidad Económica

El aspecto económico del proyecto se encuentra definido básicamente por los siguientes factores:

- ◆ *Factor Humano*: La aplicación es desarrollada como un proyecto de tesis con el único objetivo de aportar algo benéfico para la FI y para la comunidad estudiantil por lo que el desarrollo de esta aplicación no contempla ninguna remuneración económica para el equipo de trabajo.

- ◆ *Factor Tecnológico:* En lo concerniente al hardware y software, no se requiere hacer gasto alguno para la adquisición de equipo y software, puesto que la USECAD proporciona el equipo y software necesario para la puesta en producción del proyecto.

Cabe mencionar que las herramientas de desarrollo propuestas se distribuyen de manera libre en Internet, por lo que tampoco se genera gasto alguno.

Con respecto a las tarjetas de red, las coordinaciones de SS ya cuentan con el presupuesto para adquirir estos dispositivos.

1.6.3 Factibilidad Operacional

1.6.3.1 Beneficios Tangibles

Tiempo

- ◆ El tiempo que dedica el alumno en acudir a la CSSDI que le corresponde para recoger los formatos e inscribirse al trámite de SS, consultar el estado de su trámite y fechas de entrega de reportes es un factor que se reduce con el desarrollo de la aplicación.
- ◆ El tiempo de envío de formatos de altas y bajas de alumnos de una CSSDI a la CSSFI es un factor que se anula con la funcionalidad del nuevo sistema.
- ◆ El tiempo de respuesta de las autoridades correspondientes para deliberar la liberación de un trámite de SS también se reduce.
- ◆ Agilizar el proceso de impresión de cartas de liberación de SS.

Eficiencia

- ◆ La disponibilidad de la información de la base de datos para administrativos, es una ventaja que resultará en el beneficio y mejor desempeño de las labores del personal administrativo.
- ◆ El mejoramiento del proceso de generación e impresión de cartas de liberación de SS.
- ◆ La difusión de los programas de SS es otro factor que se verá beneficiado.

1.6.3.2 Beneficios Intangibles

Organización de los trámites

- ◆ Los procesos del trámite de SS quedarán más claros de acuerdo a los reglamentos y políticas de la CSSFI.
- ◆ Los alumnos de la FI podrán conocer el estatus de su proceso de SS sin necesidad de acudir personalmente a las oficinas de la Administración Escolar, ya que el sistema les permitirá consultar su avance en el proceso. Esta característica beneficia también a los Coordinadores de SS de las Divisiones de Ingeniería y al Coordinador de SS de la FI, ya que no ocuparán más tiempo en hacer una revisión manual del estatus del alumno, cuando éste acude preguntar a las oficinas de la Facultad.

Imagen de la Coordinación de Servicio Social de la Facultad de Ingeniería.

- ◆ La CSSFI y las CSSDI resultan beneficiadas al automatizar sus procesos de trámite, lo que dará buenas referencias a la FI en cuanto al manejo de la información de forma más eficiente y confiable.

Conceptos Básicos

CAPÍTULO 2

CONCEPTOS BÁSICOS

2.1 Ingeniería de Software

La Ingeniería de Software es un tema tan extenso y abundante, que tratar de abordarlo detalladamente resultaría una labor difícil de desarrollar, que además, no se encuentra dentro de los objetivos del presente trabajo. Por lo anterior, se describirán a continuación algunos de los aspectos más importantes con la intención de resaltar la importancia de su aplicación en la solución o mejoramiento de problemas.

El desarrollo de *software* no es una tarea fácil ya que debido a la complejidad de los sistemas informáticos que se demandan, se hace a veces necesario el desarrollo de proyectos software de miles de líneas de código. Esto no puede ser abordado directamente, empezando a programar sin más, si no que es necesario analizar qué y cómo se va a hacer, cómo se van a coordinar todas las personas que van a intervenir en el proyecto y cómo se controlará el desarrollo del mismo de forma que al final se obtengan los resultados esperados.

2.1.1 El Ciclo de Vida del Software

El Ciclo de Vida del Software es el conjunto de actividades que se realizan durante el proyecto de desarrollo de software, intentando determinar el orden de las etapas involucradas, así como los criterios de transición asociados entre las mismas.

Un modelo de ciclo de vida describe las fases principales del desarrollo, ayuda a administrar el progreso del proyecto y provee un espacio de trabajo para la definición detallada del proceso de desarrollo.

Existen diferentes representaciones y composiciones del ciclo de vida del software, sin embargo las etapas comunes y el orden de ocurrencia son:

1. Estudio Preliminar
2. Análisis
3. Diseño
4. Implementación
5. Pruebas
6. Mantenimiento

2.1.2 Definición y Objetivos de la Ingeniería de Software

La ingeniería de software difiere de la programación tradicional en que se utilizan técnicas de ingeniería para especificar, diseñar, instrumentar, validar y mantener los productos dentro del tiempo y el presupuesto establecidos.

La ingeniería de software es la aplicación de técnicas de ingeniería y administración al proceso de crear productos de software utilizando metodologías, herramientas y procesos.

Las actividades que se desempeñan en la ingeniería de software están regidas por la aplicación de tres elementos importantes:

1. *Metodología*: Incluye la identificación de las necesidades, planificación y estimación de proyectos, sistemas de calidad, análisis de requerimientos, métodos de diseño, de codificación y de pruebas, así como el mantenimiento de productos.
2. *Procesos*: Un proceso de ingeniería de software es una definición del conjunto completo de actividades necesarias para transformar los requisitos de un usuario en un producto de software. Definen la documentación de cada una de las fases de desarrollo y los controles que aseguren la calidad del producto a lo largo del proyecto.
3. *Herramientas*: Son sistemas automáticos que apoyan la aplicación de las metodologías en el desarrollo del producto. Incluyen herramientas como las llamadas CASE (Computer-Aided Software Engineering), de control automático de configuración, ensayo automático de software, bases de datos de registro, lenguajes de programación, sistemas operativos, estructuras de red, computadoras, entornos de desarrollo, etc.

Entre los principales objetivos de la ingeniería de software se encuentran:

- ◆ Construir un producto de software o mejorar uno existente.
- ◆ Mejorar la calidad del producto que se entrega al cliente.
- ◆ Aumentar la productividad de los ingenieros del software.
- ◆ Asegurar la eficiencia del proceso de desarrollo y mantenimiento del producto.
- ◆ Fomentar la reusabilidad de los componentes de software.
- ◆ Mejorar la planificación de los proyectos.

En la industria del software, el control y la calidad de los procesos para aumentar la calidad del producto está convirtiéndose en una norma, por lo que la necesidad de contar con una guía para el mejoramiento del proceso de desarrollo se convierte en algo imprescindible. Como una posible solución a esta necesidad, surge el Modelo de Madurez de Capacidades (CMM) el cual propone una serie de lineamientos orientados a mejorar los procesos de control y desarrollo de proyectos para las organizaciones que se dedican al desarrollo de software y buscan alcanzar o mantener un lugar dentro del mercado competitivo en el cual el concepto de calidad es fundamental.

2.1.3 El Modelo de Madurez de Capacidades: CMM (Capability Maturity Model)

El CMM es un *modelo* de prácticas fundamentales que deben ser implementadas por toda organización dedicada al desarrollo de software que esté interesada en mejorar la calidad de sus productos, así como su productividad.

Este modelo, desarrollado por el Software Engineering Institute (SEI) de la Universidad de Carnegie Mellon, se basa en conceptos de calidad total y de mejoramiento continuo y ha sido ampliamente aceptado por la comunidad de ingeniería de software.

El modelo de madurez de capacidades sirve como una guía para el mejoramiento del desarrollo y mantenimiento de las prácticas de software. Consiste en prácticas claves que

expresan métodos óptimos para producir software de calidad, con aumento de productividad, con un presupuesto y fechas previstas [1].

Además, el CMM se utiliza no sólo como un manual de prácticas recomendables, sino que además como referencia para llevar a cabo auditorías y evaluaciones internas en las organizaciones que producen y mantienen software.

2.1.3.1 Arquitectura del CMM

La arquitectura del CMM tiene la forma de una jerarquía que contiene cinco niveles de madurez. Cada nivel puede subdividirse en un número variable de áreas de proceso clave, conocidas a través de la documentación del CMM por su sigla inglesa KPA (key process area), cada una de éstas comprende 5 categorías de prácticas llamadas características comunes, finalmente se encuentran las prácticas claves, las cuales pueden encontrarse en todas las categorías de características comunes. Estas prácticas constituyen atributos que indican si la implementación y la institucionalización de un proceso clave son efectivas y duraderas.

Los cinco niveles del CMM son:

1. *El nivel 1 ó inicial:* Es el estado primario en la evolución de las organizaciones que desarrollan software. En este nivel se encuentran todas las empresas que no han logrado implementar las prácticas básicas de gestión de proyectos o de ingeniería de software definidas a partir del nivel 2.

Una empresa se encuentra en el nivel inicial cuando los proyectos que desarrolla no se pueden planear y los requerimientos no se tienen bajo control, cuando no se esté en condiciones de controlar las versiones del producto, cuando la calidad es percibida como un burocracia innecesaria y se acepte que los procesos son una cosa personal, donde cada individuo sigue un proceso propio de tal manera que sus desarrollos difícilmente pueden ser entendidos por otra persona.

2. *El nivel 2:* En este nivel es posible la implantación de prácticas mínimas de administración de proyectos, de control de requerimientos, así como de versiones de producto.

El nivel 2 no garantiza que todos los proyectos dentro de la organización estén al mismo nivel de madurez, sino que algunos pueden estar aún en el nivel inicial. Este nivel identifica prácticas que son aplicables a todo tipo de organizaciones de desarrollo de software y la ausencia de cualquiera de éstas pone en peligro el éxito de la empresa. El mayor beneficio obtenido de la implantación de este nivel, es la planificación realista de los proyectos.

3. *El nivel 3:* Este es el estado en el que la empresa ha definido un conjunto de procesos, metodologías y herramientas comunes a todos los proyectos de la organización. El proceso generalizado para todos los proyectos debe estar suficientemente documentado en una biblioteca accesible a todos los desarrolladores, además, de que todo el personal debe recibir el entrenamiento necesario para la comprensión de este proceso estándar. Otra característica importante, es la existencia de pautas y criterios definidos para adaptar dicho proceso a las necesidades y características propias de cada proyecto.

Entre las principales ventajas de implantar el nivel 3, se encuentran los mecanismos de comunicación entre proyectos y departamentos, lo que garantiza una visión común del producto y una rápida acción para enfrentar problemas; la comparación entre proyectos pasados y en curso, permite planificar y comparar el rendimiento, además el riesgo de depender en individuos irremplazables es baja.

4. *El nivel 4:* En este nivel, la organización se encuentra en condiciones de medir la calidad del producto y del proceso de software, ambos de forma cuantitativa y mediante métricas detalladas. Estas mediciones permiten detectar cuando las variaciones del rendimiento se salen de los rangos aceptables, de manera que es posible tomar medidas correctivas que permitan asegurar la calidad tanto del producto como del proceso de desarrollo.

Los beneficios que una corporación obtendrá al implantar este nivel son: la medición de la productividad y calidad de los procesos de software a través de todo el proyecto, la empresa puede establecer metas cuantitativas para la calidad de los productos y de los procesos de software, además de que es posible controlar la variación del rendimiento de los productos y procesos para mantenerla dentro de las fronteras cuantitativas aceptables.

5. *El nivel 5:* Este es el nivel optimizado en el que la característica principal es el mejoramiento continuo del proceso con base en la realimentación cuantitativa y la práctica de nuevas ideas y tecnologías innovadoras.

Al implantar el nivel 5, las organizaciones se benefician principalmente en que cuentan con los medios para identificar las debilidades y reforzar el proceso con objeto de prevenir la ocurrencia de errores. Los datos relativos a la eficacia del proceso de software se usan para analizar el costo y el beneficio de usar nuevas tecnologías y de implementar cambios al proceso de software.

En este nivel, los proyectos analizan los defectos para determinar sus causas y así mismo prevenir que los defectos conocidos vuelvan a ocurrir, además, las lecciones aprendidas son difundidas a otros proyectos.

2.2 Tipos de Metodologías

2.2.1 Definición y Objetivos de la Metodología

Una *metodología* de desarrollo es un conjunto de técnicas, fases, procedimientos, reglas, herramientas y documentación que indican cómo desarrollar un proyecto de software.

Este conjunto de fases se descompone en subfases, las cuales guían a los desarrolladores en la elección de las técnicas que debe elegir para cada estado del proyecto, y facilita la planificación, gestión, control y evaluación de los proyectos.

Los objetivos de una metodología de desarrollo son:

- ♦ Registrar los requisitos de un sistema de información de forma acertada.
- ♦ Proporcionar un método sistemático de desarrollo de forma que se pueda controlar su progreso.

- ◆ Construir un sistema de información dentro de un tiempo apropiado y unos costos aceptables.
- ◆ Elaborar la documentación del sistema detalladamente y que sea fácil de mantener.
- ◆ Proporcionar un sistema que satisfaga a todas las personas afectadas por el mismo, ya sean clientes, directivos, auditores o usuarios.
- ◆ Garantizar la calidad de la aplicación.
- ◆ Promover la creación de un proceso estándar en la organización.

Una metodología puede seguir uno o varios modelos de ciclo de vida. El ciclo de vida indica qué es lo que hay que obtener a lo largo del desarrollo del proyecto, mientras que la metodología indica el cómo.

2.2.2 Clasificación de las Metodologías

Las metodologías se clasifican en dos tipos de acuerdo al modo en que el sistema a desarrollar es analizado o modelado. Estos tipos de metodologías son las estructuradas y las orientadas a objetos; ambas se describen a continuación.

2.2.2.1 Metodologías Estructuradas

Estas metodologías, examinan los sistemas desde el punto de vista de las funciones o tareas que deben realizar, tareas que se van descomponiendo sucesivamente en otras tareas más pequeñas y que forman los bloques o módulos de las aplicaciones. Además, las metodologías estructuradas proponen la creación de modelos del sistema que representan los procesos, los flujos y la estructura de los datos de una manera descendente, dando lugar a metodologías orientadas a procesos y orientadas a datos (tanto a datos jerárquicos como a datos no jerárquicos).

Metodologías Orientadas a Procesos

Los datos se introducen en el sistema y el sistema responde ante ellos transformándolos para obtener las salidas. Una especificación estructurada es un modelo gráfico, particionado, descendente y jerárquico de los procesos del sistema y de los datos utilizados por los procesos.

Estas metodologías tienen como elementos primordiales:

1. *Diagramas de flujos de datos (DFD)*: Representan los procesos que debe llevar a cabo un sistema a distintos niveles de abstracción y los datos que fluyen entre las funciones. Los procesos más complejos se descomponen en nuevos diagramas hasta llegar a procesos sencillos.
2. *Diccionario de datos*: Es el conjunto de definiciones de todos los datos que aparecen en el DFD, tanto almacenados como en los flujos de datos.
3. *Especificaciones de proceso*: Describen con más detalle lo que ocurre dentro de un proceso.

Metodologías Orientadas a Datos Jerárquicos

Se orientan más a las entradas y salidas. Se definen las estructuras de datos y a partir de éstas se derivan los componentes procedimentales.

En las metodologías orientadas a datos jerárquicos, la estructura de control del programa debe ser jerárquica y se debe derivar de la estructura de datos del programa. El proceso de diseño consiste en definir, primero, la estructura de los datos de entrada y salida, mezclarlas todas en una estructura jerárquica de programa y después ordenar detalladamente la lógica procedimental para que se ajuste a esta estructura. Adicionalmente, el diseño lógico debe preceder y estar separado del diseño físico.

Metodologías Orientadas a Datos no Jerárquicos

Los datos constituyen el corazón del sistema de información, son más estables que los procesos que actúan sobre ellos. El estudio de los procesos viene derivado de una definición inicial de los datos (modelo de datos) constituido por el conjunto de entidades de datos básicas y las interrelaciones entre ellas.

Metodologías más Representativas de la Programación Estructurada

Los ejemplos más representativos de las metodologías estructuradas y los pasos que propone cada una de ellas son:

- ◆ *Metodología de De Marco:*
 1. Estudio del entorno físico actual: se comienza haciendo un modelo del sistema actual, realizando un conjunto de DFD's físicos actuales.
 2. Derivación del correspondiente modelo lógico actual.
 3. Derivación del nuevo modelo lógico: se toman en consideración las nuevas necesidades de los usuarios, obteniendo una especificación estructurada.
 4. Crear un conjunto de modelos físicos alternativos.
 5. Valorar cada opción.
 6. Seleccionar una opción.
 7. Empaquetar la especificación.

- ◆ *Metodología de Yourdon/Constantine:*
 1. Realizar los DFD del sistema.
 2. Realizar el diagrama de estructuras, obteniéndolo a partir de los DFD's mediante dos técnicas, al análisis de transformación y el análisis de transacción.
 3. Evaluación del diseño, midiendo la calidad de la estructura resultante mediante el acoplamiento y la cohesión.
 4. Preparación del diseño para la implantación (cuadernos de carga).

2.2.2.2 Metodologías Orientadas a Objetos

En las metodologías orientadas a objetos, el modelado del sistema, es un aspecto que cobra mucha más importancia, de tal manera, que el dominio del problema se examina como un conjunto de objetos que interactúan entre sí. A diferencia de las metodologías estructuradas, en las cuales existe una división entre las funciones que llevan a cabo los programas y los datos que se almacenan en las bases de datos, en las metodologías orientadas a objetos existe un enfoque unificador de ambos aspectos, los cuales se encapsulan en los objetos.

Existen dos enfoques en las metodologías orientadas objetos, las cuales se describen a continuación.

Metodologías Revolucionarias o Puras

Extienden la orientación al objeto como un cambio profundo, que convierte a las metodologías estructuradas en obsoletas. Una metodología representativa de este enfoque es OOD de Booch.

Metodologías Sintetistas o Evolutivas

Piensen que el análisis y diseño estructurado, constituyen la base para el desarrollo orientado al objeto, pudiéndose combinar elementos del análisis y diseño estructurado con los de orientación al objeto. En este enfoque podemos distinguir como ejemplo a la metodología OMT desarrollada por Rumbaugh.

Además de estos enfoques, han surgido nuevas metodologías denominadas como “segunda generación”, que toman como base las dos corrientes anteriores, entre las cuales, es posible mencionar: FUSION propuesta por Coleman, OOA/D de Booch, MOSES de Henderson-Sellers y Edwards, SYNTROPY de Cook y Daniels, MÉTODO UNIFICADO Booch, Rumbaugh, y Jacobson, y MEDEA de Piattini.

Metodologías más Representativas de la Programación Orientada a Objetos

A continuación, se explicarán de forma general, tres de las metodologías más utilizadas en la orientación a objetos.

- ♦ *Técnica de Modelado por Objetos (OMT, Object Modeling Technique)*

Esta metodología creada por Rumbaugh, cumple las fases de análisis, diseño e implantación, de acuerdo a las siguientes etapas:

1. *Análisis*: Se construye un modelo de la situación del mundo real mostrando las propiedades más importantes. El modelo de análisis, es una abstracción precisa y concisa de que es lo que debe de hacer el sistema deseado y no de como debe de hacerlo. Los objetos en el modelo, deben de ser del dominio del problema y no conceptos del dominio del software.

2. *Diseño del Sistema*: Se toman las decisiones de alto nivel acerca de la arquitectura total. Durante esta etapa, el sistema a desarrollar es organizado en subsistemas basados tanto en la estructura del análisis, como en la arquitectura propuesta. El diseñador del sistema, debe decidir qué características de desempeño optimizar, escoger una estrategia para atacar el problema y hacer asignaciones de recursos tentativas.
3. *Diseño de Objetos*: En esta fase, se construye un modelo de diseño, basado en el modelo de análisis pero conteniendo detalles de implementación. El enfoque del diseño de objetos, son las estructuras de datos y los algoritmos necesarios para implementar cada clase. Las clases de objetos del análisis siguen siendo útiles, pero son aumentadas con estructuras de datos y algoritmos del dominio del software seleccionado para optimizar importantes medidas de desempeño.
4. *Implementación*: Las clases de objetos y las relaciones desarrolladas durante el diseño de objetos son traducidas finalmente a una implementación en un lenguaje de programación, base de datos o hardware particular. Durante la implementación, es importante el seguir una buena práctica de la ingeniería de software de tal forma que el seguimiento del diseño sea directo y que el sistema implantado permanezca flexible y extensible.

La metodología OMT, utiliza tres tipos de modelos para describir un sistema: el modelo de objetos, describiendo los objetos en el sistema y sus relaciones; el modelo dinámico, describiendo las interacciones entre los objetos del sistema; y el modelo funcional, describiendo las transformaciones de los datos del sistema. Cada modelo es aplicable durante todas las etapas de desarrollo y adquieren detalles de implementación conforme avanza el desarrollo. Una descripción completa de un sistema requiere los tres modelos.

♦ *El Método de Booch*

Este método ofrece una gran libertad en la producción del software, de forma muy general, cubre las fases de análisis y diseño de acuerdo a lo siguiente:

1. *Análisis*: Se realiza el modelado descubriendo las clases y objetos que forman parte del dominio del problema.
2. *Diseño*: Se aplicarán las abstracciones y los mecanismos necesarios para conseguir el comportamiento que exige el modelo establecido en la fase de análisis.

Booch, propone varias formas de describir un sistema en orientación al objeto, utilizando una variedad de diagramas, los cuales se explican a continuación:

Diagrama de clases: Este diagrama es estático y muestra las relaciones entre las abstracciones de un sistema. En este diagrama se representan las clases, con sus atributos y métodos, así como las relaciones entre las clases.

Diagrama de objetos: Es parte de la notación del diseño orientado a objetos. Muestra el comportamiento dinámico del sistema, por lo que en él, se representan todos o algunos de los objetos junto a sus relaciones en el modelo lógico del sistema.

Diagrama de transición de estados: Se utiliza para mostrar el espacio de estados de una clase, los eventos (o mensajes) que disparan una transición de un estado a otro y las acciones que resultan del cambio de estado.

Diagrama de módulos: En este diagrama, se muestra la colocación de clases y objetos en los módulos en el diseño físico de un sistema. Los dos elementos esenciales del diagrama son los módulos y las dependencias.

Diagrama de interacción: Traza la ejecución de un escenario en el mismo contexto que un diagrama de objetos. Permite leer el paso de mensajes de unos objetos a otros en perfecto orden.

En conclusión, la metodología de Booch es muy poco rígida y ofrece libertad al usuario de la misma. La identificación de las fases de análisis y diseño del sistema son responsabilidad del usuario, siempre que se respete la notación y los resultados sean coherentes con los conceptos de orientación a objetos, podrá considerarse que estas fases se han llevado a cabo correctamente.

♦ *Ingeniería de Software Orientada a Objetos (OOSE, Object Oriented Software Engineering)*

Es una metodología que se utiliza fundamentalmente para el modelado de negocios, basándose en la premisa "Se puede ejecutar una reingeniería del sistema de información desde los casos de uso" [2]. Por lo anterior, está basada en los casos de uso de Ivar Jacobson.

El objetivo, es madurar un modelado de negocio representado en un caso de uso para conseguir hacerlo un modelo de objetos robusto. Éste derivará en un modelo de objetos real, implementable y verificable.

Las fases de OOSE y las acciones a realizar en cada una de ellas son:

1. *Análisis de los requisitos para los casos de uso:* En esta etapa, se determinan los actores, los casos de uso y se describen éstos últimos. Se deben mostrar los flujos de eventos, tanto los normales como las excepciones. Para la creación de un caso de uso, se debe asignar un nombre y una descripción, determinar el flujo de los efectos e incluir las precondiciones y las postcondiciones.
2. *Diseño ideal:* Se definen las relaciones entre los objetos, usando la notación definida para OMT, o bien, se usan diagramas de interacción de objetos, definiendo además su comportamiento y asociaciones para agruparlos en módulos.
3. *Diseño real:* Se mantienen las relaciones ya definidas, pero se modifican con la idea de la implementación. Se mantienen las conexiones del diseño ideal para poder ver las conexiones con los requisitos. En esta etapa se utilizan diagramas de interacción de objetos.

4. *Pruebas verificadas*: Se prueba cada clase como si fuera una caja negra y se comprueba que hace lo que estaba estipulado. Además, se pueden probar casos como muestra del comportamiento del sistema en diferentes circunstancias.

2.3 La Unificación de las Metodologías Orientadas a Objetos en un Proceso de Desarrollo y un Lenguaje de Modelado

Debido a que existían múltiples metodologías orientadas a objetos, cada una con su grupo de seguidores, con algunas similitudes, pero también con distintas notaciones o términos, Grady Booch, Jim Rumbaugh e Ivar Jacobson trabajaron en la creación de un proceso y de un lenguaje que unificara las mejores características de cada una de las metodologías individuales de estos autores.

El Proceso Unificado es el resultado de tres décadas de desarrollo y uso práctico. Su desarrollo como producto ha recibido influencias de muchas fuentes, entre las cuales pueden mencionarse las empresas Ericsson y Rational Software Corporation.

En 1996 Grady Booch, Jim Rumbaugh e Ivar Jacobson proponen el anteproyecto del Lenguaje de Modelado Unificado (UML, Unified Modeling Language). El UML es un lenguaje de modelado y es independiente del proceso, por lo que no se considera una metodología.

Cabe mencionar que el proceso y el lenguaje son independientes, ya que pueden adoptarse como una opción de forma conjunta o individual, esto es, UML puede ser utilizado sin que esto signifique la obligación de utilizar el proceso unificado como método de desarrollo. UML debe en parte su gran aceptación a que no incluye un proceso como parte de su propuesta.

2.3.1 El Proceso Unificado de Desarrollo de Software

El Proceso Unificado es un proceso de desarrollo de software y más aún, es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas de software, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitud y diferentes tamaños de proyecto [3].

Una parte esencial del Proceso Unificado es el Lenguaje Unificado de Modelado, ya que éste se utiliza para representar todos los esquemas de un sistema de software.

El Proceso Unificado tiene tres características principales: dirigido por casos de uso, centrado en la arquitectura, e iterativo e incremental. Cada una de estas características se describen a continuación.

1. *Dirigido por casos de uso*: Los casos de uso nos obligan a pensar en términos de importancia para el usuario y no sólo en términos de funciones que sería bueno tener, sin embargo, los casos de uso además de ser una herramienta para especificar los requisitos de un sistema, también guían su diseño, implementación y prueba, es decir, guían el proceso de desarrollo. Con base en el modelo de casos de uso, los desarrolladores crean una serie de modelos de diseño e implementación que llevan a cabo los casos de uso, los ingenieros de pruebas prueban la implementación para garantizar que los componentes del modelo de implementación cubran correctamente

los casos de uso [3]. Por lo anterior, podemos decir que los casos de uso no sólo inician el proceso de desarrollo sino que le proporcionan un hilo conductor. En síntesis, *dirigido por casos de uso* significa que el proceso de desarrollo sigue un hilo, es decir, avanza a través de una serie de flujos de trabajo que parten de los casos de uso, los cuales, se especifican y se diseñan con base en las necesidades de los usuarios.

2. *Centrado en la arquitectura*: El concepto de arquitectura de software incluye los aspectos estáticos y dinámicos más significativos del sistema. La arquitectura surge de las necesidades de la empresa o institución, además de que también se ve influida por otros factores como la plataforma en la que tiene que funcionar el software (arquitectura de hardware, sistema operativo, sistema de gestión de base de datos), bloques de construcción reutilizables de que se dispone, requisitos de implantación, sistemas heredados y requisitos no funcionales (como el rendimiento). La arquitectura es una vista de diseño con las características más importantes resaltadas. Debe existir interacción entre los casos de uso y la arquitectura, ya que los casos de uso deben encajar en la arquitectura cuando se llevan a cabo, así como, la arquitectura debe permitir el desarrollo de todos los casos de uso requeridos. La arquitectura debe diseñarse para permitir que el sistema evolucione.
3. *Iterativo e incremental*: En el desarrollo de sistemas es muy recomendable dividir el trabajo en mini proyectos, los cuales son iteraciones que resultan en el incremento del proyecto general. Las iteraciones hacen referencia a pasos en el flujo de trabajo y los incrementos al crecimiento del producto [3]. Los mini proyectos comienzan con los casos de uso y continúan a través de las siguientes fases de desarrollo (análisis, diseño, implementación y prueba), lo cual, termina convirtiendo en código ejecutable los casos de uso que se desarrollan en la iteración. Es importante recalcar, que un incremento no necesariamente es aditivo, ya que por ejemplo, en las primeras fases de desarrollo se puede reemplazar un desarrollo superficial por uno más detallado. En fases posteriores los incrementos son típicamente aditivos.

Para alcanzar los objetivos planteados, el equipo de trabajo debe intentar seleccionar sólo las iteraciones requeridas, para lo cual, debe realizarlas en un orden lógico. Idealmente, un proyecto debe ejecutarse de una forma directa, sólo con pequeñas desviaciones del curso que los desarrolladores planearon inicialmente, sin embargo, en la medida en que se añaden iteraciones o se altere el orden de las mismas por problemas inesperados, el proceso de desarrollo consumirá más tiempo y esfuerzo.

2.3.2 El Lenguaje Unificado de Modelado (Unified Modeling Language, UML)

El UML es un lenguaje de modelado destinado a los métodos que utilizan conceptos orientados a objetos. El lenguaje de modelado es la notación (principalmente gráfica) que usan los métodos para expresar un diseño.

La estandarización de un lenguaje de modelado es invaluable, ya que es la parte principal de comunicación entre los analistas y desarrolladores, de tal manera, que si se quiere discutir un diseño con alguien más, ambos deben conocer el lenguaje de modelado y no así el proceso o método que se siguió para obtenerlo.

El UML nació en 1994 por iniciativa de Grady Booch y Jim Rumbaugh quienes laboraban en Rational Software Corporation y combinaron sus dos famosos métodos diseñados

individualmente: el de Booch y el OMT (Object Modeling Technique) de Rumbaugh. Un año después, Ivar Jacobson, creador del método OOSE (Object Oriented Software Engineering), ingresó a Rational, uniendo sus esfuerzos a los antes mencionados.

Los anteproyectos del UML (alrededor de 1996) circularon rápidamente en la industria del software generándose así considerables modificaciones. Cuando los diversos corporativos vieron que el UML era útil a sus propósitos, se conformó un consorcio del UML de cuyos miembros puede destacarse a DEC, Hewlett-Packard, Intellicorp, Microsoft, Oracle, Texas Instruments y Rational.

El Object Management Group (OMG), realizó una petición para definir un lenguaje y una notación estándar del lenguaje de construcción de modelos, por lo que en 1997 el consorcio produjo la versión 1.0 del UML y lo propuso como candidato.

El consorcio aumentó y generó la versión 1.1 del UML, misma que puso nuevamente a consideración del OMG. Dado que sus creadores representan métodos muy difundidos de la primera generación del análisis y diseño orientado a objetos, este lenguaje recibió la aprobación del grupo a finales de 1997.

El OMG se encargó de la conservación del UML y produjo otras dos revisiones en 1998. El UML ha llegado a ser el estándar de facto en la industria del software, y su evolución continúa.

El UML estandariza los artefactos y la notación, pero no define un proceso oficial de desarrollo ya que es independiente de la metodología.

Para mayor referencia de UML y sus artefactos, consulte el Apéndice B.

2.4 Tipos de Arquitectura

2.4.1 Arquitectura Lógica de Sistemas

La forma en como se distribuyen los servicios de una aplicación se le conoce como capas. La manera de determinar el número de capas que formarán parte de la solución de un sistema de información depende de que tantos sistemas o subsistemas estén involucrados para obtener la información necesaria para dar solución a una necesidad, o bien, del rendimiento o disponibilidad que deberá proporcionar una aplicación.

2.4.1.1 Modelo de Dos Capas

En una arquitectura a dos capas, la lógica de presentación y lógica del negocio se encuentran formando una capa del sistema, mientras que los servicios de almacenamiento o base de datos constituyen la segunda capa.

En este modelo, los objetos que constituyen la interfaz de usuario tienen embebido el código de las reglas del negocio, lo cual impide la reutilización del código y dificulta el mantenimiento del sistema ya que esta capa constituye el sistema *cliente*, mientras que los servicios de base de datos residen del lado del *servidor* de base de datos.

Es inusual que todas las reglas del negocio prevalezcan del lado del sistema cliente, ya que se pueden emplear otro tipo de servicios o utilerías del manejador de base de datos que den soporte a servicios de lógica del negocio.

2.4.1.2 Modelo de Tres Capas

En una arquitectura a tres capas existe de manera más explícita una división de los servicios de lógica de presentación, lógica de negocio y almacenamiento, cada una de ellas se pueden implementar en diferentes servidores, o bien, utilizar múltiples servidores para los servicios de lógica de negocio y servicios de base de datos.

La descripción clásica de la arquitectura de tres capas es la siguiente:

1. *Presentación*: Ventanas, mensajes, reportes, etcétera.
2. *Lógica del Negocio*: Tareas y reglas que rigen el negocio.
3. *Almacenamiento*: Repositorios de información.

2.4.1.3 Modelo de N Capas

Cuando se analiza un sistema desde la perspectiva de una descomposición más detallada, se habla de una arquitectura multicapas, en la cual se encuentra implícita la capa intermedia de la lógica del negocio.

En un diseño orientado a objetos, se habla de arquitectura de N capas cuando es posible dividir la capa de la lógica de aplicaciones en otras menos densas como son:

1. *Objetos del Dominio*: Objetos que representan los conceptos del dominio.
2. *Servicios*: Objetos que proveen servicios como la interacción con la base de datos, generación de reportes, etcétera.

2.4.1.4 Beneficios para Utilizar Arquitectura de Tres o Más Capas

El uso de una arquitectura de tres o más capas ofrece beneficios entre los cuales se encuentran:

- ◆ Aislamiento de la lógica del negocio en componentes independientes susceptibles de reutilizarse después en otros sistemas.
- ◆ Distribución de las capas en varios nodos físicos de cómputo y en varios procesos, lo cual, puede mejorar el desempeño, la coordinación y el compartir la información.
- ◆ Asignación de tareas por capas, por ejemplo un equipo que trabaje exclusivamente en la capa de presentación, mientras que otros equipos de trabajo con conocimientos especializados en desarrollo se dedican a la construcción de la lógica del negocio y al almacenamiento de los datos.

2.4.2 Arquitectura Física de Sistemas

Una arquitectura lógica, puede desplegarse o implementarse físicamente en varias configuraciones, entre las cuales se encuentran:

- ◆ Capas de la lógica de presentación y del negocio en la computadora del cliente y la capa de almacenamiento o base de datos en un equipo servidor.
- ◆ La presentación en la computadora del cliente, la lógica del negocio en un servidor de aplicación y el almacenamiento en un servidor de datos independiente.

La utilización de múltiples servidores para soportar modelos de arquitectura de N Capas permiten separar los procesos que se encuentran en diferentes dominios, o bien, que se encuentran en regiones geográficas distintas o simplemente para incrementar el rendimiento de la funcionalidad de un sistema de información.

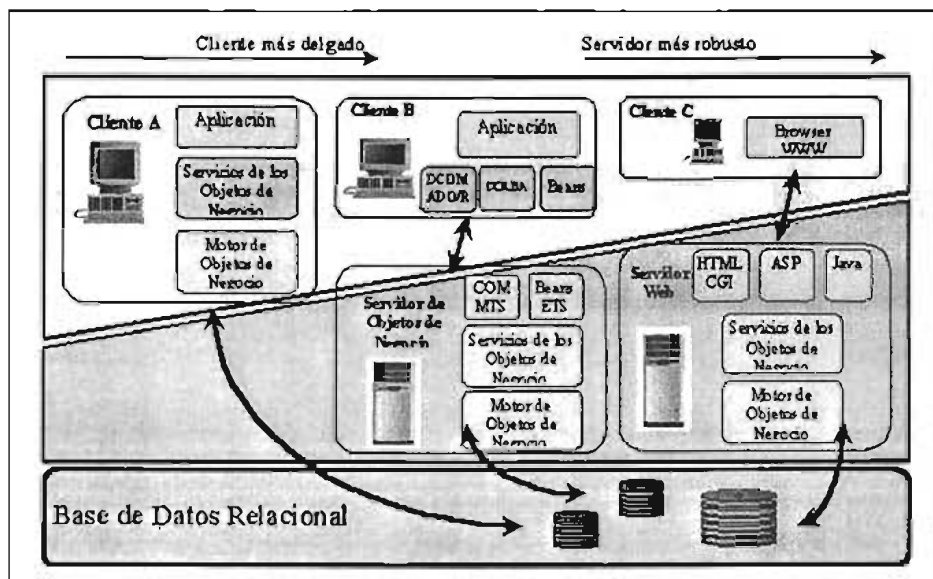


Figura 2.1 Ejemplo de una Arquitectura Física de Tres Capas.

Evaluación Tecnológica

CAPÍTULO 3

EVALUACIÓN TECNOLÓGICA

3.1 Herramientas de Programación en Tecnología Web

3.1.1 Common Gateway Interface (CGI)

Los CGI son un mecanismo para desarrollar páginas Web activas en el servidor. Sirven de interfaz de comunicación entre aplicaciones externas y servidores Web, tales como, procesar la información de formularios, generar páginas con contenidos dinámicos, o mandar y recibir cookies. Los CGI's residen del lado del servidor y dependiendo del servidor de Web, éstos deben residir en una carpeta específica para poder ser ejecutados al momento de ser invocados por la página Web. Cuando esto sucede, el Servidor de Web ejecuta un *proceso* hijo que recibirá los datos que envía el usuario en un formulario (en caso de que los haya), realiza el procesamiento de la información, pone a disposición del mismo algunos datos en forma de variables de ambiente y captura la salida del programa para generar una respuesta, usualmente una página *HTML*. La respuesta del CGI es enviada al servidor de Web, el cual envuelve la respuesta en una respuesta *HTTP*. Posteriormente, la respuesta *HTTP* es enviada de regreso al *navegador*.

Los CGI's pueden programarse en cualquier lenguaje soportado por el sistema operativo del servidor, pero los más utilizados son los siguientes:

Lenguajes de Programación:

- ◆ C/C++
- ◆ Fortran

Lenguajes de Script:

- ◆ PERL
- ◆ Shell's de Unix
- ◆ Visual Basic
- ◆ AppleScript

La diferencia de desarrollar CGI's en lenguajes de programación o en lenguajes de script reside en la forma en como se ejecutará el código, ya que para los lenguajes de programación es necesario compilar el código además de que el servidor Web debe ligar el código fuente con el código compilado, mientras que en los lenguajes de script solo es necesario el código fuente, el cual será interpretado cada vez que se haga la llamada al CGI.

Conexión con Bases de Datos

Pueden emplearse las facilidades que proporciona IDC (Internet Database Connector) para el acceso a bases de datos con *ODBC* (Open DataBase Connectivity).

Ventajas

Los programas CGI pueden ser escritos en una variedad de lenguajes, aunque el más utilizado es PERL. Los CGI's son fácilmente referenciados, ya que el desarrollador puede referenciarlos en una línea en la página Web.

Debido a que los programas CGI se ejecutan en su propio shell del sistema operativo, estos programas no tienen conflictos de concurrencia con otras peticiones HTTP ejecutando el mismo CGI.

Desventajas

El tiempo de respuesta de un programa CGI es muy alto, debido a que éste se ejecuta en su propio shell de sistema operativo y la creación de un nuevo shell es una actividad pesada para el sistema operativo. La ejecución es muy ineficiente debido al proceso de carga del código en memoria que se realiza cada vez que un usuario requiere su ejecución por medio de la página Web que lo invoca. Este funcionamiento resulta obsoleto, y además, compromete el funcionamiento de las máquinas que soportan servidores Web muy utilizados.

Un CGI no es escalable. Si el número de personas accediendo la aplicación Web incrementa de 50 a 5000, por ejemplo, el CGI no puede ser adaptado para manejar la carga. Hay una limitante en el número de procesos separados de sistema operativo que una computadora puede correr.

Resulta difícil mantener las páginas Web que se devuelven al cliente, ya que las instrucciones HTML se encuentran insertadas en el propio código del programa CGI mezclándose sus funcionalidades.

Los lenguajes para CGI no son siempre robustos u orientados a objetos; además, los lenguajes de script son dependientes de la plataforma.

3.1.2 Practical Extracting and Reporting Language (PERL)

PERL es un lenguaje desarrollado por Larry Wall, que está optimizado para el tratamiento de archivos de texto, extracción de información e impresión de informes basados en dicha información.

Combina algunas características de C, sed, awk y sh. Su sintaxis es muy cercana a C. PERL es el lenguaje más utilizado en la realización de CGI's por su versatilidad y potencia. Los programas para procesar formularios en la Internet llevan la extensión .pl, están escritos en PERL. En general, los programas en PERL se ejecutan en el servidor, como todos los programas CGI.

Para la ejecución de estos CGI's es necesario tener instalado un intérprete de PERL en el sistema y hacer residir los programas en el directorio cgi-bin del servidor, o alternativamente, residir en otro directorio y tener permisos CGI (es decir, el servidor reconoce que son CGI's y los ejecuta), además de ser archivos con permisos de ejecución.

Conexión con Bases de Datos

Mediante una serie de módulos adicionales, tales como el ODBC, es posible realizar conexiones a bases de datos, desde BD gratuitas hasta propietarias o mediante la utilización de interfaces específicas como por ejemplo OraPERL. Sin embargo lo más habitual es la utilización de archivos secuenciales manejados directamente desde el lenguaje para la gestión de datos.

Ventajas

El uso de un sistema de gestión de bases de datos no es algo obligatorio ya que PERL realiza la manipulación de información utilizando archivos planos, lo cual reduce un coste económico.

Es posible construir programas que pueden ser usados como filtros para obtener información de archivos o realizar búsquedas.

Se puede utilizar en varios entornos, como puede ser Windows 95, OS/2, sin realizar cambios de código, siendo únicamente necesaria la introducción del interprete PERL correspondiente a cada sistema operativo.

Desventajas

Es más eficiente la utilización de archivos planos que la de sistemas de gestión de bases de datos en caso de que el volumen de datos manejados no sea muy alto.

3.1.3 Servlets

Los Servlets son un reemplazo efectivo de los CGI, se escriben en código Java y manejan en forma más óptima los recursos del servidor que los proporciona. Solucionan el problema de hacer la programación del lado del servidor con APIs específicos de la plataforma, ya que están desarrollados con el API Java servlet, que es una extensión estándar de Java.

La tecnología Servlet proporciona las mismas ventajas del lenguaje Java en cuanto a portabilidad y seguridad, ya que un servlet es una clase de Java igual que cualquier otra, y por tanto tiene en ese sentido todas las características del lenguaje.

Los servlets corren dentro de un contenedor de componentes llamado contenedor Web (nuevo término para el servlet engine), el cual, es una máquina virtual de Java (Java Virtual Machine, JVM) que suministra una API de servlets.

Los servlets pueden entender peticiones HTTP y por lo tanto, obtener fácilmente información acerca del cliente (la permitida por el protocolo), tal como su dirección IP, el puerto que se utiliza en la llamada, el método utilizado (*GET*, *POST*), etc. Permiten además la utilización de *cookies* y sesiones, de forma que se puede guardar información específica acerca de un usuario determinado, personalizando de esta forma la interacción cliente-servidor.

Conexión con Bases de Datos

Los servlets pueden actuar como enlace entre el cliente y una o varias bases de datos en arquitecturas cliente-servidor o en arquitecturas de tres capas (si la base de datos está en un servidor distinto).

La conexión y operaciones con bases de datos se hacen mediante las API's *JDBC/ODBC*, creando un *DSN* (Data Source Name) y mandándolo llamar ya sea desde el servlet o bien desde un *Java Bean* que haga el proceso de conexión con la base de datos.

Ventajas

Una de las principales ventajas de los servlets con respecto a los programas CGI, es la del rendimiento, ya que mientras es necesario cargar los programas CGI tantas veces como peticiones de servicio existan por parte de los clientes, los servlets, cuando son llamados por primera vez, quedan activos en la memoria del servidor hasta que el programa que controla el servidor los desactiva. De esta manera se minimiza en gran medida el tiempo de respuesta.

El contenedor de Web es en sí un proceso de sistema operativo, pero corre como un servicio y está disponible continuamente. Los servlets son escalables, muchas peticiones pueden ser ejecutadas debido a que el servlet se ejecuta como un *hilo* dentro del proceso del Contenedor Web en lugar de como un proceso separado del sistema operativo. Por lo tanto, cuando varios clientes envían peticiones, éstas son procesadas concurrentemente usando un hilo Java por cada petición dentro del contenedor Web.

Son independientes de la plataforma. Los servlets son independientes del servidor utilizado y de su sistema operativo.

El contenedor de Web provee servicios adicionales tales como el manejo de errores y seguridad.

Desventajas

La principal desventaja de los servlets es la escasez de separación de la lógica del negocio y la lógica de presentación, lo cual los hace difíciles de usar en la generación de respuestas Web.

Si los administradores o los diseñadores necesitan cambiar la apariencia de la página, ellos tienen que editar y recompilar el servlet aunque la lógica del negocio esté trabajando correctamente.

3.1.4 Hypertext Preprocessor (PHP)

PHP es un lenguaje interpretado de alto nivel que puede ser incluido en las páginas HTML y ejecutado por el servidor de Web.

PHP realiza cualquier tarea que puede realizar un CGI, como procesar la información de formularios, generar páginas con contenidos dinámicos, o mandar y recibir cookies. PHP nos permite embeber sus pequeños fragmentos de código dentro de la página HTML y realizar

determinadas acciones de una forma fácil y eficaz sin tener que generar programas programados íntegramente en un lenguaje distinto al HTML.

Existen tres modalidades básicas de desarrollo con PHP, las cuales se describen a continuación:

- ◆ **Scripts del lado del servidor:** Esta es la implementación más utilizada y para ello se requieren tres elementos, los cuales son, el parser PHP (CGI ó módulo), el servidor web y el navegador. El resultado del programa PHP se obtiene a través del navegador al realizar la petición de la página al servidor.
- ◆ **Scripts en línea de comandos:** Solamente se necesita el parser PHP para usarlo en scripts ejecutados regularmente desde *cron* (en Unix ó Linux) ó desde el Task Scheduler (en Windows). Estos scripts también pueden ser usados para tareas simples de procesamiento de texto.
- ◆ **Aplicaciones gráficas:** PHP no es probablemente el mejor lenguaje para escribir aplicaciones gráficas, pero si se conoce bien PHP, se podría utilizar PHP-GTK para escribir dichos programas. Es también posible escribir aplicaciones independientes de una plataforma. PHP-GTK es una extensión de PHP, no disponible en la distribución principal.

Conexión con Bases de Datos

La conexión y operaciones con bases de datos se hacen mediante ODBC, creando un DSN y mandándolo llamar desde la página PHP, o bien por medio de comunicación directa para algunos manejadores de base de datos.

Ventajas

PHP, aunque multiplataforma, ha sido concebido inicialmente para entornos UNIX y es en este sistema operativo donde se pueden aprovechar mejor sus características.

PHP presenta una filosofía totalmente diferente y, con un espíritu más generoso, es progresivamente construido por colaboradores desinteresados que implementan nuevas funciones en nuevas versiones del lenguaje.

Desventajas

Las características de autenticación HTTP en PHP solo se encuentran disponibles cuando se está ejecutando como un módulo en Apache y hasta ahora, estas características no están en la versión CGI.

La lógica del negocio y la de presentación se encuentran mezcladas en los programas, de tal manera que el mantenimiento de los códigos no es sencillo.

3.1.5 Active Server Pages (ASP)

Es un lenguaje de *scripting* desarrollado por Microsoft para crear y ejecutar aplicaciones dinámicas e interactivas en el Web. Las páginas ASP se interpretan del lado del servidor. Cuando un usuario envía una petición al servidor por medio del browser, el servidor llama al ASP que lee la petición y entonces la página HTML de respuesta se genera dinámicamente por sentencias HTML y procesos de script definidos en la página ASP. Finalmente, la página HTML de respuesta es enviada al navegador del cliente.

Las páginas ASP pueden hacer uso de componentes *COM*, que son objetos escritos en otro lenguaje, de manera que si se tiene algo ya programado, las páginas ASP's a través del Internet Information Server (*IIS*) pueden hacer uso de los métodos de estos objetos. El esquema de trabajo que se recomienda para implementar soluciones Web con páginas ASP, es mediante el desarrollo de objetos COM que contengan la lógica del negocio, y la parte html de las páginas ASP forme parte de la interfaz del usuario.

Conexión con Bases de Datos

Para realizar conexión con base de datos se recomienda la tecnología *ADO*, la cual es una suite de objetos que permiten realizar la conexión y operaciones con distintos manejadores de base de datos.

Ventajas

Esta tecnología permite separar la lógica del negocio de la presentación al usuario, lo cual proporciona beneficios que se manifiestan en el mejor desempeño de las aplicaciones. La lógica del negocio residirá en los componentes que se encuentren en el servidor, mientras que la lógica de presentación residirá en el cliente. Esta estructura permite además, ocultar el código importante.

Cuando se envía la respuesta al cliente, éste no sabe si lo que recibe proviene de un archivo estático o ha sido generado a través de un lenguaje de programación, ya que el resultado se envía en HTML standard interpretable por cualquier navegador, lo cual beneficia al programador ya que el trabajo que desarrolla es válido para cualquier navegador.

Desventajas

La desventaja de las ASP, radica principalmente en que éstas son soportadas por tecnología Microsoft, lo cual, es una limitante para las empresas y desarrolladores que no cuentan con ella. Para la implantación de un servidor Web que soporte ASP, se necesita de IIS 5.0 o *PWS*.

3.1.6 Java Server Pages (JSP)

Es una tecnología de *scripting* basada en el lenguaje Java que permite el desarrollo de sitios Web dinámicos. JSP fue desarrollada por Sun Microsystems para interpretarse del lado del servidor. Los JSP's son archivos HTML con etiquetas especiales conteniendo código Java, el cual provee el contenido dinámico; además los JSP's pueden comunicarse con clases Java, objetos RMI, CORBA, etc.

Las páginas JSP son opuestas a los servlets, en lugar de código Java que contiene HTML, las páginas JSP son etiquetas HTML que contienen código Java embebido.

Las páginas JSP son convertidas por el contenedor Web en un servlet, el cual es compilado en una clase para posteriormente crear la instancia del servlet. La creación y compilación automática del servlet ocurre la primera vez que se acceda a la página y es este servlet el que procesa cada petición que es enviada al JSP. Esta característica de las páginas JSP es una ventaja sobre otras tecnologías debido a que las páginas JSP son compiladas en byte code de tecnología Java, mientras que las páginas ASP o PHP son interpretadas en cada petición HTTP.

Dependiendo del comportamiento del servidor Web, el servlet será activo en memoria durante algún periodo para utilizarlo una y otra vez sin necesidad de recrearlo y recompilarlo.

El esquema de trabajo que se recomienda para implementar soluciones Web con páginas JSP, es mediante el desarrollo de componentes Java Beans que contengan la lógica del negocio, y la parte HTML de las páginas JSP forme parte de la interfaz del usuario.

Conexión con Bases de Datos

La conexión y operaciones con base de datos se hacen mediante las API's JDBC/ODBC, creando un DSN y mandándolo llamar ya sea desde el JSP o bien desde un Java Bean que haga el proceso de conexión con la base de datos.

Ventajas

Las aplicaciones Web que utilizan páginas JSP tienen alto desempeño y escalabilidad debido a que se utilizan hilos de proceso en vez de procesos completos del sistema operativo.

La tecnología JSP está construida sobre tecnología Java, por lo tanto, es independiente de la plataforma. Además, los elementos de script pueden ser escritos en lenguaje Java así que los JSP's pueden tomar ventaja del lenguaje orientado a objetos y de todas sus API's.

Desventajas

La principal desventaja de esta tecnología es que la lógica de la aplicación está hecha dentro de las páginas JSP utilizando elementos scripting. Al poner la lógica del negocio en las páginas JSP, se crea un problema opuesto al que los servlets sufren, la separación entre la lógica del negocio de la lógica de la presentación es muy pobre. Sin embargo es posible utilizar los servlets y los JSP juntos para la separación de dicha lógica, para lograrlo, los servlets actúan como un controlador, el cual, verifica la forma de los datos, los actualiza y determina la siguiente vista como la respuesta. Los JSP's actúan como la vista, la cual representa la respuesta HTML y provee los formularios HTML para permitir la interacción con el usuario.

3.1.7 JavaScript

Es un lenguaje utilizado para aumentar el dinamismo a las páginas desarrolladas con HTML, permitiendo mayor interactividad entre las mismas. Este código se ejecuta en el cliente y ayuda a

disminuir la carga por la realización de muchas funciones relacionadas con la gestión de interfaces de usuario, aumentando el rendimiento global del sistema cliente-servidor.

Java script no es un lenguaje de propósito general, sino que, se incrusta entre el código de las páginas HTML. Es interpretado por un intérprete incorporado en el navegador que posibilita la ejecución de código generado dinámicamente. Establece una jerarquía de objetos encabezada por el propio navegador, permitiendo la toma de control y el acceso a todos los elementos contenidos en él. Permite la captura y tratamiento de una serie de eventos provocados tanto por el usuario como por el navegador. Sirve de base para incorporar otros elementos tecnológicos como son ActiveX, XML, controles multimedia o HTML dinámico.

3.1.8 Applets

Son aplicaciones escritas en Java que permiten agregar funcionalidad a las páginas Web para ser ejecutadas por el browser y sobre el sistema operativo de la máquina cliente. Estas aplicaciones residen en el servidor de Web y se invocan desde las páginas Web para ser descargadas y ejecutadas en la máquina cliente.

Los applets ya que se ejecutan del lado de la máquina que invoca la página Web, están limitados a realizar operaciones que afecten el sistema operativo, así como lecturas y escrituras a disco y comunicaciones con otras máquinas diferentes al servidor Web de donde se invoca el applet. Para el manejo de estas operaciones es necesario manejar esquemas de seguridad basados en certificados de seguridad, los cuales implementan llaves públicas y privadas del lado del cliente y del servidor.

3.2 Tecnologías de Componentes para Aplicaciones Web

3.2.1 Enterprise Java Beans (EJB)

Conforman la tecnología de componentes en la plataforma de Java, la cual permite desarrollar aplicaciones basadas en objetos distribuidos transaccionales, los cuales corren del lado del servidor, son componentes que representan las reglas del negocio del cliente.

Estos componentes de software se ejecutan en un entorno especial llamado contenedor EJB, el cual controla cada acceso del Bean en tiempo de ejecución, incluyendo accesos remotos al Bean, seguridad, transacciones, concurrencia y accesos a un almacén de recursos.

EJB es la adaptación de un modelo de componente Java para servidores de bases de datos, transacciones y aplicaciones. Cada uno de estos funciona como un contenedor para componentes EJB, los cuales se intenta que sean portátiles entre servidores y sistemas operativos. Los desarrolladores escriben componentes que requieran de servicios del contenedor EJB, y el servidor secundario descifrá qué servicio puede proporcionar y cual deberá ser enviado a otro servidor.

Ventajas

La arquitectura de los Java Beans proporciona la capacidad de reutilizar componentes desarrollados, de tal manera que estos pueden ser ejecutados en cada sistema operativo y dentro de cualquier ambiente de aplicación.

Los EJB son componentes con arquitectura de plataforma neutral para el ambiente de aplicaciones Java. Representan una opción ideal para el desarrollo o unión de aplicaciones heterogéneas en hardware y sistemas operativos.

Los EJB pueden trabajar con plataformas propietarias como los *ActiveX*, para esto, la arquitectura de los EJB se conecta vía *bridges* con el modelo de componentes *ActiveX*.

Desventajas

La principal desventaja de los EJB es que requieren de muchos recursos del sistema, lo cual los convierte en una opción no viable para sistemas con recursos de hardware limitados.

3.2.2 COM, DCOM, COM+

Son un estándar que proveen la tecnología de componentes desarrollados en tecnologías de plataforma Microsoft, para ser ejecutados ya sea del lado del servidor mediante la administración de un contenedor de componentes, o del lado del cliente para ser invocados a través de una página Web y ejecutados por el browser de la máquina cliente.

Estos modelos de componentes se crearon con la finalidad de reutilizar código y agregar funcionalidades que son más fáciles de desarrollar en otro lenguaje de programación que en el que se está escribiendo la aplicación, es decir, que cualquier lenguaje de programación que soporte esta tecnología puede reutilizar componentes que estén o no escritos en el mismo lenguaje que el de la aplicación que los invoca.

Los componentes COM no tienen restricciones de acceso a los recursos del sistema, y cuando se instalan, se identifican por medio de un ID y una versión que se almacena en el Sistema, lo que permite identificar si el componente ya está instalado o no, y actualizarlo cuando sea necesario.

La diferencia entre estas tres tecnologías reside en la distribución y comunicación entre los mismos componentes.

3.3 Navegadores de Internet

Un navegador es un programa que permite acceder a los sitios publicados en Internet. El primer navegador desarrollado fue NCSA Mosaic en los 90's por el Centro Nacional de Aplicaciones de Super cómputo. Los navegadores actuales tomaron las bases de funcionalidad de NCSA Mosaic y además se caracterizan por ser *freeware* y algunos *shareware*.

Existen en la actualidad diferentes navegadores. Los más populares son Internet Explorer y Netscape, y seguidos en menor grado Opera y Lynx. Los navegadores se actualizan en la medida que HTML se actualiza y esto se ha logrado gracias a la superación de las limitantes en cuanto al ancho de banda y la velocidad de procesamiento de las computadoras, pero no todos los navegadores soportan los mismos estándares de HTML, así como efectos multimedia. Soportan servicios de Internet como http, ftp, telnet, gopher, WAIS, correo electrónico, noticias.

Las funciones principales de un navegador son:

- ◆ Conexión vía Internet a un servidor Web
- ◆ Presentar documentos en formato HTML así como gráficos, archivos de texto, etc.
- ◆ Interactuar con el sistema operativo para imprimir, salvar archivos, etc.
- ◆ Permitir la interacción con componentes (ActiveX, JavaScript, Java Applets, Plug-ins)

3.3.1 Netscape

Características Generales

Navegador que provee además de las características de un navegador, una herramienta para manejo de correo electrónico, además de una herramienta para diseño de páginas html.

Sistemas Operativos Soportados

Es un navegador que se encuentra disponible para las siguientes plataformas: OS 8.5-9.x, OSx, Win95/98/ME, NT/2000/XP, variaciones de Unix.

3.3.2 Mozilla

Características Generales

Navegador gratuito que distribuye su código fuente para su libre manipulación, fue el código inicial para Netscape Navigator ahora Netscape Communicator.

Sistemas Operativos Soportados

Win 9x, Win NT, Win 2000, Win Me, Win XP, variaciones UNIX.

3.3.3 Internet Explorer

Características Generales

Navegador desarrollado por Microsoft cuyas versiones se distribuyen con las diferentes versiones de sistemas operativos Microsoft, y se pueden obtener las actualizaciones vía Internet. Internet Explorer viene integrado en el sistema operativo windows.

Sistemas Operativos Soportados

Win 9x, Win NT, Win 2000, Win Me, Win XP, MacOS.

3.3.4 Opera

Características Generales

Opera, a diferencia de Netscape o Internet Explorer, no está basado en el código del NCSA Mosaic, sino que fue escrito desde cero.

Opera es un navegador pequeño pero con buenas características como: soporte para frames, tablas, códigos JavaScript, CSS o Java. Además, cuenta con características muy especiales como el zoom de una página o navegación simultánea de varias páginas en un solo ambiente de navegación.

Una de las cosas más importantes de Opera, es que es rápido, no se necesita de una máquina muy poderosa para navegar por la Web ya que, por ejemplo, instalado en un equipo Pentium I con 64 Mb RAM corre a muy buena velocidad. Opera tiene costo, pero versiones beta se pueden obtener en Internet. Se utiliza principalmente para tecnologías de telefonía celular.

Sistemas Operativos Soportados

Win 9x, Win NT, Win 2000, Win Me, Win XP, variaciones de UNIX, MacOS.

3.3.5 Konqueror

Características Generales

Konqueror utiliza un motor de rendimiento para HTML llamado KHTML. Las principales características de este motor son: habilidad para soportar applets de Java, JavaScript, CSS, componentes DOM1, DOM2 y DOM3 y por supuesto amplio soporte de HTML.

Sistemas Operativos Soportados

Soporta cualquier plataforma que corra en ambiente de escritorio KDE como UNIX en cualquiera de sus versiones. Linux y recientemente en Windows con una versión muy limitada.

3.3.6 Lynx

Características Generales

Navegador que permite visualizar la información únicamente en modo texto, lo cual lo hace muy rápido y ligero. Debido a que no provee la posibilidad de visualizar gráficos es de uso limitado.

Sistemas Operativos Soportados

Unix, VMS, Windows 95/98/NT, DOS, OS/2.

3.3.7 Estándares de Web y HTTP

El 3WC establece estándares de Web que deben ser soportados por los navegadores para el correcto funcionamiento de las aplicaciones. Los navegadores que soportan estos estándares son: Netscape v7, Mozilla FireFox, Opera 7, Konqueror e Internet Explorer 6.0 o mayor.

Los estándares de Web que deben ser soportados son: HTML 4.0, XML 1.0, CSS, XHTML 1.0 y 1.1 y DOM.

Con respecto a los estándares del protocolo HTTP que soportan los navegadores son HTTP 1.0 y HTTP 1.1.

3.4 Servidor Web y Servidor de Aplicación

El servidor de páginas Web es la parte primordial de cualquier sitio de Internet, ya que es el encargado de generar y enviar la información a los usuarios finales. En un sentido muy estricto, un servidor Web no es lo mismo que servidor de aplicación, pero últimamente estos dos términos se prestan a una gran confusión.

Cuando se crearon los primeros servidores de páginas como Apache, éste solo era encargado de enviar los datos al usuario final, pero cualquier otra información que requiriera de algún tipo de personalización era realizada por un intérprete que ejecutaba un script, generalmente PERL. Sin embargo, conforme las demandas de los servidores de páginas incrementaron, fue necesario hacer más eficiente este proceso, ya que el llamar a un intérprete para que ejecutara otro programa, disminuía el rendimiento del servidor de páginas.

Hoy en día, se pudiera decir que todos los servidores de páginas son servidores de aplicación ya que se les ha implementado funcionalidad especial que les permite realizar aplicaciones de servidor. Y para agregar confusión en la Industria, hoy día ya abundan los servidores de aplicación de java (Java Application Servers) que ya son designados servidores de aplicación.

Dependiendo de la funcionalidad, se atribuye complejidad al sistema, ya sea en requerimientos del sistema (memoria, procesadores), carga administrativa (configuración, tiempo de desarrollo) o alguna otra.

A continuación se presentan los servidores Web, servidores de aplicación y servidores de aplicaciones Java más utilizados en el mercado:

3.4.1 Apache

Características Generales

Apache es uno de los servidores de páginas más utilizados, ya que ofrece instalaciones sencillas para sitios pequeños. Provee el servicio de ejecutar aplicaciones interpretadas en los siguientes lenguajes: PERL, TCL, PHP, PYTHON, etc.

Otra característica importante es que puede conectarse directamente a una base de datos. Por cada ejecución de un programa o conexión a la base de datos, se levanta un proceso diferente, lo que implica demanda de procesamiento y memoria. Es un producto *open source*.

Sistemas Operativos Soportados

Solaris, Linux, HP-UX, Windows NT/2000, BeOS, MacOSX.

3.4.2 AOL Server

Características Generales

Es un producto open source, que fue diseñado tomando en cuenta las deficiencias que presentaba Apache en sus modelos iniciales. Implementa el manejo de hilos para compartir memoria del proceso general en varios subprocesos o hilos, esto no solo hace más eficiente las conexiones al servidor de páginas sino también reduce la carga sobre el mismo.

Integra el intérprete en su estructura interna, evitando generar un proceso nuevo por aplicación del servidor y mantiene grupos de conexiones latentes pools hacia bases de datos también para evitar generar procesos nuevos.

Intérprete de páginas Aol Dynamic Pages (ADP) que son muy similares a las ASP's de Microsoft o JSP's de Sun, la diferencia estriba que ADP's utilizan el lenguaje TCL y un API especialmente diseñado para acceder los elementos del servidor, pero su funcionamiento es igual al de ASP y JSP, que es mezclar elementos de HTML con elementos de programación para generar contenido dinámico.

Sistemas Operativos Soportados

Solaris, Linux, HP-UX, Windows 2000

3.4.3 Internet Information Server (IIS)

Características Generales

IIS es el servidor de páginas desarrollado por Microsoft para Windows NT/2000. Es el intérprete de páginas ASP, sencillo de instalar y permite realizar conexiones con diferentes manejadores de base de datos.

Sistemas Operativos Soportados

Windows NT/2000.

3.4.4 Servlet Engine

Un servlet engine es capaz de responder a requisiciones de Internet, actuando como servidor de páginas, sin embargo, aunque esto sea posible, la gran mayoría de las implementaciones de servlet engines no funcionan tan eficiente como un servidor de páginas, es por esto que se opta por utilizar un servidor de páginas (Apache, Aol, Netscape) en conjunción con un servlet engine. Un servidor de páginas con un servlet engine, suele utilizarse cuando se requiere utilizar encriptación o se tiene un sitio que contiene documentos puros en HTML, las cuales son dos áreas que un servidor de páginas supera en desempeño comparado con un servlet engine. La comunicación entre el servidor de páginas y el servlet engine se lleva a cabo mediante el protocolo denominado ajp12 y recientemente con su sucesor ajp13.

El servlet engine ofrece un ambiente donde habitan los JSP y Servlets, es ahí donde se contemplan una gran cantidad de funcionalidades como: manejo de hilos, de sesiones, conectividad con el servidor de páginas, es por esto que al servlet engine también se le denomina contenedor Web (Web-Container).

También se requiere de un Java Development Kit (*JDK*), el cual llevará acabo la ejecución de los programas Servlets y JSP's en Java; como toda otra implementación existen diversas versiones de JDK's, esto se debe a que cada JDK debe ser diseñado alrededor de un sistema operativo, algunos JDK's son: Java 2 Standard Edition (J2EE) de Sun y JDK de IBM.

Tres servlet engines que están en amplio uso y son utilizados con servidores de páginas son: Tomcat, ServletExec y RESIN, los cuales se describen a continuación.

TOMCAT

Tomcat surgió de Sun Microsystems cuando desarrollaban un servidor de páginas que utilizara Java, y posteriormente cedieron el código fuente a la fundación Apache. A pesar del nombre Apache-Tomcat; Tomcat no requiere de Apache para su funcionamiento, solo requiere de un JDK. Es un producto open source.

SERVLETEXEC

Servlet engine que permite distribuir Servlets y JSP's sobre IIS, Netscape Enterprise Server (NES), iPlanet Web Server (IWS) y Apache. No es un producto open source.

RESIN

Es un producto open-source. Permite utilizar Java Script como lenguaje de Script dentro de JSP's y acceso a XSL.

3.4.5 Java Application Servers

Son servidores que integran y administran los componentes con las reglas de negocio del sistema de información, desarrollados para soportar las especificaciones de J2EE. A este servidor de aplicación se le conoce como *middleware*, ya que es un servicio dedicado a servir componentes, y no se interpone entre la capa de presentación y datos en un sistema de información de n-capas.

Dependiendo de la empresa que desarrolle el servidor de aplicación éste puede contener inclusive hasta un servidor de páginas o algún otro desarrollo propietario, sin embargo, los dos elementos primordiales son el servlet engine y enterprise bean engine (bean-container).

El bean container ofrece un ambiente donde residen EJB's, en el bean container se contemplan varias funcionalidades: pooling hacia bases de datos (JDBC), control de transacciones (JTA-JTS), aplicaciones legacy (CORBA), entre otras cosas.

La mayor ventaja de este tipo de arquitectura se debe a la separación de funcionalidades y uso de protocolos de red como RMI/CORBA, esto facilita que puedan existir 4 o 5 Hosts en diferentes regiones geográficas, cada uno empleando cualquiera de los componentes antes mencionados.

Los servidores de aplicación que son denominados "Fully J2EE Compliant", es decir, que cumplen con todas las especificaciones J2EE indicadas por Sun son los siguientes:

- ◆ WebLogic
- ◆ Websphere
- ◆ JRun
- ◆ Oracle 9i Application Server
- ◆ iPlanet (Previamente Netscape Enterprise o Kiva)
- ◆ JBoss

3.5 Manejadores de Bases de Datos

Las diferencias entre los diversos manejadores de bases de datos relacionales en el mercado además de los factores comerciales como el costo de la adquisición y mantenimiento, se pueden establecer a partir de varios factores tecnológicos como:

Disponibilidad: La posibilidad de proporcionar los mecanismos que permiten recuperar la bases de datos contra fallos lógicos o físicos que destruyan los datos parcial o totalmente, manejo de esquemas de replicación y manejo de arquitecturas para tolerancia a fallos, como cluster y bases de datos en espejo.

Escalabilidad: La posibilidad de particionar una base de datos ya sea a nivel general de la base de datos, por tabla o por índice.

Seguridad: Se basa en tres conceptos principales, confidencialidad, accesibilidad e integridad. La confidencialidad se refiere a no develar datos a usuarios no autorizados, es decir, privacidad de la información que se puede reforzar mediante la encriptación de la misma; La

accesibilidad se refiere a la disponibilidad de la información y la integridad se refiere a que los datos sean consistentes y no reciban cambios no deseados.

Administración: La posibilidad de contar con herramientas que permitan realizar las labores administrativas del administrador de la base de datos de una manera más eficiente y en el menor tiempo, dichas herramientas pueden ser monitores de consultas de SQL, monitores de desempeño de la base de datos y procesos automáticos de depuración, respaldo y replicación.

Interoperabilidad: La posibilidad de extraer información almacenada en una base de datos origen y moverla a una diferente base de datos destino, ya sea entre bases de datos homogéneas o heterogéneas.

Ambiente de desarrollo: Se refiere a los diferentes lenguajes de programación con los que puede integrarse el manejador de base de datos con los cuales se pueden hacer diversas funcionalidades administrativas o de aplicación dentro del mismo.

Data warehousing : La posibilidad de que el manejador de base de datos cuente con el motor para realizar el análisis de información histórica almacenada en una base de datos del mismo para el soporte del proceso de toma de decisiones del negocio en el que se aplica.

Con base en estos criterios se presentan algunas características de los manejadores de bases de datos más reconocidos en el mercado.

3.5.1 Oracle

Características Generales

La solución de desarrollo de aplicaciones escalables de Oracle incluye dos series de productos: Oracle Designer/2000 y Oracle Developer/2000™. Designer/2000 soporta el diseño de complejos sistemas con reingeniería de procesos de negocios (Business Process Reengineering, BPR), análisis, y diagramadores de diseño. Developer/2000 brinda a las organizaciones la capacidad de construir rápida y productivamente sistemas sofisticados que escalan de grupo de trabajo a empresa. Con un repositorio común, soporte de metodología y diseño flexible, un entorno de desarrollo de servidor y cliente unificado, y una arquitectura portable abierta, Designer/2000 y Developer/2000 constituyen las únicas herramientas de desarrollo y diseño de cliente/servidor de segunda generación de la industria

Oracle ha integrado a Java en sus tecnologías respectivas; así los desarrolladores pueden crear con facilidad aplicaciones de cómputo portátiles en red. En la actualidad, Oracle tiene miles de desarrolladores de aplicaciones en Java, que ahora tienen la opción de crear aplicaciones en ambientes de múltiples niveles, y cuyas aplicaciones actuales correrán aceleradamente en esta poderosa plataforma integrada.

Factores Tecnológicos

Plataformas soportadas: Linux, Unix y Windows.

Disponibilidad: Provee diferentes herramientas de monitoreo, automatización y control, provee mecanismos para prevenir y corregir discos corruptos, provee el servicio de Real

Application Clusters como servicio de *cluster*, provee la aplicación Recovery Manager para la generación y restauración de respaldos.

Escalabilidad: Cualquier cambio a un atributo físico de la tabla se puede hacer en línea así como los cambios a algunos atributos lógicos, la tabla se puede particionar o cambiar de lugar.

Seguridad: Permite el manejo de seguridad a partir de certificados de seguridad (Llaves públicas y llaves privadas), encriptación de datos, manejo de políticas de seguridad para usuarios, maneja los permisos a nivel de un registro o registros de una tabla de la base de datos, maneja la administración centralizada a través del protocolo *LDAP*.

Administración: Permite realizar procesos de manera autónoma, por ejemplo deshacer de transacciones, manejo de memoria dinámica, permite definir espacios de trabajo temporales en donde se pueden hacer pruebas sobre funcionalidades sin que afecte al desempeño de la base de datos.

Ambiente de Desarrollo: Enterprise Java Engine, PL/SQL, JAVAscript, VBscript, and Apple Script.

Data warehousing: Oracle provee los servicios Online Analytical Processing (*OLAP*), Data Mining y Extraction Transformation and Loading (*ETL*).

Versión más reciente: Oracle 10g.

3.5.2 Sybase Server

Características Generales

Sybase Adaptive Server proporciona un alto rendimiento y escalabilidad para aplicaciones *OLTP*, *Datawarehouse* e Internet. Para responder hoy en día a los ambientes de negocio competitivos, muchas organizaciones están mejorando su eficacia adoptando estructuras descentralizadas. Como condición de cambio, los negocios son reorganizados frecuentemente. Sus sistemas de información deben soportar los requisitos de la organización. Sybase Replication Server (la primera arquitectura práctica para construir un sistema distribuido de alto rendimiento, robusto y eficiente) provee un acercamiento flexible a la entrega de la información que se ajusta a las necesidades cambiantes del negocio.

Factores Tecnológicos

Plataformas soportadas: Compaq Tru64, HP/UX (32 & 64 bit), IBM AIX (32 & 64 bit), Linux, Microsoft Windows NT, Microsoft Windows 98 (Cliente), SGI IRIX (32 & 64 bit), Sun Solaris (32 & 64 bit).

Disponibilidad: Soporta servicios de cluster como: Compaq - TruCluster, IBM - HACMP, Sun - Sun Cluster, HP - ServiceGuard, Windows NT - Microsoft Cluster Software (MSCS), Veritas Cluster Software, SGI FailSafe.

Escalabilidad: Permite particionar la base de datos o bien las tablas de la misma en diferentes páginas dentro de los dispositivos definidos, utilizando el método de round robin para la distribución de la información.

Seguridad: Maneja la seguridad de usuarios centralizada, se acopla a diferentes ambientes de autenticación, la seguridad en tablas se maneja a nivel de registro y permite que la información se pueda encriptar. Permite manejar seguridad a nivel de tareas administrativas, como manejo de passwords en los archivos de respaldo, manejo de permisos a nivel de comandos administrativos y sobre los catálogos del sistema de la base de datos.

Administración: La aplicación Job Scheduler permite ejecutar diversas tareas administrativas programadas, como respaldos, verificar la consistencia de la Base de Datos, administración de índices y generación de reportes.

Interoperabilidad: Posee aplicaciones como Replication Server para replicar información; OmniSQL Gateway y Turnkey Gateway para conexión a orígenes de datos heterogéneos.

Ambiente de Desarrollo: C, XML, Java.

Data warehousing: Cuenta con el motor de análisis Sybase IQ 12.5.

Versión más reciente: Adaptive Server Enterprise 12.5.

3.5.3 Informix

Características Generales

Se creó en 1980, se considera como un gran exponente en el área de bases de datos relacionales, la cual adopta la arquitectura OLTP, los lenguajes de cuarta generación y la tecnología de bases de datos relacionales con orientación a objetos, sobre plataformas UNIX y Windows.

Las áreas de aplicación en las que se destaca Informix son:

- ◆ Datawarehouse.
- ◆ Manejo dinámico de contenidos Web.
- ◆ Servidores de bases de datos.
- ◆ Herramientas para el desarrollo de las aplicaciones en áreas de procesamiento de transacciones en línea.
- ◆ Sistemas empresariales de computación distribuida.

Factores Tecnológicos

Plataformas soportadas: AIX, HP-UX, IRIX, Linux, Solaris, Windows 2000, Windows 2003 y Windows XP.

Disponibilidad: Soporta diferentes soluciones de cluster, posee la herramienta XBSA para una fácil integración con herramientas de respaldo de terceros, permite realizar respaldos y restauración de respaldos a nivel de tabla.

Escalabilidad: Permite particionar la información en diferentes discos distribuyendo la información mediante el algoritmo de round robin.

Seguridad: Maneja los permisos a nivel de objeto y de comandos.

Administración: Posee una gama de utilerfas para hacer mas fácil la administración de la Base de Datos, como un administrador del espacio asignado para la Base de Datos, ServerStudio JE que es un ambiente de desarrollo que permite realizar programas para automatizar tareas administrativas y el Administrador de Informix, que permite realizar tareas de respaldo y restauración de respaldos, manejo de la configuración del servidor y monitoreo del mismo por medio de un navegador.

Interoperabilidad: Posee los servicios de replicación High Availability Data Replication (HDR) para casos de replicación ante fallas destructivas y Enterprise Replication (ER) para replicación en diferentes áreas geográficas. Se integra cien por ciento a la arquitectura de procesamiento en paralelo que maneja IBM, así como las diversas soluciones que publica.

Ambiente de Desarrollo: IBM Informix 4GL, ODBC, JDBC, OLE/DB, SQLJ, ServerStudio JE.

Data warehousing: Informix Dynamic Server no posee una herramienta de análisis de datos integrada en el manejador pero se puede operar con el servidor Redbrick Warehouse de IBM para este fin.

Versión mas reciente: IBM Informix Dynamic Server 9.40.UC40.

3.5.4 SQL Server

Características Generales

SQL Server es un gestor de base de datos relacionales que utiliza la arquitectura cliente servidor. Es un entorno de trabajo sobre servidores Windows, siendo sus principales armas la rapidez, la integración con el sistema operativo y una gran facilidad de administración del sistema.

Una de las principales cualidades de SQL Server es aprovechar al máximo las posibilidades del sistema operativo, característica que hace que los rendimientos que obtiene SQL Server en Windows sean muy superiores a sus competidores.

- ◆ Multithreading.
- ◆ Multiproceso simétrico (SMP).
- ◆ Independencia del soporte de red (TCP/IP, IPX/SPX, etc.).
- ◆ Unificación de validación de accesos (Usuarios SQL ó Windows).
- ◆ Integración de control.

Se adapta a las necesidades de la empresa, soportando desde unos pocos usuarios a varios miles. Diseñada desde su inicio para trabajar en entornos Internet, capaz de integrar los nuevos desarrollos para estos entornos con los desarrollos heredados de aplicaciones tradicionales.

Factores Tecnológicos

Plataformas soportadas: Windows 2000, Windows NT.

Disponibilidad: Posee herramientas de respaldo, restauración de respaldos y replicación, soporta el servicio de bases de datos en espejo y de cluster.

Escalabilidad: Permite crear particiones a nivel de tabla o índices para poder distribuir, clasificar la información y hacer búsquedas más eficientes, soporta aplicaciones a 64 bits.

Seguridad: Permite encriptación y manejo de diferentes esquemas de seguridad de la información, políticas de definición de passwords, permisos a nivel de objeto, permite asociar un usuario del sistema operativo para autenticarse en la base de datos.

Administración: Posee herramientas de administración, programación de actividades de operación y monitoreo que le permiten al administrador de la base de datos hacer las actividades del día a día de manera más eficiente, así como verificar el desempeño de la base de datos.

Interoperabilidad.

Se puede integrar y conectar con diferentes aplicaciones y Bases de Datos. Se integra completamente a productos de Microsoft como Servidores Windows, Visual Studio y herramientas de BackOffice.

Ambiente de Desarrollo: Plataforma integrada con las tecnologías de Internet como ActiveX, ADC y Microsoft Transaction Server y con herramientas de gestión y desarrollo para Internet como FrontPage97, Microsoft Office97 y Visual Interdev. Además .NET y XML.

Data warehousing: Cuenta con el motor SQL Server 2005 Analysis Services.

Versión más reciente: SQL Server 2005.

3.6 Requerimientos de Hardware y Software

Dadas las necesidades del proyecto y aprovechando las ventajas en cuanto a mantenimiento del sistema que la arquitectura en tres capas nos provee, separando la lógica del negocio, el repositorio de información y la presentación al usuario, se requerirá diseñar e implementar un sistema vía Internet, en el que manejaremos componentes, los cuales validarán las reglas del negocio, un servidor de base de datos, que se utilizará como repositorio de información además de validar algunas reglas del negocio y la parte de la presentación al usuario, que serán páginas Web las cuales el usuario las accederá a través de un navegador de Internet.

Ante el escenario planteado y dados los recursos con los que cuenta la USECAD para la liberación de este servicio se tienen los siguientes requerimientos:

3.6.1 Requerimientos de Software

Software para Servidores

♦ *Servidor Web y de Aplicación*

Sistema Operativo: Linux.RedHat 7.0

Servicio de Aplicación: Tomcat 4.1.2

Compilador: Java 2 SDK, Standard Edition v. 1.3 o 1.4

- ◆ *Servidor de Base de Datos*

Sistema Operativo: UNIX Solaris System V.7

Manejador de Base de Datos: Sybase 11.0

Software para Clientes

Navegadores: IE Explorer 4.0 o mayor, ó Netscape 5 o mayor.

3.6.2 Requerimientos de Hardware

Hardware para Servidores

- ◆ *Servidor Web y de Aplicación*
- ◆ *Servidor de Base de Datos*

Hardware para Clientes

- ◆ *PC's*

CPU: Pentium o mayor

RAM: 64MB

Acceso a Internet

Ante la arquitectura planteada y dado que el hardware y software para la base de datos y el servidor web son proporcionados por USECAD, se eligió como arquitectura de desarrollo a Java ya que además de ser libre de licencias, esta soportado por Sun, por lo que provee soporte; es independiente de la plataforma en la que se desarrolle, y lo mas importante, que es un lenguaje cien por ciento orientado a objetos el cual nos ayudará en gran medida para la implantación de nuestro diseño que estará enfocado a una metodología orientada a objetos, que cada vez va tomando más fuerza en el mercado, lo que implica que el mantenimiento del sistema podrá ser soportado por mayor número de desarrolladores.

Dado a que Java es una plataforma que facilita una arquitectura en la cual los componentes de negocio son ubicados en una capa separada [4], nos provee de herramientas para desarrollo en Internet, por lo que se utilizará diferentes tecnologías Java para el desarrollo de este proyecto, como son JSP's, los cuales fungirán como capa de presentación al cliente y reglas de negocio respectivamente y para la presentación de reportes.

A continuación se muestra el diagrama de despliegue propuesto para la aplicación.

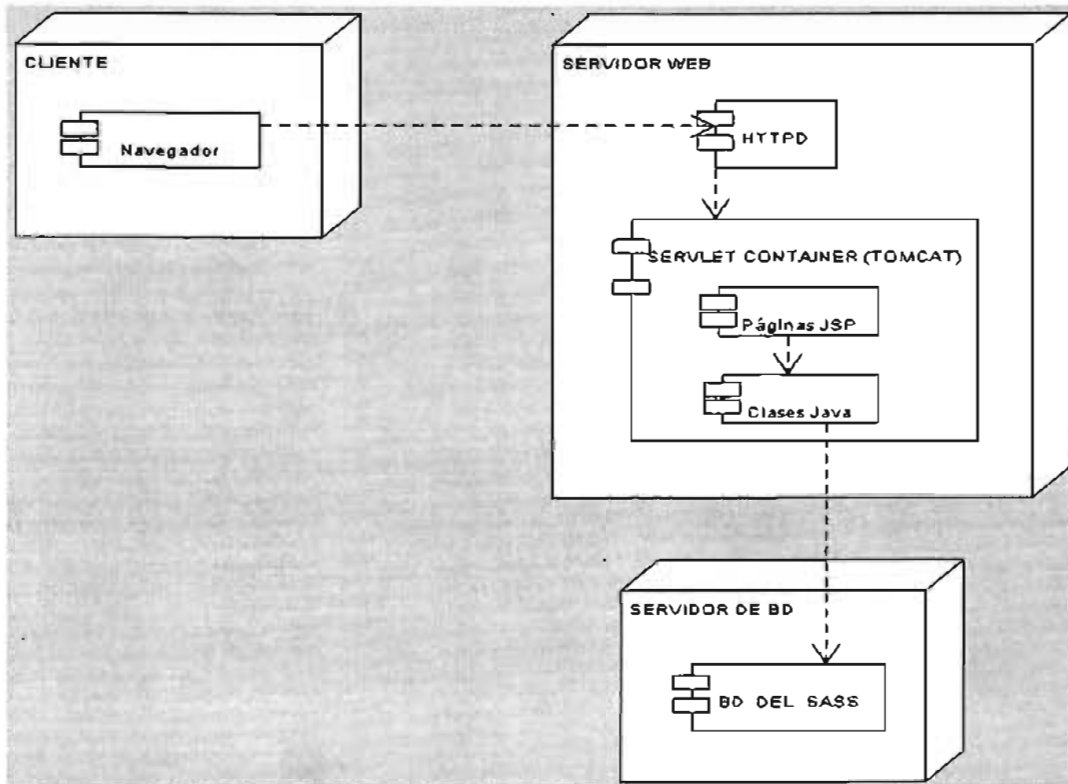


Figura 3.1 Propuesta para la arquitectura física del SASS.

Análisis y Diseño del Sistema

CAPÍTULO 4

ANÁLISIS Y DISEÑO DEL SISTEMA

4.1. Análisis del Sistema

4.1.1. Análisis de Requerimientos

Dados los resultados obtenidos durante el Estudio Preliminar (Capítulo 1), podemos definir que los requerimientos del usuario se traducen en la necesidad de desarrollar los siguientes módulos del sistema.

- ◆ *Módulo de Registro.* Comprende el registro del alumno en el programa de SS de su preferencia, para esto es necesario que llene el Formato de Solicitud de Autorización de Prestación de SS e Información Estadística y lo entregue a la CSSDI correspondiente.
 - ◆ *Módulo de Prealta.* La CSSDI verifica y valida la información proporcionada por el Alumno en el Formato de Solicitud de Autorización de Prestación de SS e Información Estadística y de acuerdo a los lineamientos de SS decide autorizar o rechazar dicha solicitud.
 - ◆ *Módulo de Alta.* Permite dar de alta a un alumno en un programa de SS con la información establecida en el Formato de Solicitud de Autorización de Prestación de SS e Información Estadística. Cabe mencionar que para que este proceso se lleve a cabo, es necesario que la documentación correspondiente para la alta de SS del alumno se le envíe a la CGSS.
 - ◆ *Módulo de Prebaja.* Aquí, la CSSDI puede realizar la prebaja de un alumno cuando éste no cumple con los lineamientos de SS estipulados.
 - ◆ *Módulo de Baja.* La CGSS autoriza la baja definitiva del alumno previamente solicitada por la CSSDI. No se omite precisar que para que este proceso se lleve a cabo es necesario que el formato de Notificación de Bajas de SS emitida por cada División de Ingeniería se le envíe a la CGSS.
 - ◆ *Módulo de Actualización de Datos.* Permite realizar modificaciones sobre los datos personales de los alumnos inscritos en algún programa de SS. Cabe mencionar que cada alumno es responsable de mantener la veracidad de la información registrada en el sistema.
 - ◆ *Módulo de Informes Bimestrales.* Posterior a la revisión de los informes entregados por los alumnos y de acuerdo a los lineamientos de SS se realiza la actualización de los estatus de éstos, así como la especificación de la fecha de entrega. Estas actividades son realizadas por el personal de la CSSDI.
 - ◆ *Módulo de Consulta de Avance.* Cada Alumno que ha registrado su SS tiene la posibilidad de consultar su nivel de avance en el ejercicio de su SS.
 - ◆ *Módulo de Mensajes.* Tanto los Coordinadores de SS de cada División de Ingeniería, como los Alumnos que ejercen su SS, consultan los mensajes que les hayan sido enviados como
-

consecuencia de cualquiera de los siguientes movimientos: rechazo de prealta, rechazo de alta, autorización de prebaja, rechazo de terminación o rechazo de liberación.

- ◆ *Módulo de Terminación.* La CSSDI, consulta, verifica y autoriza los registros de aquellos alumnos cuyo periodo de duración en meses se haya cumplido, esto de acuerdo a la información proporcionada en el Formato de Solicitud de Autorización de Prestación de SS e Información Estadística en el proceso registro. Cabe mencionar que para realizar la autorización de la terminación, es necesario que el alumno entregue la Carta de Terminación expedida por la Dependencia donde realizó su SS.
- ◆ *Módulo de Liberación.* Permite validar la terminación del SS del alumno, así como imprimir la carta de Liberación. Cabe mencionar que para que este proceso se lleve a cabo es necesario que tanto los informes bimestrales del alumno como la Carta de Terminación de SS se le envíen a la CGSS que es quien realiza esta actividad.
- ◆ *Módulo de Ingreso Comentarios.* Tanto los Coordinadores de cada División de Ingeniería como el Coordinador General de SS podrán ingresar al sistema comentarios adicionales sobre los alumnos, los cuales les permitirán registrar información extra que consideren necesaria para las labores administrativas que desempeñan.
- ◆ *Módulo de Estadísticas.* Este módulo es utilizado por la CGSS y la CSSDI y permite aprovechar la información almacenada en la base de datos para emitir estadísticas sobre los diferentes criterios mostrados a continuación:
 1. Reporte de Alumnos activos. (Columnas: Generación-Carrera)
 2. Reporte de Alumnos activos. (Columnas: Carrera-Nombre)
 3. Reporte de Alumnos activos. (Columnas: Situación-Número de cuenta)
 4. Reporte de Bajas. (Columnas: Generación-Carrera)
 5. Reporte de Bajas. (Columnas: Carrera-Nombre)
 6. Reporte de Alumnos que terminaron. (Columnas: Generación-Carrera)
 7. Reporte de Alumnos que terminaron. (Columnas: Carrera-Nombre)
 8. Reporte de Inicios en programas int-ext. (Columnas: Carrera-Nombre)
 9. Reporte de Inicios en programas int-ext. (Columnas: Carrera-Fecha)
 10. Reporte de Inicios en programas int-ext. (Columnas: División)
 11. Reporte de Terminaciones en programas int-ext. (Columnas: Carrera-Nombre)
 12. Reporte de Terminaciones en programas int-ext. (Columnas: Carrera-Fecha)
 13. Reporte de Terminaciones en programas int-ext. (Columnas: División)
 14. Reporte de Emisión de cartas de Terminación. (Columnas: Generación-Carrera)
 15. Informes Aceptados. (Columnas: Generación-Carrera)
 16. Observaciones. (Columnas: Cuenta-Observaciones)
 17. Alumnos sin Beca. (Columnas: Nombre-Teléfono-Email)
- ◆ *Módulo de Reportes.* En este módulo, la CGSS y la CSSDI generarán, con base en la información almacenada, reportes sobre los siguientes criterios:
 1. Alumnos Activos por Nivel de Avance en el SS (Posibles estatus: Todos, Registro, Preatla, Alta, Informes, Liberación, Terminación, Carta de Liberación)
 2. Bajas por Motivo. (Posibles estatus: Todos)

3. Inicios y/o Terminaciones en Programas Internos-Externos (Posibles estatus: Todos, Inicios, Terminaciones)

- ◆ *Módulo de Cambio de Contraseña.* Cada usuario del sistema realiza el cambio de su contraseña cuando así lo requiera.
- ◆ *Módulo de Reinicio de Contraseña.* La CGSS y la CSSDI pueden reiniciar la contraseña de un alumno que se encuentra en ejercicio de su SS.
- ◆ *Módulo de Mantenimiento.* La CGSS realiza las altas y bajas de las cuentas del personal de cada CSSDI.

4.1.2. Identificación de los Procesos del Sistema

Las diferentes reuniones sostenidas con los usuarios arrojan como resultado la identificación y conocimiento de los procesos, de los cuales, se compone el Proceso de Realización de SS. A continuación se describen los procesos principales.

4.1.2.1 Proceso de Alta del Alumno

Registro de un Alumno en el Programa de SS

1. El alumno consulta los programas de SS y elige el más apropiado, la forma en como se publican estos programas es en carteles que se pegan en las instalaciones de las diferentes Divisiones Ingeniería, además también es posible consultarlos en la página Web de SS de la Facultad de Ingeniería.
2. Dirigirse al lugar en donde se está promoviendo dicho programa y pedir una carta de aceptación para la prestación del SS
3. Dirigirse a la División de Ingeniería correspondiente y llenar la Solicitud de Autorización de Prestación del SS e Información Estadística, y entregar con ésta, la Carta de Aceptación e Historial Académico.

Aprobación de Prealta del Alumno por el Coordinador de SS de la División

4. Verifica la Carta de Aceptación de prestación de SS del Alumno, la Solicitud de Autorización de Prestación del SS e Información Estadística y que en el Historial Académico cumpla con el porcentaje de créditos requerido.
5. Se establecen las fechas de entrega de informes de avance.

Aprobación de Alta del Alumno por el Coordinador General de SS de la Facultad de Ingeniería

6. Verifica la Carta de Aceptación de Prestación de SS del Alumno, la Solicitud de Autorización de Prestación del SS e Información Estadística y que en el Historial Académico cumpla con el porcentaje de créditos requerido.
7. Da de alta al alumno en el Sistema de Control de SS que tienen actualmente.

4.1.2.2 Proceso de Entrega de Informes de Avance

8. Cada bimestre, el alumno elaborará un reporte de actividades, el cual deberá llevar el visto bueno de la dependencia donde presta el SS.
9. Entregar el reporte en la unidad de SS de la División correspondiente.
10. En la entrega del penúltimo reporte la unidad de SS se le entrega al alumno el Formato de Terminación de SS.

4.1.2.3 Proceso de Terminación de SS

11. Entrega con el último reporte la carta de terminación de SS.
12. Se le indica al alumno cuando puede pasar por su carta de Liberación.

Aprobación de Terminación del Alumno por el Coordinador de SS de la División

13. Verifica la carta de terminación de SS.
14. Prepara todos los documentos del alumno para enviárselos a la Coordinación General de SS de la Facultad de Ingeniería.

Aprobación de Liberación del Alumno por el Coordinador General de SS de la Facultad de Ingeniería

15. Verifica la carta de terminación de SS y los informes bimestrales entregados.
16. Imprime y entrega la carta de Liberación del Alumno.

4.1.3. Diagrama de Flujo de los Procesos del Sistema

La Figura 4.1 muestra el diagrama de flujo del proceso de Servicio Social.

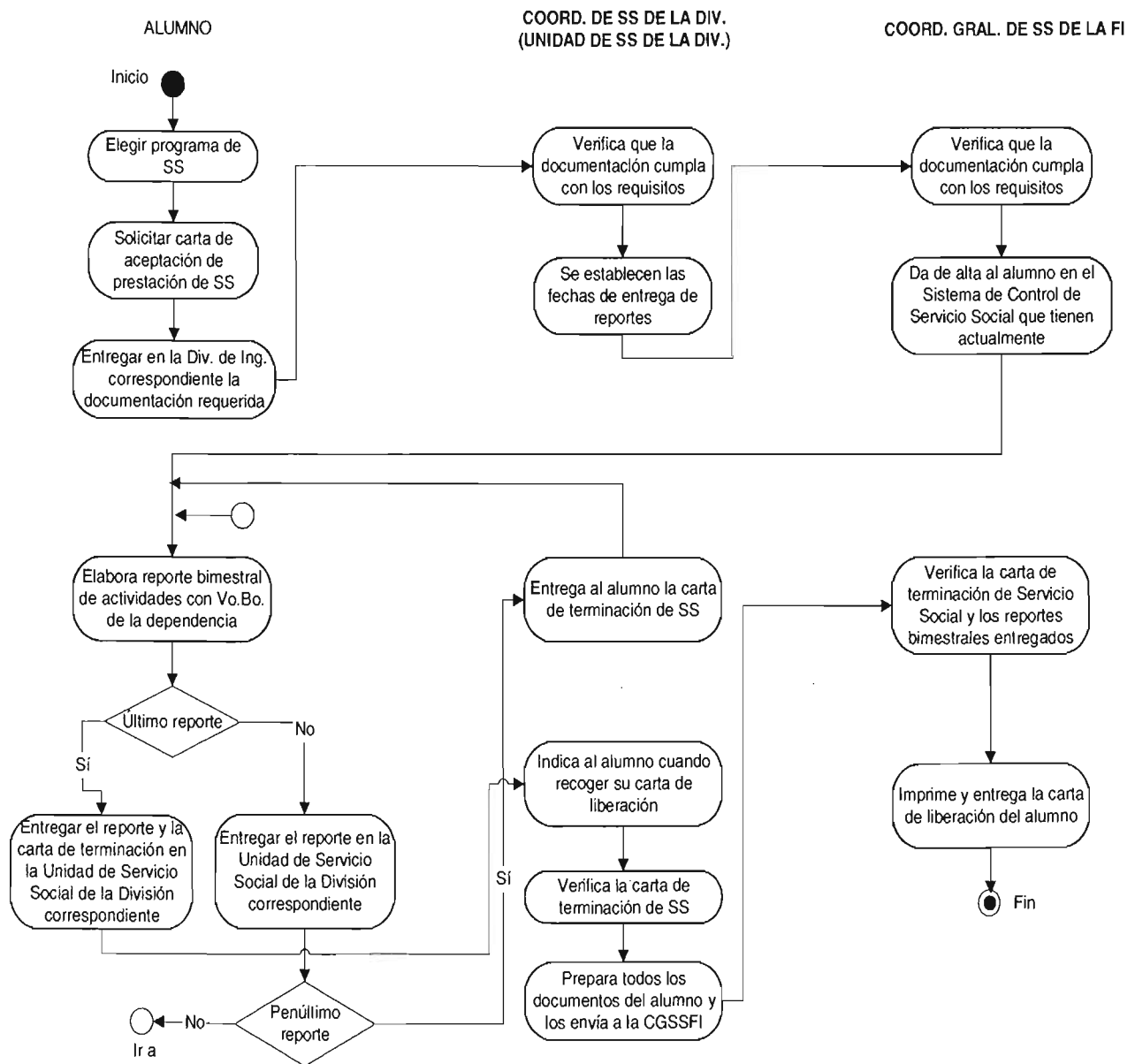


Figura 4.1 Diagrama de Flujo de los Procesos de SS

4.1.4. Diagrama General de Casos de Uso

Se trata de descubrir los *casos de uso* del sistema, así como los actores que iniciarán los casos de uso y los que serán beneficiados.

El diagrama de casos de uso representa la forma en como un Cliente (*Actor*) opera con el sistema, además, muestra también la forma, tipo y orden en como los elementos (operaciones o casos de uso) interactúan.

A continuación se presenta el diagrama general de casos de uso del sistema.

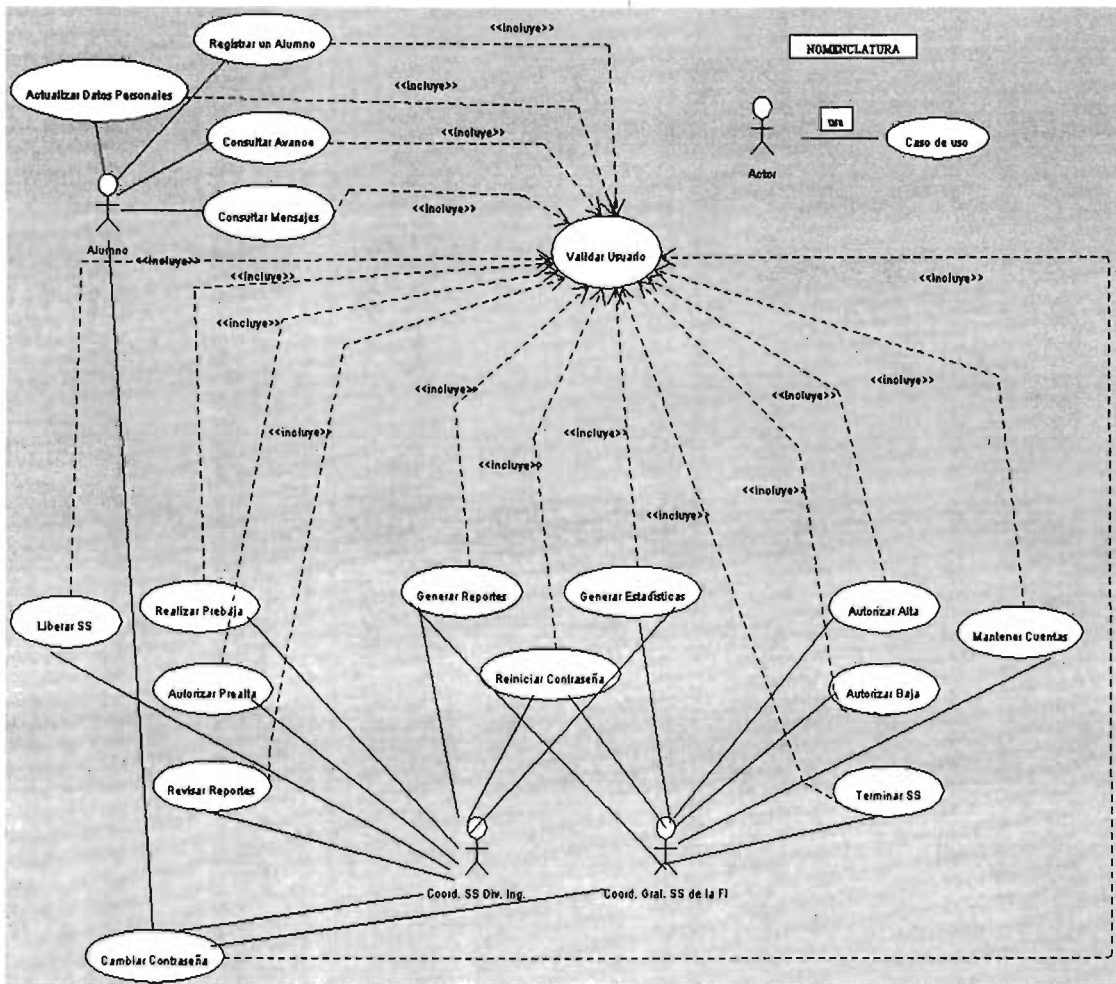


Figura 4.2 Diagrama General de Casos de Uso.

4.1.5. Descripción de los Casos de Uso Esenciales

Los casos de uso esenciales, describen la interacción del usuario con el sistema sin tener como base el punto de vista funcional. Los casos de uso esenciales del Sistema de Administración de SS se presentan a continuación.

Caso de uso: Autorizar el alta de un Alumno en el Sistema de Administración de SS

Actores: Alumno, Coordinador de SS de la División de Ingeniería correspondiente, Jefe de SS de la Facultad de Ingeniería.

Descripción: El Alumno ingresa al sistema y llena la Solicitud de Autorización de Prestación del SS e Información Estadística, con el fin de iniciar su trámite de registro de SS en alguna de las dependencias que le correspondan y en el programa que él elija. Todo esto de acuerdo a la clave de su carrera.

El Coordinador del SS de la División correspondiente al alumno ingresa al sistema, y verifica los datos que el Alumno ingresó en el proceso de registro. Si estos datos son correctos y cumplen con las políticas del SS, el Coordinador de la división de SS aprueba el registro del alumno.

El Jefe de SS ingresa al sistema y selecciona la división a la que pertenece el alumno y verifica nuevamente los datos del alumno. Si estos datos son correctos y cumplen con las políticas del SS acepta el alta definitiva del alumno en su SS notificándole tal proceso.

Caso de uso: Autorizar la baja de un Alumno en el Sistema de SS.

Actores: El Coordinador de SS de la División de Ingeniería correspondiente, Jefe de SS de la Facultad de Ingeniería.

Descripción: El Coordinador de SS de la División de Ingeniería ingresa al sistema y busca a los alumnos que caigan en el incumplimiento de los requisitos mínimos para permanecer ejerciendo su SS, finalmente notifica al Jefe de SS la baja de dicho alumno, indicándole el motivo de la baja. El Jefe de SS ingresa al sistema y selecciona la división a la que pertenece el alumno, verifica los datos del alumno y el motivo por el cual se le da de baja, finalmente el Jefe de SS procede a dar la baja y notifica al alumno de tal proceso.

Caso de uso: Validar la entrega de informes bimestrales del Alumno.

Actores: Coordinador de SS de la División de Ingeniería correspondiente.

Descripción: El Coordinador de SS de la División de Ingeniería ingresa al sistema y busca al alumno que va a entregar algún informe, con base en el informe bimestral entregado, el Coordinador de SS de la División de Ingeniería actualiza el estado del informe.

Caso de uso: Autorizar Liberación de un Alumno en el Sistema de SS

Actores: Coordinador de SS de la División de Ingeniería correspondiente, Jefe de SS

Descripción: El Coordinador de SS de la División de Ingeniería ingresa al sistema, consulta los movimientos realizados por el alumno a lo largo del proceso de SS. Si la información presentada acredita que el alumno ha dado seguimiento a cada uno de los procesos obligatorios en el SS, se notifica al Jefe de SS que el alumno ya está en condiciones de liberar su SS.

El Jefe de SS ingresa al sistema y selecciona la división a la que pertenece el alumno, verifica que el alumno haya cumplido con los procesos obligatorios en el SS, finalmente el Jefe de SS procede a autorizar la liberación

Caso de uso: Consultar Avance de un Alumno en el Sistema de Administración del SS.

Actores: Alumno.

Descripción: El Alumno ingresa al sistema, consulta el estado actual de su proceso de SS, así como todos los movimientos relacionados para el cumplimiento del mismo.

4.1.6. Análisis del Dominio y el Modelo Conceptual

El análisis del dominio tiene como objetivo producir un *modelo conceptual*, el cual, se logra al descomponer el espacio del problema en unidades comprensibles llamadas *conceptos* las cuales se relacionan entre sí por medio de *asociaciones*.

Para llegar al modelo conceptual es necesario identificar los sustantivos, verbos y construcciones verbales. Algunos de los sustantivos se convertirán en conceptos dentro del modelo, y algunos otros en atributos. Los verbos y construcciones verbales se podrán convertir en asociaciones.

De acuerdo a los resultados de las entrevistas y las minutas originadas de las reuniones con el usuario se da origen al *modelo conceptual inicial*, en el cual, se muestran únicamente los conceptos más idóneos ya que el resto pueden convertirse en atributos. En el modelo conceptual se distinguen también las relaciones que existen entre los conceptos, es decir, las asociaciones. Todo lo anterior se ilustra en la siguiente figura:

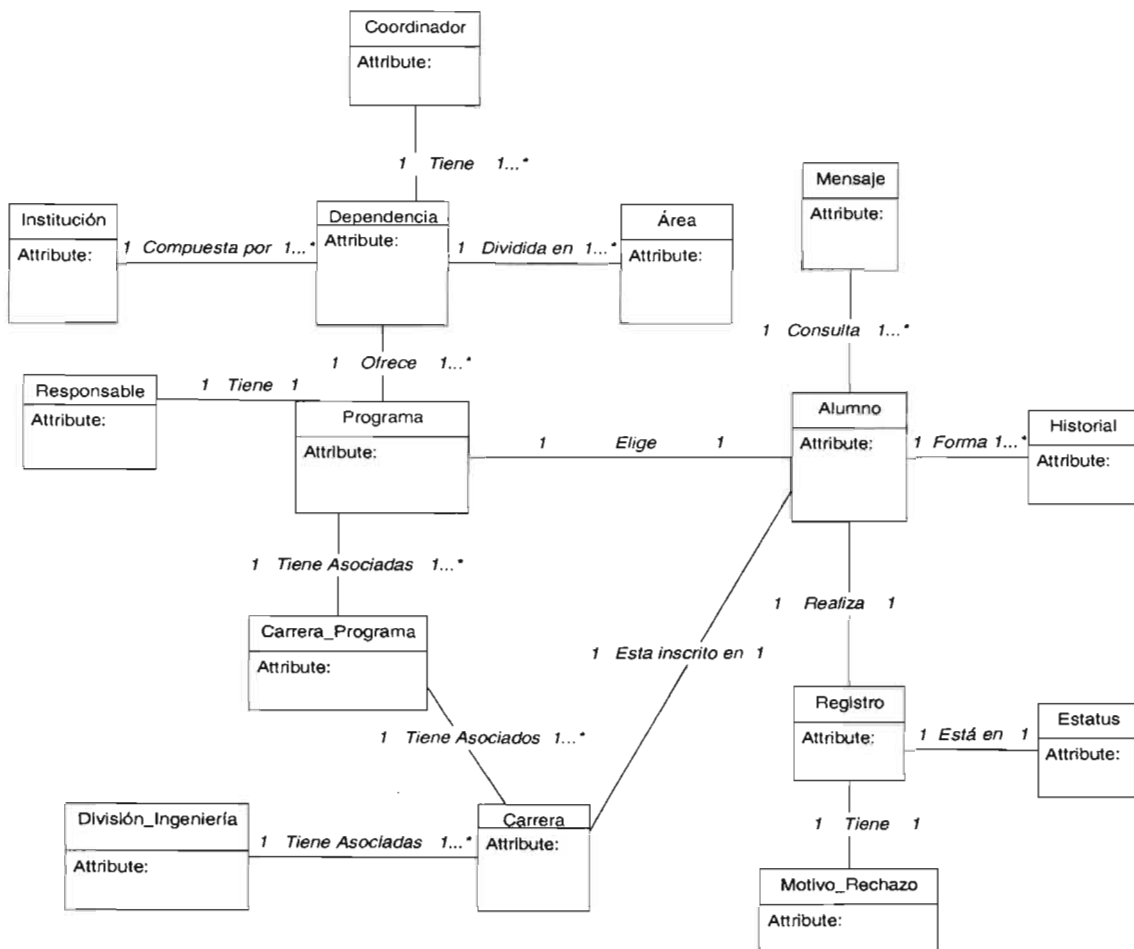


Figura 4.3 Modelo Conceptual del Proceso General de SS.

4.1.7. Diagrama de Flujo del Sistema

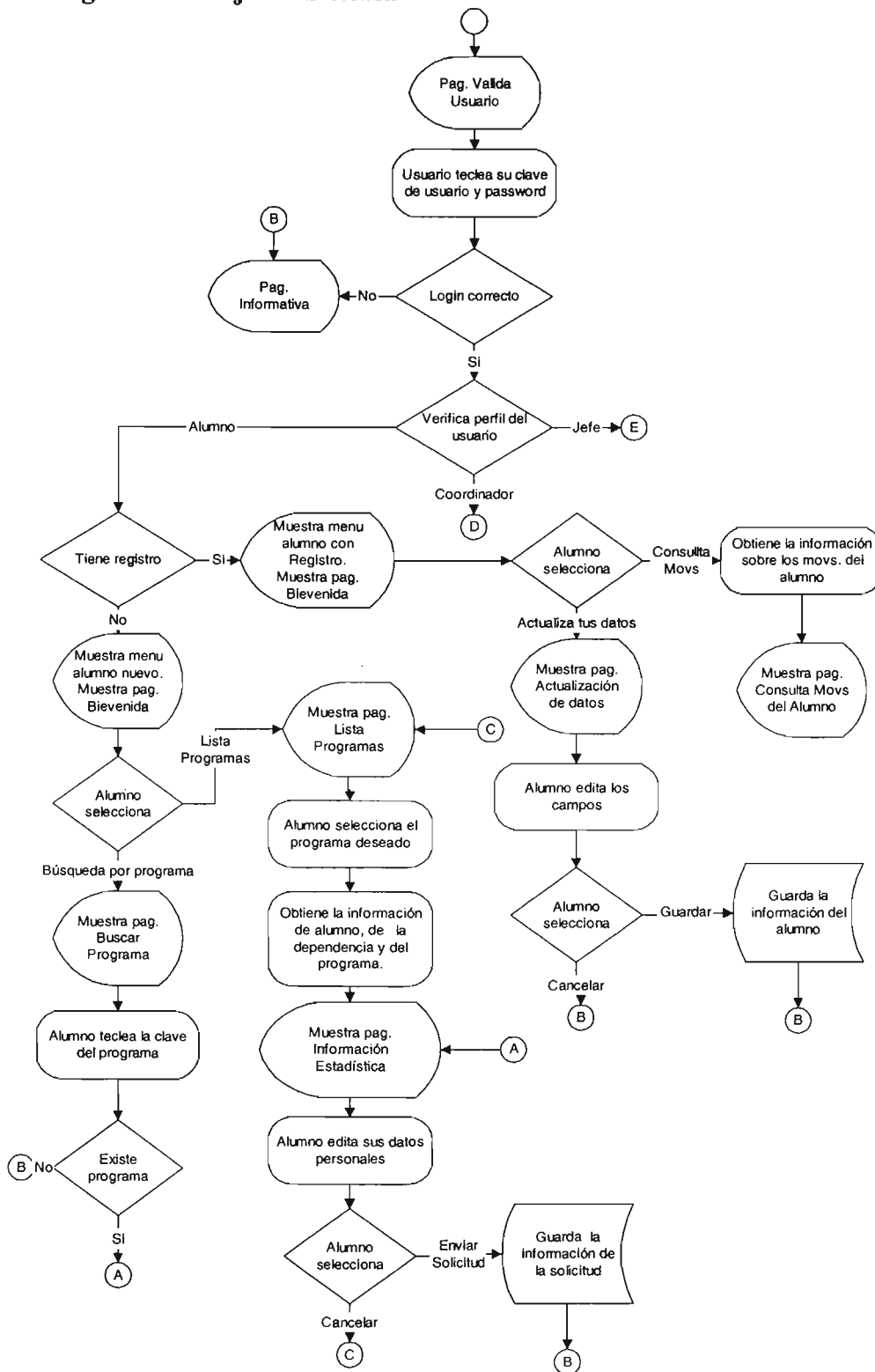


Figura 4.4 Diagrama de Flujo del Sistema (Primera Parte).

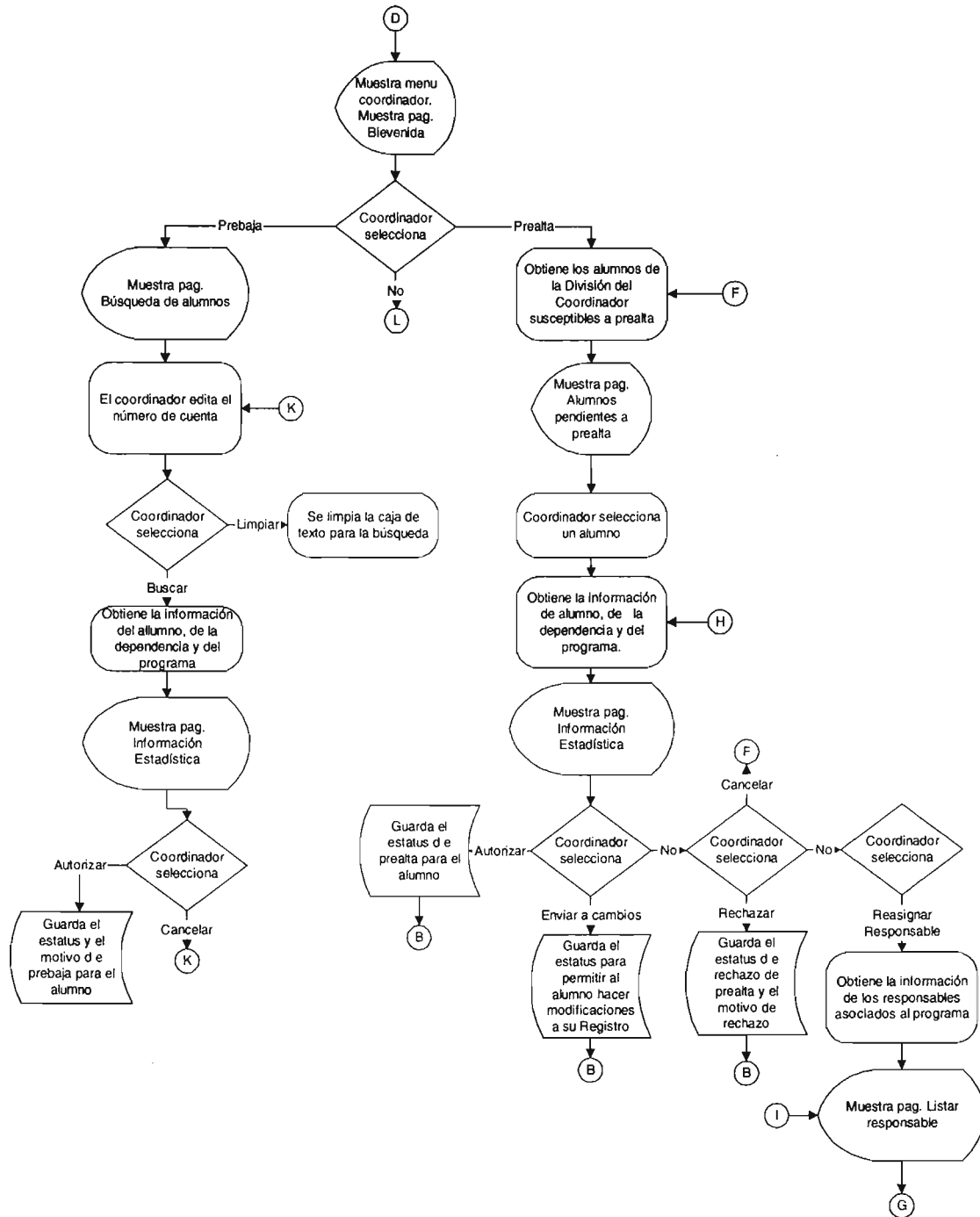


Figura 4.4 Diagrama de Flujo del Sistema (Segunda Parte).

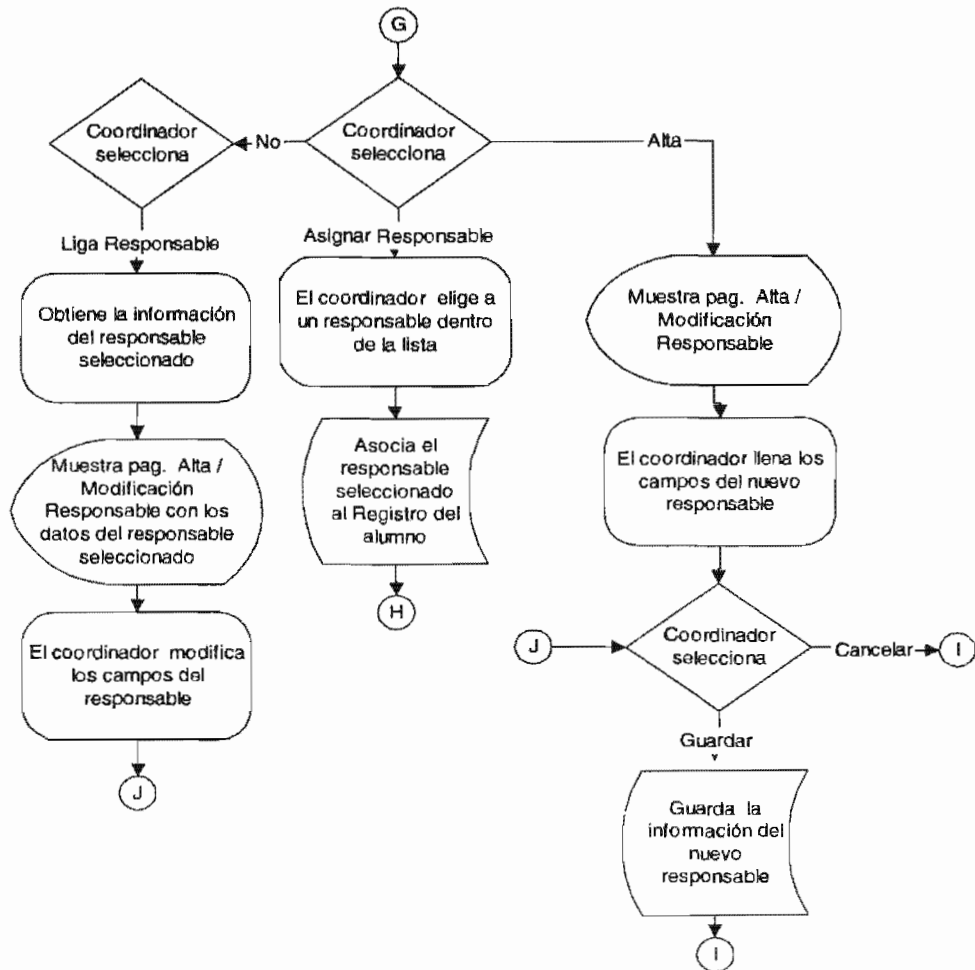


Figura 4.4 Diagrama de Flujo del Sistema (Tercera Parte).

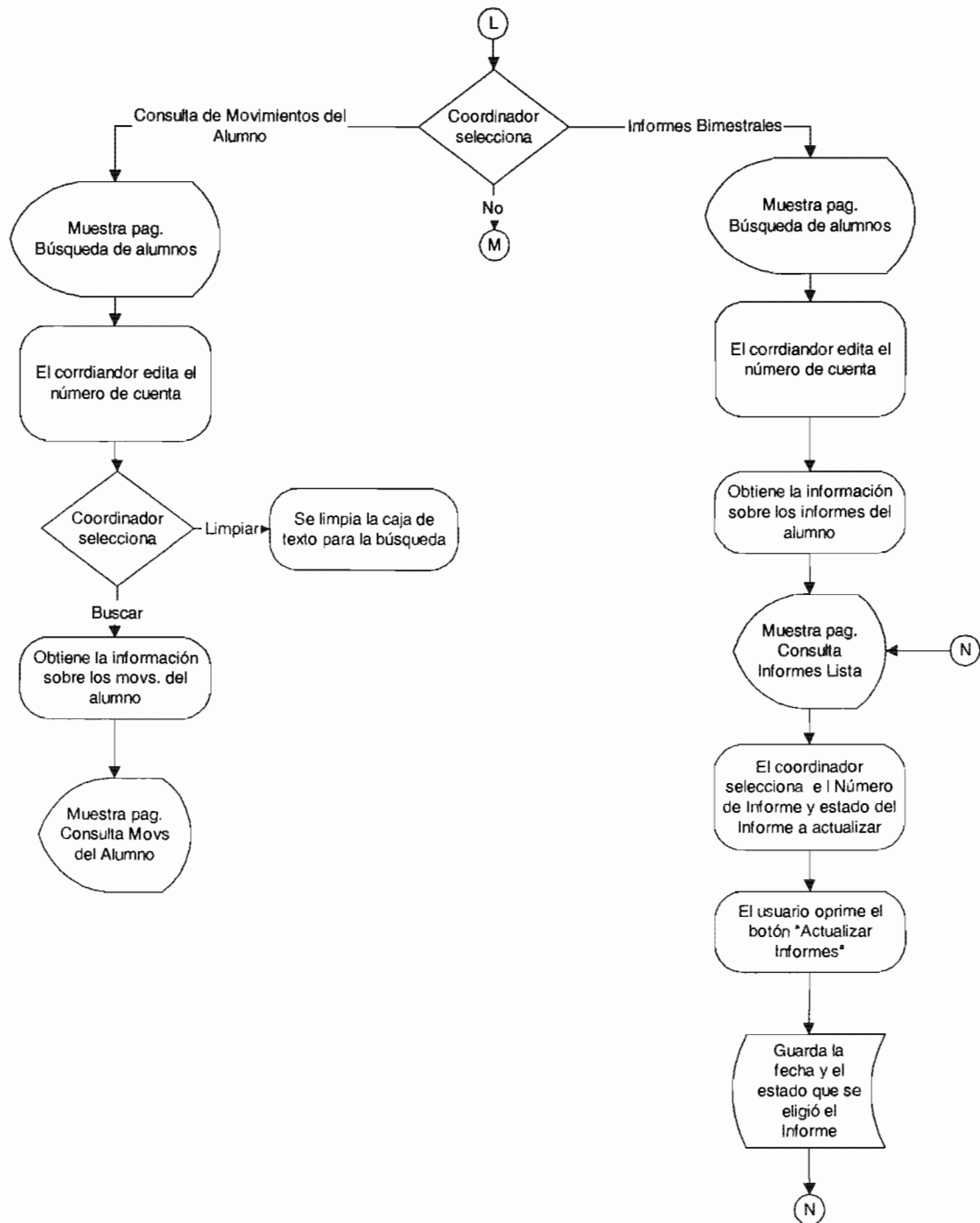


Figura 4.4 Diagrama de Flujo del Sistema (Cuarta Parte).



Figura 4.4 Diagrama de Flujo del Sistema (Quinta Parte).

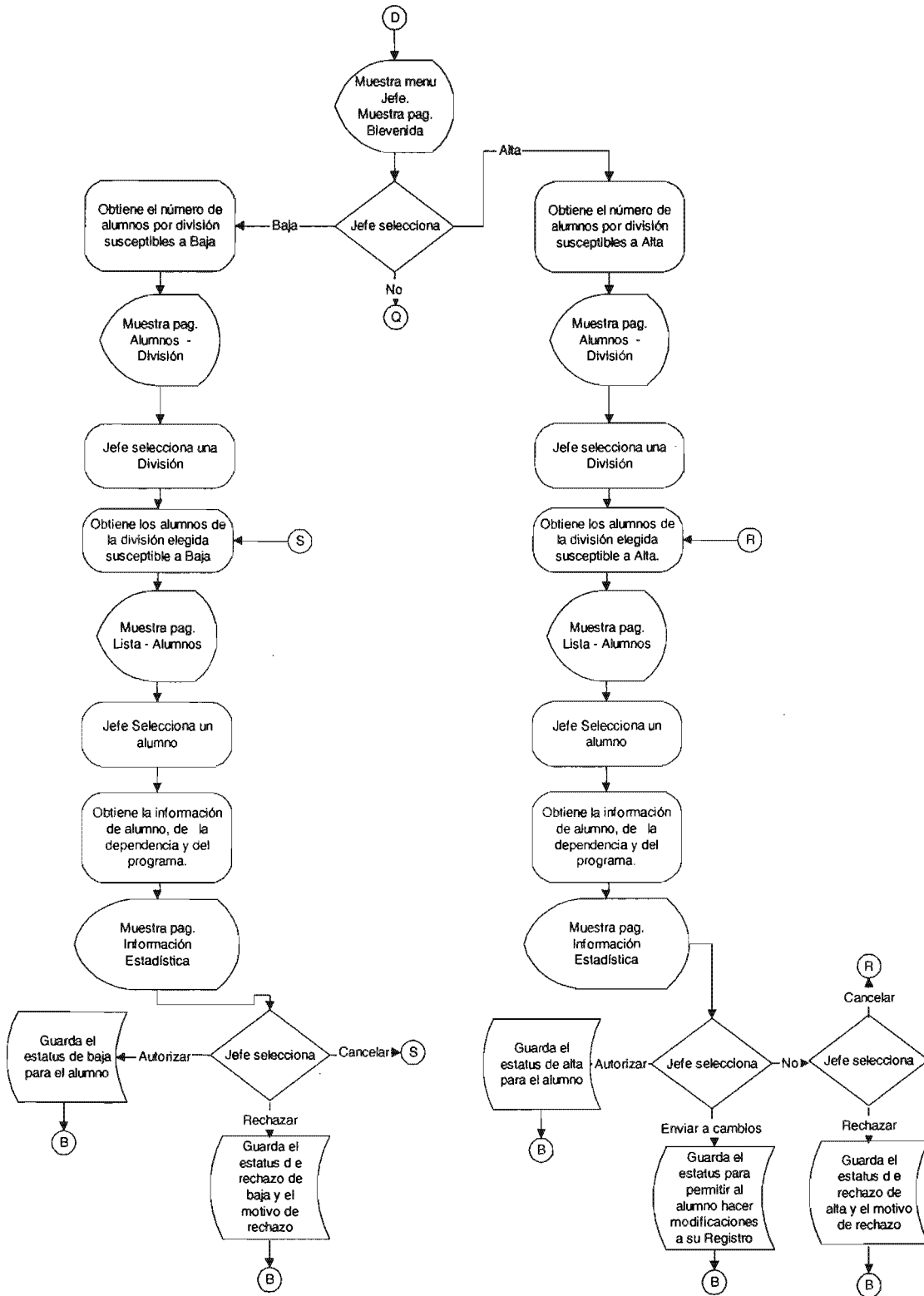


Figura 4.4 Diagrama de Flujo del Sistema (Sexta Parte).



Figura 4.4 Diagrama de Flujo del Sistema (Séptima Parte).



Figura 4.4 Diagrama de Flujo del Sistema (Octava Parte).

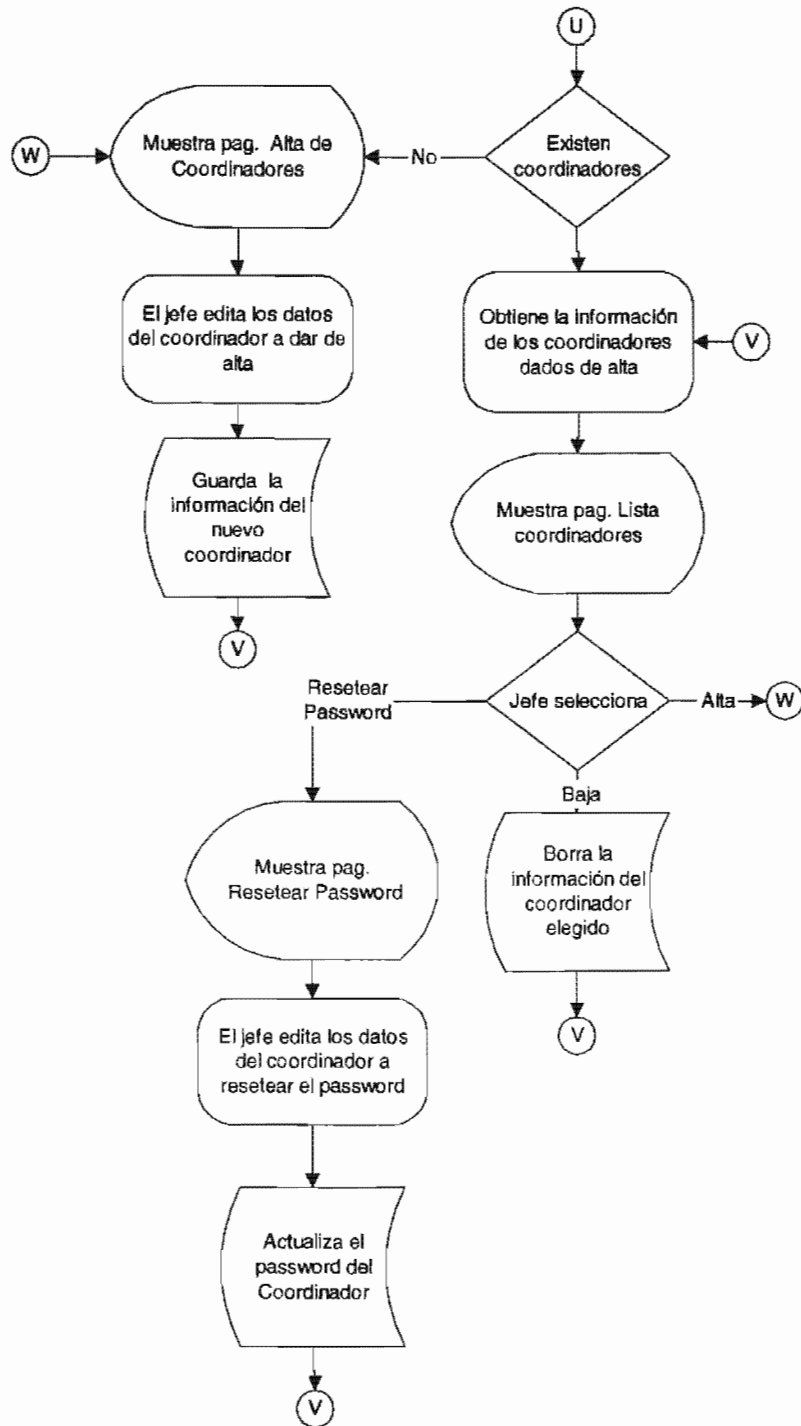


Figura 4.4 Diagrama de Flujo del Sistema (Novena Parte).

4.1.8. Bitácora de Riesgo

La creación del Sistema de Administración de SS (SASS), es un proyecto que conlleva la interacción de diferentes áreas de administrativas de la Facultad de Ingeniería, así como de la Dirección General de Orientación y Servicios Educativos (DGOSE), por lo que existen varios tipos de riesgo que influyen directamente en las diferentes etapas del proyecto. Estos riesgos son los que a continuación se mencionan:

- ◆ Disponibilidad de tiempo por parte del cliente para las reuniones de recopilación de información, presentaciones de avances y pruebas del sistema.
- ◆ Disponibilidad de la información proporcionada por la DGOSE, la cual, forma parte de los catálogos del sistema (Programa, Dependencia, Institución, Área, Coordinador y Responsable).
- ◆ Consistencia, integridad y confiabilidad de la información proporcionada por la DGOSE.
- ◆ Disponibilidad del hardware y software para la implantación del sistema.
- ◆ Curva de aprendizaje de los desarrolladores del sistema.
- ◆ Riesgos de integración del sistema para los desarrolladores.

4.1.9. Lista de Personas

En el contexto del proyecto, principalmente en la fase de análisis, es posible identificar los siguientes roles dentro del sistema:

Cliente

- ◆ Ing. Gonzalo Condado Bernal.
Coordinador General de SS de la Facultad de Ingeniería.
 - ◆ Ing. Elí Israel Hernández García.
Coordinador de SS de la División de Ciencias Básicas.
 - ◆ Ing. Luis Candelas Ramírez e Ing. Heriberto Esquivel Castellanos.
Coordinadores de SS de la División de Ingeniería Civil, Topográfica y Geodésica.
 - ◆ Ing. Víctor Manuel Vázquez Huarota.
Coordinador de SS de la División de Ingeniería Mecánica e Industrial.
 - ◆ Ing. José Arturo Origel Coutiño.
Coordinador de SS de la División de Ingeniería Eléctrica.
 - ◆ Ing. Alfredo Victoria Morales.
Coordinador de SS de la División de Ingeniería en Ciencias de la Tierra.
-

Proveedor

- ◆ Galicia González Gpe. Miriam
- ◆ Bruno David Vázquez Chávez

Usuarios del Sistema

- ◆ Coordinación General de Servicio Social de la Facultad de Ingeniería.
- ◆ Coordinaciones de Servicio Social de las Divisiones de Ingeniería.
- ◆ Comunidad de Alumnos de la Facultad de Ingeniería.

Administrador

- ◆ Administrador del Sistema
- ◆ Administrador de la Base de Datos.

4.2. Diseño del Sistema

4.2.1. Análisis de Requerimientos del Diseño

Como parte importante del diseño del sistema deberá elaborarse la descripción de los casos de uso reales, es decir, aquellos que describen la interacción entre el usuario y el nuevo sistema desde un punto de vista funcional. A continuación, se describen los casos de uso para cada uno de los módulos del sistema.

Clave: SASS01AU.

Caso de uso: Validar Usuario.

Casos de uso incluidos: Ninguno.

Actores: Alumno, Coordinador de SS de la División de Ingeniería, Jefe de SS de la FI y Administrador del Sistema.

Descripción: El sistema realiza la validación del usuario que intenta ingresar a la aplicación, si es un usuario válido permite el acceso y le habilita las opciones correspondientes de acuerdo a su perfil, en caso contrario despliega el mensaje de error correspondiente e impide el acceso.

Curso Normal de los Eventos	
Acción del Actor	Respuesta del Sistema
1. El usuario da un clic en la liga "Sistema de Administración del Servicio Social" para entrar al sistema	2. El sistema despliega una pantalla de validación de usuario.
3. El usuario teclea su clave de usuario y contraseña.	4. El sistema verifica la validez de los datos ingresados por el usuario.
	5. Si el usuario es válido, el sistema habilita las opciones que corresponden a su perfil de acuerdo a la definición de grupos del sistema.
Cursos Alternos	
Línea 4: Si el usuario es no es válido, el sistema muestra una pantalla con el mensaje correspondiente, con lo que se impide el acceso al sistema.	

Clave: SASS02R.

Caso de uso: Registrar un Alumno en el Sistema de Administración de SS.

Casos de uso incluidos: Validar Usuario.

Actores: Alumno.

Descripción: El Alumno ingresa al sistema y llena la Solicitud de Autorización de Prestación del SS e Información Estadística, con el fin de iniciar su trámite de registro de SS en alguna de las dependencias que le correspondan y en el programa que él elija. Todo esto de acuerdo a la clave de su carrera.

Curso Normal de los Eventos	
Acción del Actor	Respuesta del Sistema
1. En el módulo de registro, el usuario elige entre las opciones de "Buscar un programa" en particular o "Lista programas" disponibles.	2. El sistema despliega la pantalla correspondiente a la opción seleccionada por el alumno.

3. Si eligió la búsqueda individual, entonces ingresa los datos que componen la clave del programa en cuestión y procede a realizar la búsqueda.	4. El sistema realiza la búsqueda y despliega una pantalla los resultados de la misma.
5. El alumno da clic en la liga del programa en el que desea registrarse.	6. Muestra una pantalla con el formato de la Solicitud de Autorización de Prestación del SS e Información Estadística
	7. Los datos principales del alumno, los del programa y la dependencia se encontrarán previamente llenados en la solicitud.
8. Teclea la información requerida en los todos los campos que se encuentren vacíos en cualquiera de las tres secciones de la solicitud.	9. El número de horas a la semana se calcula automáticamente de acuerdo al número de meses de duración de la prestación del servicio social.
10. Revisa la información capturada y procede a enviar su registro.	11. El sistema valida que el porcentaje de créditos sea el adecuado de acuerdo al programa de SS
	12. Si el programa es en la Facultad de Ingeniería, validar que el porcentaje sea de al menos 35%. Si el programa es fuera de la Facultad de Ingeniería, validar que el porcentaje coincida con el estipulado en el Programa elegido.
	13. Envía una nueva pantalla con el mensaje correspondiente indicando el éxito o fracaso de la transacción

Cursos Alternos

Línea 2: Si el usuario elige la opción para consultar la lista de programas, entonces el flujo de los eventos pasa a la línea 4.

Línea 7: Si el alumno ya ha registrado el SS en alguna ocasión, entonces todos los datos personales del alumno estarán previamente llenados en la solicitud.

Línea 10: Si no desea enviar su registro, entonces procede a cancelar la operación.

Clave: SASS03PA.

Caso de uso: Autorizar Prealta de un Alumno en el Sistema de Administración del SS.

Casos de uso incluidos: Validar Usuario.

Actores: Coordinador de SS de la División de Ingeniería.

Descripción: El Coordinador del SS de la División verifica los datos que el Alumno ingresó en el proceso de registro. Si estos datos son correctos y cumplen con las políticas del SS, se acepta la prealta del alumno en el sistema.

Curso Normal de los Eventos

Acción del Actor	Respuesta del Sistema
1. El usuario elige el módulo de "Prealta".	2. El sistema despliega en la pantalla principal la lista de los alumnos que se encuentran en espera de autorización de prealta.
3. Selecciona el número de cuenta del alumno del cual desea revisar su	4. Muestra en pantalla la Solicitud de Autorización de Prestación del SS e

información.	Información Estadística llenada previamente por el alumno.
5. Revisa que los datos proporcionados por el alumno en la solicitud, cumplan con los lineamientos del SS y procede a autorizar la solicitud.	6. El sistema ejecuta el proceso de autorización de prealta para la solicitud.
	7. Se envía el mensaje correspondiente al éxito o fracaso de la transacción.
8. El usuario elige entre seguir trabajando con cualquiera de las opciones del menú principal o salir del sistema.	
Cursos Alternos	
Línea 5: Si los datos no cumplen con los lineamientos, entonces el Coordinador de SS de la División debe capturar los motivos de rechazo en el campo creado para dicho fin en la solicitud y entonces rechazar la solicitud.	
Línea 6: Si el usuario no autoriza la solicitud y decide rechazarla, el sistema ejecuta el proceso correspondiente al rechazo de la solicitud. Para esto, se verifica que el campo de motivos de rechazo haya sido llenado por el Coordinador de SS de la División, de lo contrario el sistema envía un aviso y no realiza el movimiento.	

Clave: SASS04A.

Caso de uso: Autorizar Alta de un Alumno en el Sistema de Administración del SS.

Casos de uso incluidos: Validar Usuario.

Actores: Jefe de SS de la FI

Descripción: El Jefe de SS de la FI verifica los datos que fueron validados en el proceso de prealta. Si estos datos son correctos y cumplen con las políticas del SS acepta el alta definitiva del alumno en el sistema.

Curso Normal de los Eventos	
Acción del Actor	Respuesta del Sistema
1. El usuario elige el módulo de "Alta".	2. El sistema despliega una lista con los nombres de las Divisiones de Ingeniería de la Facultad.
3. Elige la División de Ingeniería de su interés.	4. El sistema despliega la lista de los alumnos susceptibles al proceso de alta, es decir, aquellos cuya prealta ya fue aceptada.
5. Selecciona el registro del alumno que desea revisar.	6. Muestra en pantalla la Solicitud de Autorización de Prestación del SS e Información Estadística llenada previamente por el alumno y verificada por el Coordinador de SS de la División de Ingeniería.
7. Revisa que los datos contenidos en la solicitud, cumplan con los lineamientos del SS y procede a	8. El sistema ejecuta el proceso de autorización de alta para la solicitud.

autorizar la solicitud.	
	9. Se envía el mensaje correspondiente al éxito o fracaso de la transacción.
10. El usuario puede seguir trabajando con cualquiera de las opciones del menú principal, incluso salir del sistema.	
Cursos Alternos	
Línea 7: Si los datos no cumplen con los lineamientos, entonces el Jefe de SS de la FI debe capturar los motivos de rechazo en el campo creado con dicho fin en la solicitud y entonces rechazar la solicitud.	
Línea 8: Si el usuario decide rechazarla, el sistema ejecuta el proceso correspondiente al rechazo de la solicitud. Para esto, se verifica que los motivos de rechazo hayan sido ingresados por el Jefe de SS de la FI, de lo contrario el sistema envía un aviso y no realiza el movimiento.	

Clave: SASS05PB.

Caso de uso: Realizar Prebaja de un Alumno en el Sistema de Administración del SS.

Actores: Coordinador de SS de la División de Ingeniería.

Casos de uso incluidos: Validar Usuario.

Descripción: El Coordinador del SS de la División de Ingeniería realiza la prebaja de un alumno indicando los motivos de la misma.

Curso Normal de los Eventos	
Acción del Actor	Respuesta del Sistema
1. El usuario elige el módulo de "Prebaja" en el menú principal.	2. Presenta una pantalla para capturar el número de cuenta del alumno a buscar.
3. El usuario ingresa el número de cuenta del alumno y procede a realizar la búsqueda.	4. Como resultado de la búsqueda, muestra la Solicitud de Autorización de Prestación del SS e Información Estadística del alumno.
5. El usuario captura los motivos de prebaja y procede a aceptar el movimiento.	6. El sistema valida que el usuario haya ingresado los motivos de prebaja y ejecuta la transacción.
	7. Se despliega un mensaje indicando si la prebaja se realizó con éxito o no.
8. Elige cualquiera de las opciones del menú principal o incluso salir del sistema.	
Cursos Alternos	
Línea 5: Si no se desea aceptar la prebaja, entonces se elige la opción de cancelar, con lo que la aplicación no realiza movimiento alguno.	
Línea 6: Si no se ingresaron los motivos de prebaja, se envía el aviso correspondiente y el sistema no realiza movimiento alguno.	

Clave: SASS06B.

Caso de uso: Autorizar Baja de un Alumno en el Sistema de Administración de SS.

Actores: Jefe de SS de la FI

Casos de uso incluidos: Validar Usuario.

Descripción: El Jefe de SS de la FI realiza la baja de un alumno.

Curso Normal de los Eventos	
Acción del Actor	Respuesta del Sistema
1. Elige la opción "Baja" en el menú principal.	2. El sistema despliega una lista con los nombres de las Divisiones de Ingeniería de la Facultad.
3. Elige la División de Ingeniería de su interés.	4. El sistema despliega una lista de los alumnos cuya baja fue solicitada previamente por los Coordinadores de las Divisiones de Ingeniería.
5. El usuario elige el registro del alumno que desea revisar.	6. Se muestra en pantalla la Solicitud de Autorización de Prestación del SS e Información Estadística del alumno.
7. El usuario autoriza el movimiento.	8. El sistema procede a ejecutar el proceso de baja.
	9. El sistema envía el mensaje de éxito o fracaso de la operación.
10. Elige cualquiera de las opciones del menú principal o incluso salir del sistema.	
Cursos Alternos	
Línea 7: Si no desea aceptar la baja, el usuario elige la opción de cancelar con lo cual el sistema no efectuará movimiento alguno.	

Clave: SASS07IC.

Caso de uso: Ingresar Comentarios de un Alumno en el Sistema de Administración del SS.

Actores: Coordinador de SS de la División de Ingeniería, Jefe de SS de la FI

Casos de uso incluidos: Validar Usuario.

Descripción: Tanto los Coordinadores de cada División de Ingeniería como el Jefe de SS de la FI podrán ingresar al sistema comentarios adicionales sobre los alumnos, los cuales les permitirán registrar información extra que consideren necesaria para las labores administrativas que desempeñan.

Curso Normal de los Eventos	
Acción del Actor	Respuesta del Sistema
1. Elige la opción "Comentarios" en el menú principal.	2. El sistema muestra la pantalla de captura de comentarios.
3. El usuario ingresa el número de cuenta y los comentarios que desea registrar. Procede a enviar los datos.	4. Ejecuta el proceso para registrar los comentarios acerca del alumno.
	5. El sistema envía el mensaje con el resultado de la operación.

6. Elige cualquiera de las opciones del menú principal o incluso salir del sistema.	
Cursos Alternos	
Línea 4: Si el Coordinador intenta registrar comentarios para un alumno que no le corresponde, el sistema no realiza la operación y envía el mensaje correspondiente.	

Clave: SASS08GE.

Caso de uso: Generar Estadísticas.

Actores: Coordinador de SS de la División de Ingeniería, Jefe de SS de la FI

Casos de uso incluidos: Validar Usuario.

Descripción: Los Coordinadores de SS generan diferentes estadísticas para conocer el comportamiento del ejercicio del SS en la FI

Curso Normal de los Eventos	
Acción del Actor	Respuesta del Sistema
1. Elige la opción "Estadísticas" en el menú principal.	2. El sistema muestra la pantalla de selección de parámetros para la generación de las estadísticas.
3. El usuario selecciona la estadística que desea generar así como las columnas que desea visualizar y el intervalo de fechas que se desea consultar. Procede a generar la estadística.	4. Los resultados de la consulta son generados por el sistema y desplegados en la pantalla principal.
5. Elige cualquiera de las opciones del menú principal o incluso salir del sistema.	
Cursos Alternos	
Línea 4: En caso de existir algún error durante la generación de la consulta el sistema desplegará el mensaje de error correspondiente a la causa del mismo.	

Clave: SASS09GR.

Caso de uso: Generar Reportes.

Actores: Coordinador de SS de la División de Ingeniería, Jefe de SS de la FI

Casos de uso incluidos: Validar Usuario.

Descripción: Los Coordinadores de SS generan diferentes reportes para conocer el comportamiento del ejercicio del SS en la FI.

Curso Normal de los Eventos	
Acción del Actor	Respuesta del Sistema
1. Elige la opción "Reportes" en el menú principal.	2. El sistema despliega la pantalla de selección de parámetros como lo son: Nombre del reporte, estatus, fecha inicial, fecha final, división de ingeniería, carrera, generación y cuenta.
3. El usuario selecciona el reporte que desea generar así como los estatus con el que desea generarlo y el intervalo de fechas a consultar (el resto de los parámetros son	4. Los resultados de la consulta son generados por el sistema y desplegados en la pantalla principal.

opcionales). Procede a generar el reporte.	
5. Elige cualquiera de las opciones del menú principal o incluso salir del sistema.	
Cursos Alternos	
Línea 4: En caso de existir algún error durante la generación de la consulta el sistema desplegará el mensaje de error correspondiente a la causa del mismo.	

Clave: SASS10AD.

Caso de uso: Actualizar Datos Personales del Alumno.

Actores: Alumno.

Casos de uso incluidos: Validar Usuario.

Descripción: El Alumno ingresa al sistema para actualizar sus datos personales en caso de que alguno de éstos haya cambiado.

Curso Normal de los Eventos	
Acción del Actor	Respuesta del Sistema
1. Elige la opción "Actualiza tus Datos" en el menú principal.	2. El sistema despliega los datos del alumno que son susceptibles de actualizarse.
3. El usuario edita la información que desee modificar y envía los cambios al sistema.	4. Se realiza el proceso de actualización de datos del alumno.
	5. El sistema envía el mensaje correspondiente al resultado de la operación.
6. Elige cualquiera de las opciones del menú principal o incluso salir del sistema.	
Cursos Alternos	
Línea 3: Si no desea enviar las modificaciones al sistema elige la opción cancelar con la cual el sistema no realiza modificaciones a los datos del alumno.	

Clave: SASS11RC.

Caso de uso: Reiniciar Contraseña del Alumno.

Actores: Coordinador de SS de la División de Ingeniería, Jefe de SS de la FI

Casos de uso incluidos: Validar Usuario.

Descripción: El Coordinador de SS de la División de Ingeniería modifica la contraseña de un alumno cuando éste se lo solicita, éste Coordinador puede modificar sólo a los alumnos de los cuales administra su proceso de SS. El Jefe de SS de la FI puede modificar la contraseña de cualquier alumno.

Curso Normal de los Eventos	
Acción del Actor	Respuesta del Sistema
1. Elige la opción "Reiniciar Contraseña" en el menú principal.	2. El sistema despliega la pantalla de captura de datos para el reinicio de contraseña.
3. El usuario ingresa el número de cuenta y la nueva contraseña del alumno. Entonces procede a realizar el cambio.	4. Se ejecuta el proceso de actualización de contraseña del alumno.
	5. El sistema envía el mensaje correspondiente al resultado de la operación.
6. Elige cualquiera de las opciones del menú	

principal o incluso salir del sistema.
Cursos Alternos
Línea 3: Si no desea enviar las modificaciones al sistema elige la opción cancelar con la cual el sistema no realiza cambio alguno la contraseña del alumno.
Línea 4: Si el Coordinador de SS de la División de Ingeniería intenta reiniciar la contraseña de un alumno que no le corresponde, el sistema le envía un mensaje con este aviso.

Clave: SASS12CC.

Caso de uso: Cambiar Contraseña del Usuario.

Actores: Alumno, Coordinador de SS de la División de Ingeniería, Jefe de SS de la FI

Casos de uso incluidos: Validar Usuario.

Descripción: Los diferentes usuarios del sistema modifican su contraseña cuando así lo requieren.

Curso Normal de los Eventos	
Acción del Actor	Respuesta del Sistema
1. Elige la opción "Cambiar Contraseña" en el menú principal.	2. El sistema despliega la pantalla de captura de datos para el cambio de contraseña.
3. El usuario ingresa su contraseña actual, la contraseña nueva y la confirmación. Entonces procede a realizar el cambio.	4. Se ejecuta el proceso de cambio de la contraseña del usuario.
	5. El sistema envía el mensaje correspondiente al resultado de la operación.
6. Elige cualquiera de las opciones del menú principal o incluso salir del sistema.	
Cursos Alternos	
Línea 3: Si no desea enviar las modificaciones al sistema elige la opción cancelar con la cual el sistema no realiza cambio alguno a la contraseña del alumno.	

Clave: SASS13T.

Caso de uso: Autorizar Terminación de un Alumno en el Sistema de Administración del SS.

Casos de uso incluidos: Autenticar Usuario.

Actores: Coordinador de SS de la División de Ingeniería.

Descripción: El Coordinador de SS de la División de Ingeniería verifica los movimientos realizados por el alumno a lo largo del proceso de SS. Si la información presentada acredita que el alumno ha dado seguimiento a cada uno de los procesos obligatorios en el SS, se acepta la terminación del alumno en el sistema.

Curso Normal de los Eventos	
Acción del Actor	Respuesta del Sistema
1. El usuario elige el módulo de "Terminación".	2. El sistema despliega en la pantalla principal la lista de los alumnos que se encuentran en espera de autorización de terminación.
3. Selecciona el número de cuenta del alumno del cual desea revisar su información.	4. Muestra en pantalla la Consulta de los movimientos realizados por el alumno a lo largo del proceso de SS.

5. Revisa que la información del alumno cumpla con los procesos obligatorios en el ejercicio del SS.	6. El sistema ejecuta el proceso de autorización de la terminación del SS.
	7. Se envía el mensaje correspondiente al éxito o fracaso de la transacción.
8. El usuario elige entre seguir trabajando con cualquiera de las opciones del menú principal o salir del sistema.	

Clave: SASS14L.

Caso de uso: Autorizar Liberación de un Alumno en el Sistema de Administración del SS.

Casos de uso incluidos: Autenticar Usuario, Consultar Avance de un Alumno en el Sistema de Administración del SS.

Actores: Jefe de SS de la FI

Descripción: El Jefe de SS de la FI verifica la información validada en el proceso de terminación. Si la información contiene los procesos obligatorios del SS y se cuenta con la documentación necesaria que acredite que el alumno ya puede tramitar la liberación del SS, se acepta la liberación del alumno en el sistema.

Curso Normal de los Eventos	
Acción del Actor	Respuesta del Sistema
1. El usuario elige el módulo de "Liberación".	2. El sistema despliega una lista con los nombres de las Divisiones de Ingeniería de la Facultad.
3. Elige la División de Ingeniería de su interés.	4. El sistema despliega la lista de los alumnos susceptibles al proceso de liberación.
5. Selecciona el registro del alumno que desea revisar.	6. Muestra en pantalla la Consulta de los movimientos realizados por el alumno a lo largo del proceso de SS.
7. Revisa que la información del alumno contenga los procesos obligatorios en el ejercicio del SS así como la documentación que acredite que el alumno ya puede tramitar la liberación del SS.	8. El sistema ejecuta el proceso de autorización de liberación.
	9. Se envía el mensaje correspondiente al éxito o fracaso de la transacción.
10. El usuario puede seguir trabajando con cualquiera de las opciones del menú principal, incluso salir del sistema.	

Cursos Alternos
Línea 7: Si la información del alumno tiene faltas en el cumplimiento del proceso del SS, o bien, no se cuenta con la documentación necesaria que acredite que el alumno ya puede tramitar la liberación del mismo, el Coordinador de SS de la División debe capturar los motivos de rechazo en el campo creado para dicho fin y entonces rechazar la liberación.
Línea 8: Si el usuario decide rechazarla, el sistema ejecuta el proceso correspondiente al rechazo de la liberación. Para esto, se verifica que los motivos de rechazo hayan sido ingresados por el Jefe de SS de la FI, de lo contrario el sistema envía un aviso y no realiza el movimiento.

Clave: SASS15CA.

Caso de uso: Consultar Avance de un Alumno en el Sistema de Administración del SS.

Casos de uso incluidos: Autenticar Usuario.

Actores: Alumno.

Descripción: El Alumno consulta el estado actual de su proceso de SS así como todos los movimientos relacionados para el cumplimiento del mismo.

Curso Normal de los Eventos	
Acción del Actor	Respuesta del Sistema
1. El usuario elige el módulo de "Consultas".	2. El sistema despliega en la pantalla principal la historia actual de movimientos durante el proceso de SS, número de informes con sus fechas límite de entrega, fecha estimada de terminación, movimientos rechazados por el Coordinador de SS de la División de Ingeniería y el Jefe de SS de la FI, estado actual en el que se encuentra el alumno.
3. El usuario puede seguir trabajando con cualquiera de las opciones del menú principal, incluso salir del sistema.	

Clave: SASS16CM.

Caso de uso: Consultar Mensajes en el Sistema de Administración del SS.

Casos de uso incluidos: Autenticar Usuario.

Actores: Alumno, Coordinador de SS de la División de Ingeniería.

Descripción: El usuario consulta los mensajes enviados por el Coordinador de SS de la División de Ingeniería o por el Jefe de SS de la FI referentes a rechazo de movimientos durante el proceso de SS.

Curso Normal de los Eventos	
Acción del Actor	Respuesta del Sistema
1. El usuario elige la opción "Usted tiene N mensajes sin leer", donde N es el número de mensajes pendientes de leer.	2. El sistema despliega en la pantalla principal la lista de los mensajes asociados al usuario.

3. El usuario elige el mensaje a consultar.	4. El sistema despliega el detalle del mensaje a consultar
5. El usuario elige la opción regresar.	6. El sistema despliega en la pantalla principal la lista de los mensajes asociados al usuario.
7. El usuario puede seguir trabajando con cualquiera de las opciones del menú principal, incluso salir del sistema.	
Cursos Alternos	
Línea 3: El usuario decide seleccionar uno o todos los mensajes para borrarlos.	
Línea 4: El sistema despliega los mensajes relacionados al usuario o bien la leyenda "Usted no tiene mensajes".	
Línea 5: El usuario elige la opción borrar mensaje.	
Línea 6: El sistema despliega los mensajes relacionados al usuario o bien la leyenda "Usted no tiene mensajes".	

Clave: SASS17AI.

Caso de uso: Actualizar el Estado de los Informes Bimestrales de un Alumno en el Sistema de Administración del SS.

Casos de uso incluidos: Autenticar Usuario.

Actores: Coordinador de SS de la División de Ingeniería.

Descripción: Con base en el informe bimestral del Alumno entregado a la coordinación de su división, el Coordinador de SS de la División de Ingeniería actualiza el estado del informe.

Curso Normal de los Eventos	
Acción del Actor	Respuesta del Sistema
1. El usuario elige el módulo de "Informes".	2. El sistema despliega una pantalla de búsqueda.
3. El usuario ingresa el número de cuenta del alumno al cual se le quiere actualizar sus informes.	4. El sistema realiza la búsqueda y despliega una pantalla con información relacionada a los informes bimestrales: fecha límite de entrega, fecha de entrega y estatus.
5. El usuario selecciona el informe bimestral. y elige el estatus correspondiente de acuerdo al Vo.Bo. deliberado por el mismo.	6. El sistema realiza la actualización del estatus del informe.
	7. El sistema despliega una pantalla con la información actualizada de los informes bimestrales: fecha límite de entrega, fecha de entrega y estatus.
8. El usuario puede seguir trabajando con cualquiera de las opciones del menú principal, incluso salir del sistema.	

Clave: SASS18AR.

Caso de uso: Asociar Nuevo Responsable de Programa de SS a un Alumno.

Casos de uso incluidos: Autorizar Prealta de un Alumno en el Sistema de Administración del SS.

Actores: Coordinador de SS de la División de Ingeniería.

Descripción: Con base en el registro realizado por el alumno y de acuerdo a la disponibilidad de los responsables de programa de SS, el Coordinador de SS de la División decidirá qué responsable asignar al alumno, y si se incorpora uno nuevo, tendrá la posibilidad de darlo de alta y asignarlo al alumno.

Curso Normal de los Eventos	
Acción del Actor	Respuesta del Sistema
1. Presiona el botón "Reasignar responsable".	2. El sistema despliega la pantalla "Lista de responsables".
3. Presiona el botón "Alta".	4. El sistema despliega la pantalla "Alta y Modificación de responsable".
5. Llena los campos mostrados en la forma: Nombre, cargo, Teléfono, Fax y E-mail.	
6. Presiona el botón "Guardar".	7. El sistema da de alta al nuevo responsable.
	8. El sistema despliega la pantalla "Lista de responsables".
9. Elige de la lista al responsable que se le asignará al alumno.	
10. Presiona el botón "Asignar responsable".	11. El sistema asocia el responsable al registro del alumno.
	12. El sistema despliega la pantalla "Prealta" con el nuevo responsable.
Cursos Alternos	
Línea 3: Presiona el botón "Regresar".	
Línea 4: Despliega la pantalla de "Prealta" sin modificar el responsable asociado al alumno.	
Línea 3: Presiona el botón "Asignar Responsable".	
Línea 4: Asocia el responsable al registro del alumno. Despliega la pantalla "Prealta" con el nuevo responsable.	
Línea 6: Presiona el botón "Regresar".	
Línea 7: Despliega la pantalla "Lista de responsables" sin cambio en la lista.	

Clave: SASS19CL.

Caso de uso: Descargar Datos para la Carta de Liberación.

Casos de uso incluidos: Autenticar Usuario.

Actores: Jefe de SS de la FI

Descripción: Permite obtener la información sobre la carta de liberación de un alumno que ha concluido su SS.

Curso Normal de los Eventos	
Acción del Actor	Respuesta del Sistema
1. Elige el módulo de "Carta de Liberación".	2. Despliega una lista con los nombres de las Divisiones de Ingeniería de la Facultad.
3. Elige la División de Ingeniería de su interés.	4. Despliega la lista de los alumnos susceptibles al proceso de carta liberación.
5. Selecciona los registros de los alumnos que desea obtener la información.	
6. Presiona el botón "Obtener Datos para carta".	7. Obtiene la información de los alumnos seleccionados.
	8. Descarga en la máquina en la que se está consultando el archivo de Excel con la información referente a la carta de los alumnos.
9. Guarda el archivo en la ruta que necesite.	
10. Puede seguir trabajando con cualquiera de las opciones del menú principal, incluso salir del sistema	

Clave: SASS20MC.

Caso de uso: Dar mantenimiento de Cuentas de Coordinadores de SS.

Casos de uso incluidos: Autenticar Usuario.

Actores: Jefe de SS de la FI

Descripción: Permite dar de alta, baja y modificación de cuentas de usuario con perfil de Jefe de SS y Coordinador de SS de División.

Curso Normal de los Eventos	
Acción del Actor	Respuesta del Sistema
1. Elige el módulo de "Mto Coordinadores".	2. Despliega la pantalla de "Mto Coordinador-Lista" con la lista de los coordinadores y jefes de SS dados de alta.
3. Presiona el botón "Alta"	4. Despliega la pantalla "Mto Coordinador-Alta".
5. Llena los campos mostrados en la forma: Role del usuario, División, Clave, Nombre(s), Apellido Paterno,	

Apellido Materno, Contraseña, Confirma Contraseña.	
6. Presiona el botón "Guardar".	7. Guarda la información del nuevo usuario.
	8. Despliega la pantalla de "Mto Coordinador-Lista" con la lista de los coordinadores y jefes de SS dados de alta.
9. Puede seguir trabajando con cualquiera de las opciones del menú principal, incluso salir del sistema	
Cursos Alternos	
Línea 2: Presenta la pantalla "Mto Coordinador" sin usuarios dados de alta.	
Línea 3: Presiona el botón "Dar de alta"	
Línea 4: Presenta la pantalla "Mto Coordinador-Alta"	

Clave: SASS21GA.

Caso de uso: Generar Archivos con los Programas de SS Correspondientes a la Facultad de Ingeniería.

Casos de uso incluidos: Ninguno.

Actores: Jefe de SS de la FI

Descripción: Con base en la inscripción de nuevos programas de SS correspondientes a la Facultad de Ingeniería en el Sistema de Access, se obtendrá mediante macros en Access los archivos correspondientes a los programas, dependencias, instituciones, responsables, coordinadores, relación de programas con carrera, y áreas.

Curso Normal de los Eventos	
Acción del Actor	Respuesta del Sistema
1. Acceder a la Base de Datos de Access del Sistema.	2. Despliega el entorno de Access.
3. En la ventana "Base de Datos seleccionar la opción formularios"	
4. Dar doble clic sobre el formulario "frmAnoQry"	5. Se presenta la pantalla Exportación de Programas
6. Llenar los campos de año inicial y año final que determinan el rango de años en los que se ha dado de alta programas.	
7. Presionar el botón "Exportar".	8. Exporta la información relacionada con: programas, dependencias, instituciones, responsables, coordinadores, relación de programas con carrera y áreas en los archivos: programa.txt, dependencia.txt, institución.txt, responsable.txt, coordinador.txt, carrera_prg.txt y area.txt.

Clave: SASS22CA.

Caso de uso: Cargar los Archivos Generados sobre los Programas de SS en el Sistema de Administración de SS

Casos de uso incluidos: Generar archivos con los programas de SS correspondientes a la Facultad de Ingeniería.

Actores: Jefe de SS de la FI

Descripción: Cargar en la base de datos de Sybase los archivos generados por el Sistema de Access.

Curso Normal de los Eventos	
Acción del Actor	Respuesta del Sistema
1. Conectarse al Servidor de Base de Datos Sybase, mediante una conexión vía FTP con el usuario "serv_soc".	
2. Subir los archivos: programa.txt, dependencia.txt, institución.txt, responsable.txt, coordinador.txt, carrera_prg.txt y area.txt en modo texto en la carpeta "ss_arch_carga".	
3. Desconectarse del Servidor de Base de Datos Sybase.	
4. Conectarse al Servidor de Base de Datos Sybase, mediante una conexión vía Telnet con el usuario "serv_soc".	
5. Ejecutar el shell: ./ss_carga.sh	6. Presenta en pantalla información sobre los registros copiados.
7. Ejecutar el shell: ./ss_actualiza.sh	8. Presenta en pantalla las tablas que se actualizaron.

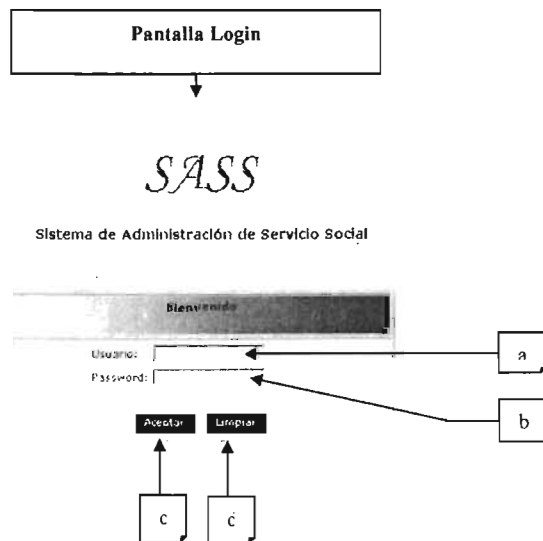
4.2.2. Secuencia de pantallas

4.2.2.1. Flujo de pantallas comunes para los diferentes usuarios (Alumno, Coordinador de SS de la División de Ingeniería, Jefe de SS de la FI)

En el sistema existen características y funcionalidad que puede ser aplicable a los diferentes grupos de usuarios del sistema y que en consecuencia siempre se encuentran disponibles, por esta razón, en esta sección se describen las pantallas que constituyen dichas características y funcionalidad genérica.

Entrada al Sistema

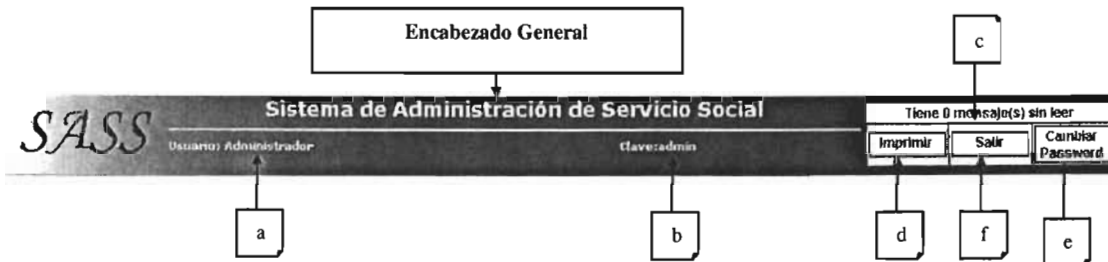
Pantalla principal para el acceso al sistema.



- a. El usuario deberá proporcionar su clave de acceso al sistema
- b. El usuario deberá proporcionar su contraseña asociada a su clave.
- c. Permite aceptar o denegar el acceso al sistema.
- d. Permite limpiar el contenido de los campos de usuario y contraseña.

Encabezado general del Sistema

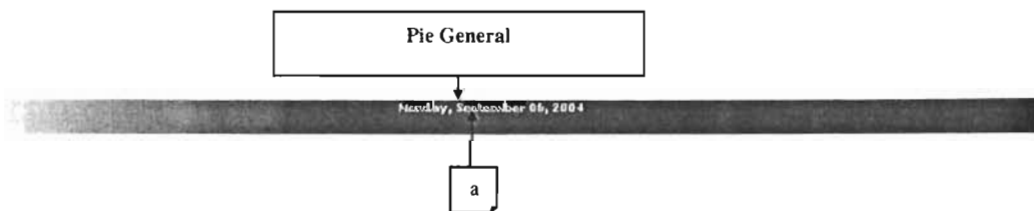
Encabezado común para los diferentes perfiles de usuario que se firman en el sistema.



- a. Nombre del usuario que accede al sistema.
- b. Clave del usuario que accede al sistema.
- c. Permite acceder a la funcionalidad para consultar los mensajes del usuario firmado.
- d. Permite imprimir la información presentada en pantalla.
- e. Permite acceder a la funcionalidad para cambiar la contraseña para acceder al sistema.
- f. Permite salir del sistema.

Pie general del Sistema

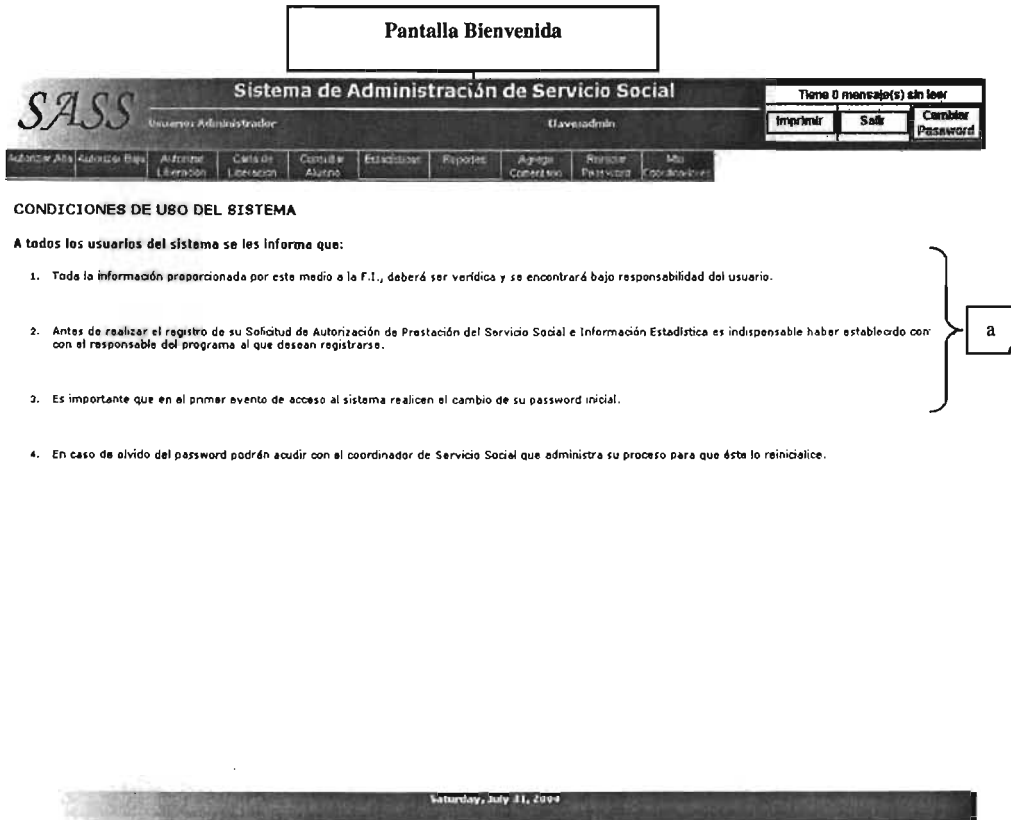
Pie común para los diferentes perfiles de usuario que se firman en el sistema.



- a. Fecha y hora del día en curso.

Pantalla de Bienvenida

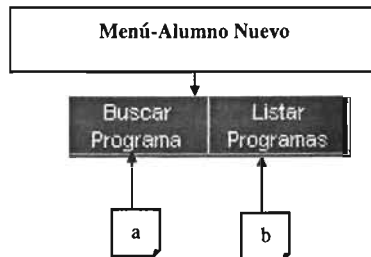
Información sobre los términos y condiciones que tiene que considerar el usuario al utilizar el sistema, el cual será diferente para cada tipo de perfil de usuario.



- a. Términos y condiciones para el uso del sistema.

Menú para el perfil de Alumno Nuevo

Menú que se presenta cuando un usuario tiene el perfil de alumno y no tiene dado de alta un registro de SS o estuvo dado de baja.

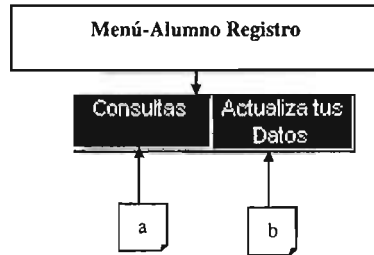


- a. Permite acceder a la funcionalidad para hacer búsquedas por clave de programa de SS al que se desea inscribir.

- b. Permite acceder a la funcionalidad para listar los programas de SS a los que se puede inscribir el alumno.

Menú para el perfil de Alumno con Registro

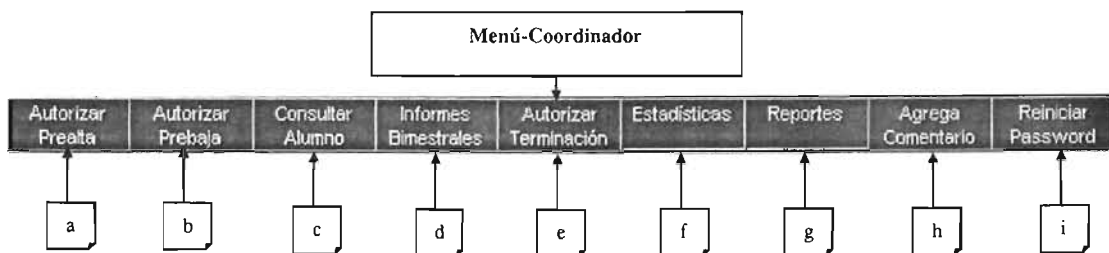
Menú que se presenta cuando un usuario tiene el perfil de alumno y tiene dado de alta un registro de SS.



- a. Permite acceder a la funcionalidad para realizar la consulta de los movimientos del alumno
- b. Permite acceder a la funcionalidad para realizar la actualización de los datos personales del alumno.

Menú para el perfil de Coordinador de SS de la División de Ingeniería

Menú que se presenta cuando un usuario tiene el perfil de coordinador de SS.

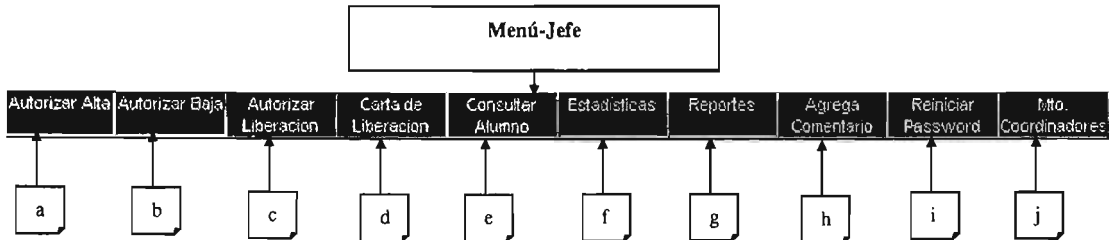


- a. Permite acceder a la funcionalidad para hacer la autorización de prealtas de SS de alumnos correspondientes a la división a la que pertenece el coordinador.
- b. Permite acceder a la funcionalidad para hacer la autorización de prebajas de SS de alumnos correspondientes a la división a la que pertenece el coordinador.
- c. Permite acceder a la funcionalidad para hacer la consulta de movimientos de alumnos correspondientes a la división a la que pertenece el coordinador.
- d. Permite acceder a la funcionalidad para hacer la actualización de informes bimestrales de alumnos correspondientes a la división a la que pertenece el coordinador.
- e. Permite acceder a la funcionalidad para hacer la autorización de terminaciones de alumnos correspondientes a la división a la que pertenece el coordinador.
- f. Permite acceder a la funcionalidad para obtener información estadística relacionada con los movimientos de los alumnos durante el ejercicio de su servicio social.
- g. Permite acceder a la funcionalidad para obtener información detallada relacionada con los movimientos de los alumnos durante el ejercicio de su servicio social.
- h. Permite acceder a la funcionalidad para agregar comentarios a los alumnos correspondientes a la división a la que pertenece el coordinador.

- i. Permite acceder a la funcionalidad para renovar las contraseñas de los alumnos correspondientes a la división a la que pertenece el coordinador.

Menú para el perfil de Jefe de SS

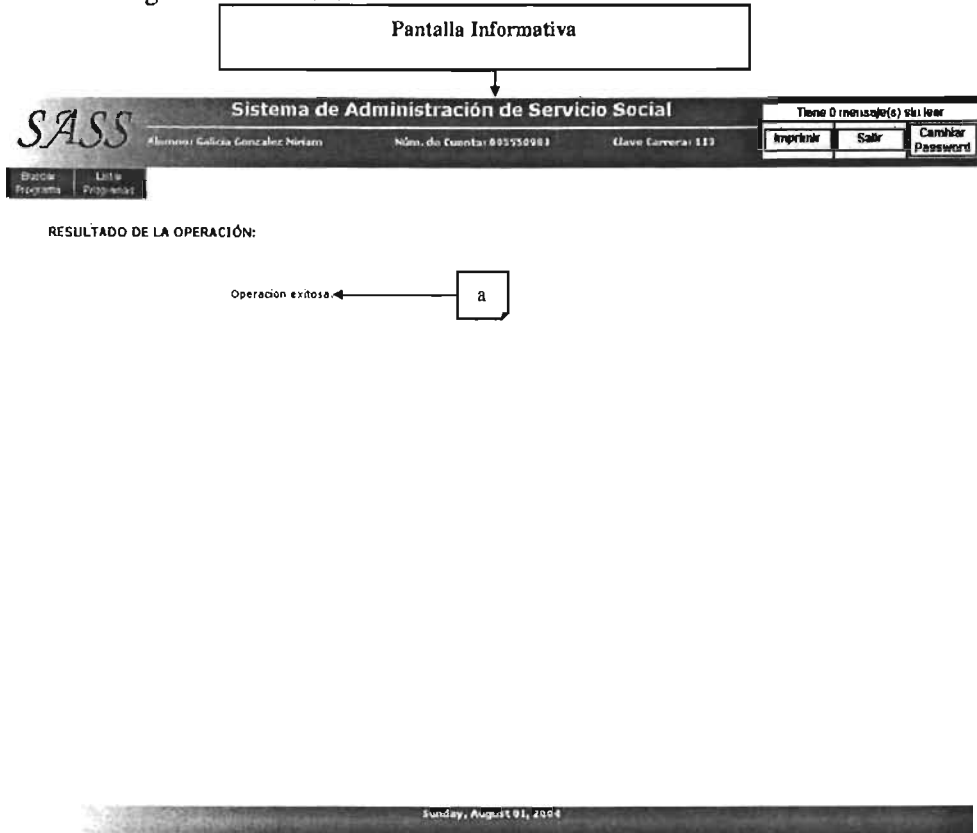
Menú que se presenta cuando un usuario tiene el perfil de coordinador de SS.



- a. Permite acceder a la funcionalidad para hacer la autorización de altas de SS de alumnos de todas las divisiones.
- b. Permite acceder a la funcionalidad para hacer la autorización de bajas de SS de alumnos de todas las divisiones.
- c. Permite acceder a la funcionalidad para hacer la autorización de liberaciones de alumnos de todas las divisiones.
- d. Permite acceder a la funcionalidad para obtener la información de los alumnos a los que se les imprimirá su carta de liberación de SS.
- e. Permite acceder a la funcionalidad para hacer la consulta de movimientos de alumnos de todas las divisiones.
- f. Permite acceder a la funcionalidad para obtener información estadística relacionada con los movimientos de los alumnos durante el ejercicio de su servicio social.
- g. Permite acceder a la funcionalidad para obtener información detallada relacionada con los movimientos de los alumnos durante el ejercicio de su servicio social.
- h. Permite acceder a la funcionalidad para agregar comentarios a los alumnos de todas las divisiones.
- i. Permite acceder a la funcionalidad para reiniciar la contraseña de los usuarios del sistema.
- j. Permite acceder a la funcionalidad para dar de alta y baja a usuarios con perfil de coordinador y jefe de SS.

Pantalla genérica de Error o Confirmación

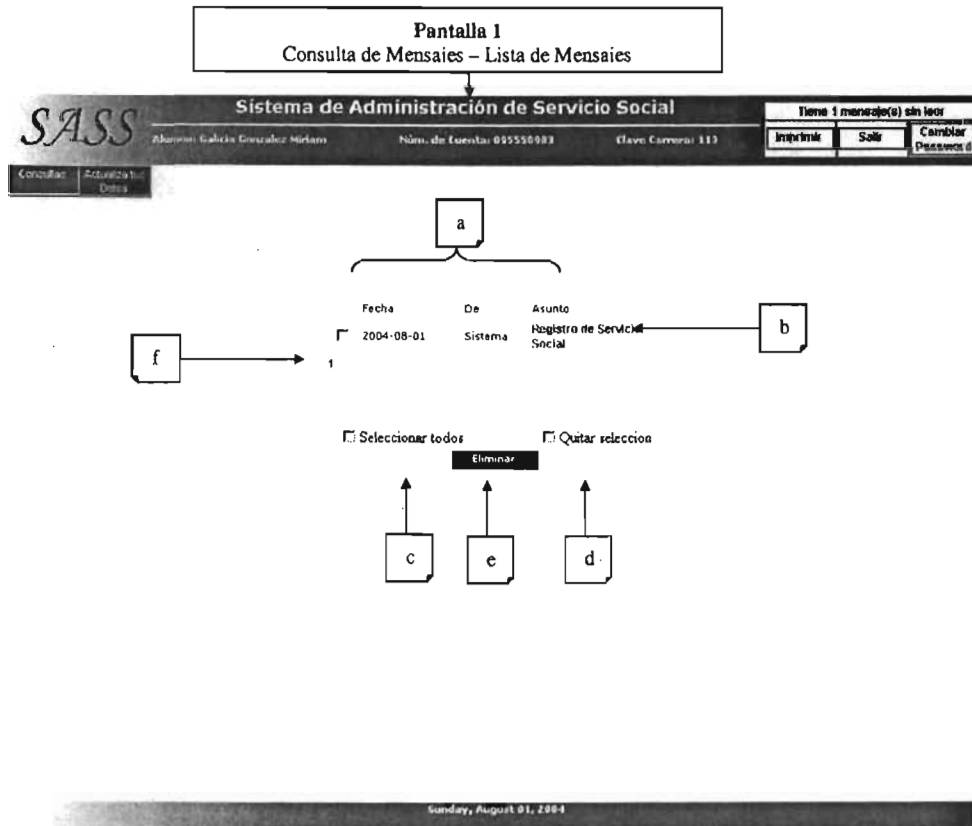
Pantalla que se presentará para toda operación en el sistema que necesite notificar error o confirmación en alguna transacción.



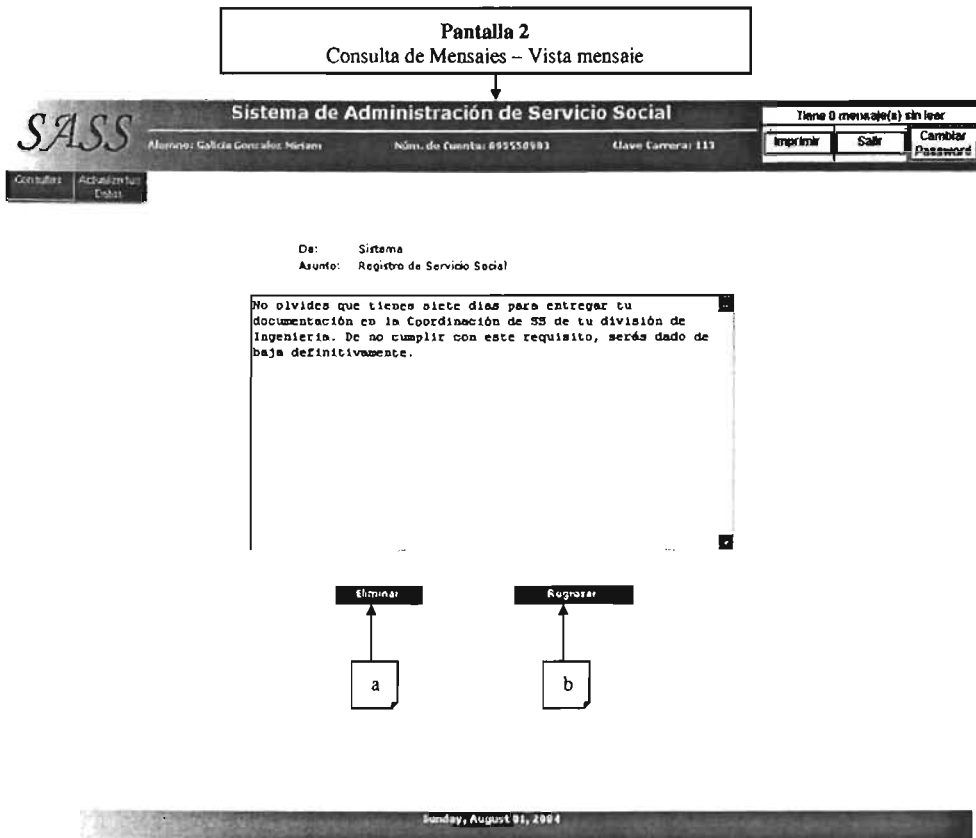
- a. Mensaje de error o confirmación.

Consulta de Mensajes

Al módulo de consulta de mensajes se accede a través de la liga “Usted tiene N mensajes sin leer” del menú principal.



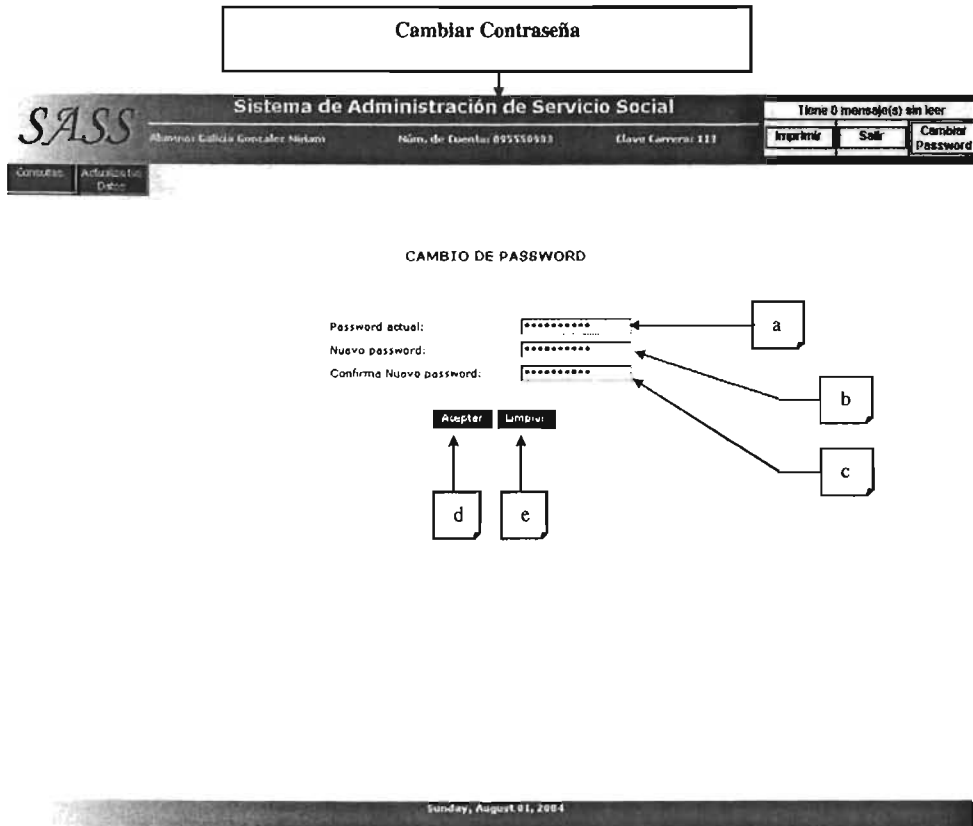
- Se le muestra al usuario información general sobre el mensaje como: fecha de envío, quién envía y el asunto.
- Remite a la pantalla de Vista Mensaje, pantalla 2.
- Selecciona todos los mensajes que aparecen en la lista.
- Quita la selección de todos los mensajes seleccionados que aparecen en la lista.
- Elimina los mensajes seleccionados, regresa a la pantalla 1.
- Numero de pantallas con información, regresa a la pantalla 1 con los registros del número de pantalla seleccionada.



- a. Elimina el mensaje en vista, regresa a Pantalla 1.
- b. Remite a la pantalla de Lista de Mensajes, Pantalla 1.

Cambiar Contraseña

Al módulo de cambio de contraseña se accede a través del botón “Cambiar Contraseña” del menú principal.

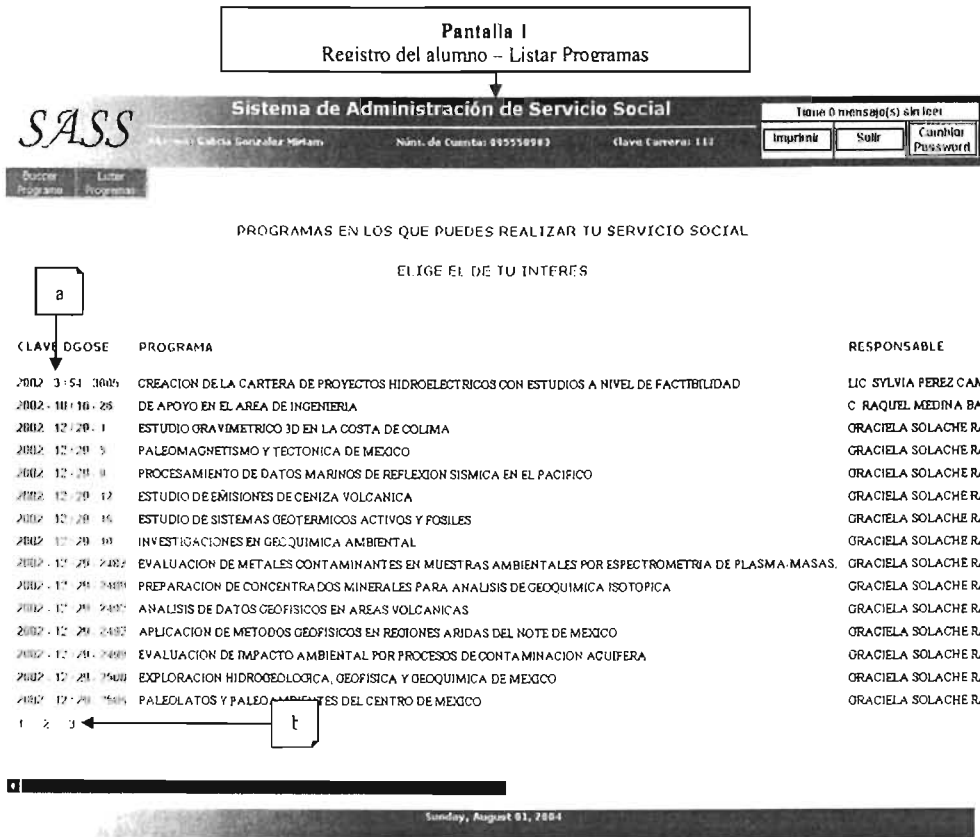


- a. Contraseña actual del usuario.
- b. Nueva contraseña del usuario.
- c. Confirmación de la nueva contraseña del usuario.
- d. Efectúa el cambio de contraseña del usuario firmado, envía la pantalla general: pantalla informativa.
- e. Limpia las cajas de texto de los campos de contraseña.

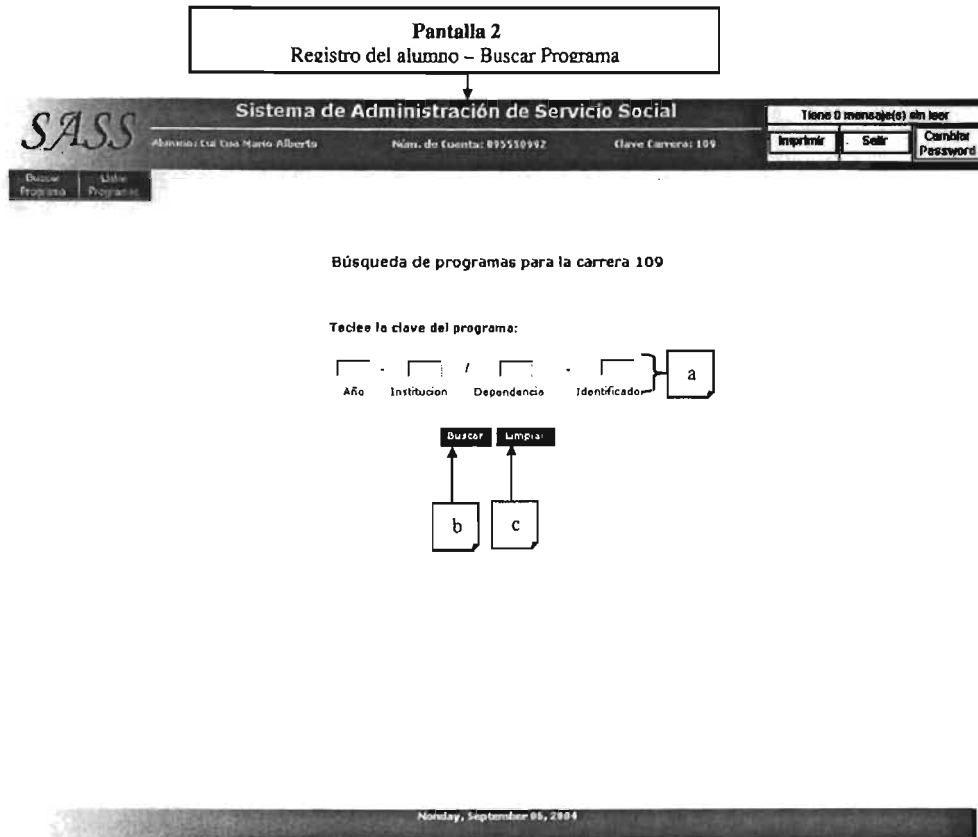
4.2.2.2. Flujo de pantallas para el Alumno

Registro del alumno

Para permitir al alumno realizar su registro de servicio social, deberá elegir un programa de servicio social mediante las opciones “**Buscar Programa**” y “**Listar Programas**” del menú para el perfil de un alumno nuevo.



- a. Remite a la pantalla 3, llena una solicitud de registro con la información relacionada al programa seleccionado.
- b. Número de pantallas con información, regresa a la pantalla 1 con los registros de la pantalla seleccionada.



- a. Información que compone la clave del programa: Año, Clave de la Institución, Clave de la dependencia, Identificador.
- b. Remite a la pantalla 3, llena una solicitud de registro con la información relacionada al programa buscado.
- c. Limpia las cajas de texto de los campos de la clave del programa.

Pantalla 3
Registro del alumno - Datos del alumno

SASS

Sistema de Administración de Servicio Social
Alumnos: Cecilia González Herrera Núm. de Cuenta: 895550988 Clave Carrera: 111

Tiene 0 mensajes(s) sin leer
Imprimir Salir Cambiar Password

Buscar Regresar
Lista Programas

SOLICITUD DE AUTORIZACIÓN DE PRESTACIÓN DEL SERVICIO SOCIAL E INFORMACIÓN ESTADÍSTICA

DATOS DEL ALUMNO

Número de cuenta: 0	Clave de la carrera: 111	Ingreso a la F.I.: 2001	
Nombre: [a] [b] [c] [d] [e] [f] [g] [h] [i] [j] [k] [l]	Oto: [c]	Mínim Nombre(s): [m]	
Edad: [a]	Ap. Pat: [b]	Ap. Mat: [g]	Sexo: [k]
Calle: [e]	Número: [c]	Colonia: [d]	
C.P.: [e]	Estado: [f]	Del. o Mpo.: [g]	
Tel. particular: [h]	Tel. celular: [i]	Localizador: [j]	PIN: [k]
Créditos aprobados: 70		Avance: 70.00	Promedio: 8.00
Fecha de inicio: [m]	Duración en Meses: [n]	Horas/Semana: [o]	Percepción mensual: [q]
Forma de remuneración: [p]			

DATOS DE LA DEPENDENCIA

[j] Sunday, 17/02/2008

- a. Edad del alumno.
- b. Calle del domicilio del alumno.
- c. Número del domicilio del alumno.
- d. Colonia del domicilio del alumno.
- e. Código postal del domicilio del alumno.
- f. Estado del domicilio del alumno.
- g. Delegación o Municipio del domicilio del alumno.
- h. Teléfono particular del domicilio del alumno.
- i. Teléfono celular del alumno.
- j. Número de localizador del alumno.
- k. Pin del localizador del alumno.
- l. E-mail del alumno.
- m. Fecha tentativa de inicio de SS del alumno.
- n. Duración en meses del SS del alumno.
- o. Horas a la semana que se trabajarán en el SS.
- p. Tipo de remuneración que se recibirá durante el ejercicio del SS.
- q. Monto de la remuneración.

Pantalla 3
Registro del alumno – Datos de la Dependencia

Sistema de Administración de Servicio Social

Alumnos: Galicia Gonzalez Marian | Núm. de Cuenta: 895550983 | Clave Carrera: 113

Tiene 0 mensaj(e)s sin leer

Imprimir
Salir
Cambiar Password

DATOS DE LA DEPENDENCIA

Nombre de la Dependencia	UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO		
Subdirección o Departamento	INSTITUTO DE GEOFISICA		
Oficina o Sección	DEPTO. DE GEOMAGNETISMO		
Dirección	CIRCUITO EXTERIOR S/N	Colonia	CD. UNIVERSITARIA
C.P.	04510	Del o Mpo	COYOACAN
Estado	D.F.	Teléfono(s)	56224128

DATOS DEL PROGRAMA

Nombre del Programa	ESTUDIO GRAVIMETRICO 3D EN LA COSTA DE COLIMA		
Clave DOOSE	2002 - 12 / 29 - 1		
Responsable del Programa	GRACIELA SOLACHE RAMIREZ		
Jefe Inmediato	DR. WILLIAM LEE BANDY		
Cargo	INVESTIGADOR TITULAR "A" DEFINITIVO	Teléfono(s)	56224128
e-mail	bandy@tonatiuh.ige.ofcu.unam.mx	Tipo de Programa	Investigacion

Solicito se me autorice cumplir con la prestación de Servicio Social en el programa mencionado.

Sunday, August 01, 2011

a

- a. Información general de la dependencia a la que pertenece el programa de SS.

Pantalla 3
Registro del alumno – Datos del Programa

SASS

Sistema de Administración de Servicio Social

Almendra Caldeza González Miram Núm. de Cuenta: 895518887 Clave Externa: 113

Tiene 0 mensaj(es) sin leer

Imprimir
Salir
Cambiar Password

Borrar Programa
Listar Programa

Nombre de la Dependencia UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO
 Subdirección o Departamento INSTITUTO DE GEOFISICA
 Oficina o Sección DEPTO. DE GEOMAGNETISMO
 Dirección CIRCUITO EXTERIOR S/N Colonia CD. UNIVERSITARIA
 C.P. 04510 Del o Mpo COYOACAN
 Estado D.F. Teléfono(s) 56224128

DATOS DEL PROGRAMA

Nombre del Programa ESTUDIO GRAVIMETRICO 3D EN LA COSTA DE COLIMA
 Clave DOGSE 2002 - 12 / 29 - 1
 Responsable del Programa GRACIELA SOLACHE RAMIREZ
 Jefe Inmediato DR. WILLIAM LEE BANDY
 Cargo INVESTIGADOR TITULAR 'A' DEFINITIVO Teléfono(s) 56224128
 e-mail bandy@tonatuh.igecofcu.unam.mx Tipo de Programa Investigación

Solicito se me b de cumplir con la prestación de Servicio Social en el programa mencionado.

México D.F. a 01/08/2004

b
Guardar Registro

c
Cancelar

a

Sunday, August 01, 2004

- a. Información general del programa de SS elegido.
- b. Guarda la información del registro del alumno, remite a la pantalla Informativa.
- c. Regresa a la pantalla 1 o a la pantalla 2 dependiendo quien la hay llamado.

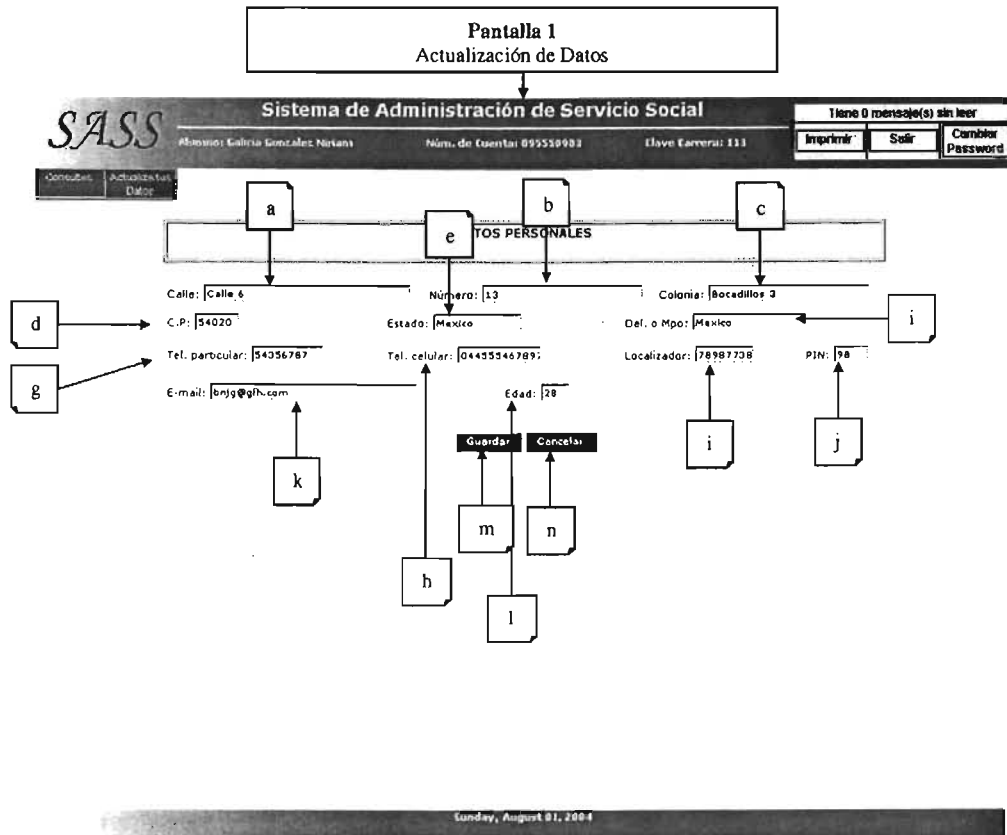
Consulta de Movimientos del Alumno

Para permitir al alumno consultar sus movimientos durante el ejercicio de su SS, deberá elegir la opción “Consultas” del menú para el perfil de un alumno con Registro.

Pantalla 1
Consulta de Movimientos

Actualización de Datos del Alumno

Para permitir al alumno actualizar sus datos personales deberá elegir la opción “Actualiza tus datos” del menú para el perfil de un alumno con registro.

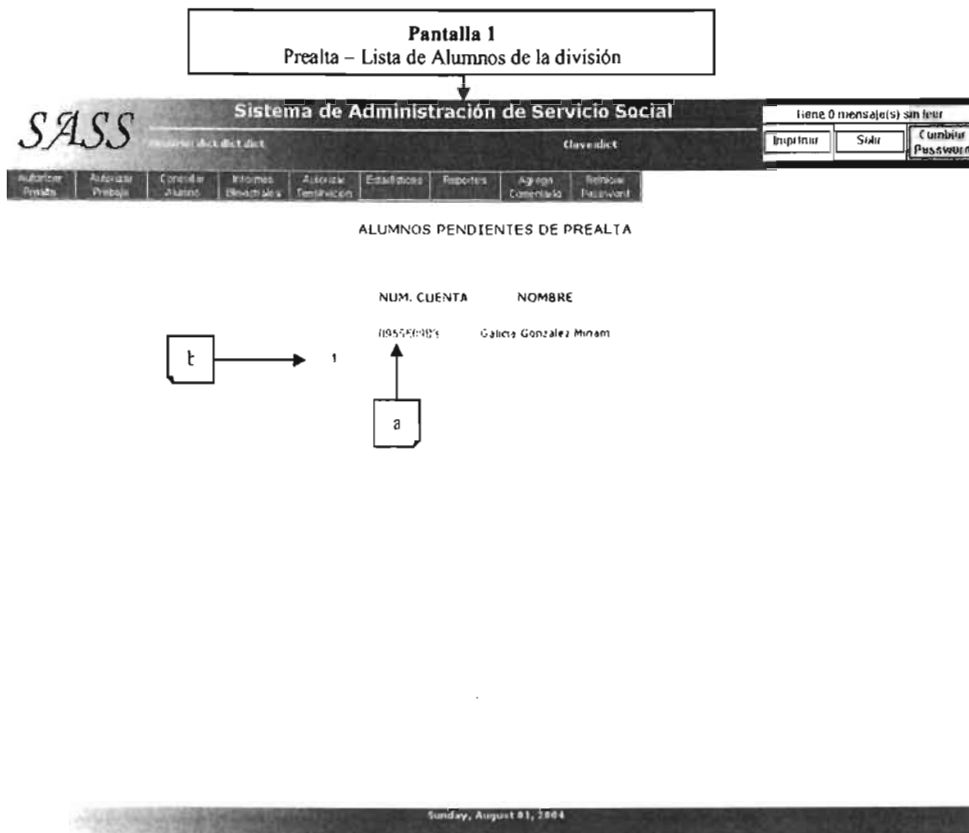


- a. Calle del domicilio del alumno.
- b. Número del domicilio del alumno.
- c. Colonia del domicilio del alumno.
- d. Código postal del domicilio del alumno.
- e. Estado del domicilio del alumno.
- f. Delegación o Municipio del domicilio del alumno.
- g. Teléfono particular del domicilio del alumno.
- h. Teléfono celular del alumno.
- i. Número de localizador del alumno.
- j. Pin del localizador del alumno.
- k. E-mail del alumno.
- l. Edad del alumno.
- m. Guarda los datos personales del alumno, remite a la pantalla informativa.
- n. Regresa a la pantalla de Bienvenida sin hacer cambios en la información personal del alumno.

4.2.2.3. Flujo de Pantallas para el Coordinador de SS de la División de Ingeniería.

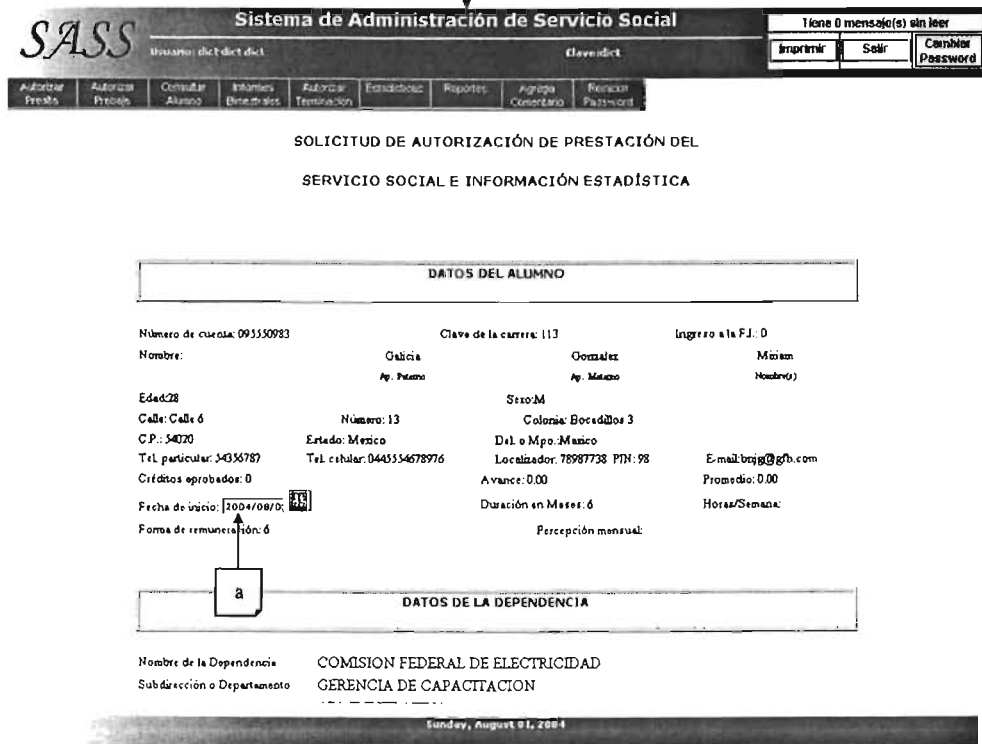
Prealta del Alumno

Para autorizar la prealta a un programa de SS a un alumno, deberá elegir la opción “**Autorizar Prealta**” del menú para el perfil de coordinador de SS



- a. Remite a la pantalla 2, obtiene la información del registro del alumno para su revisión.
- b. Número de pantallas con información, regresa a la pantalla 1 con los registros del número de pantalla seleccionada.

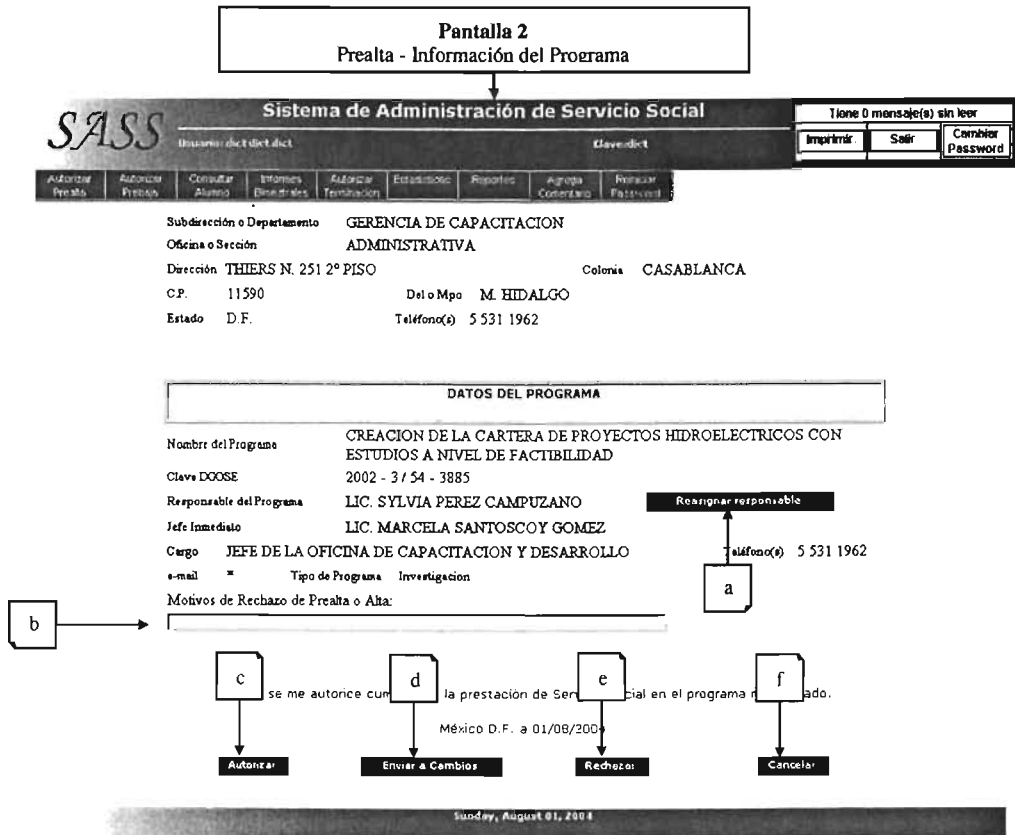
Pantalla 2
Prealta - Información del Alumno



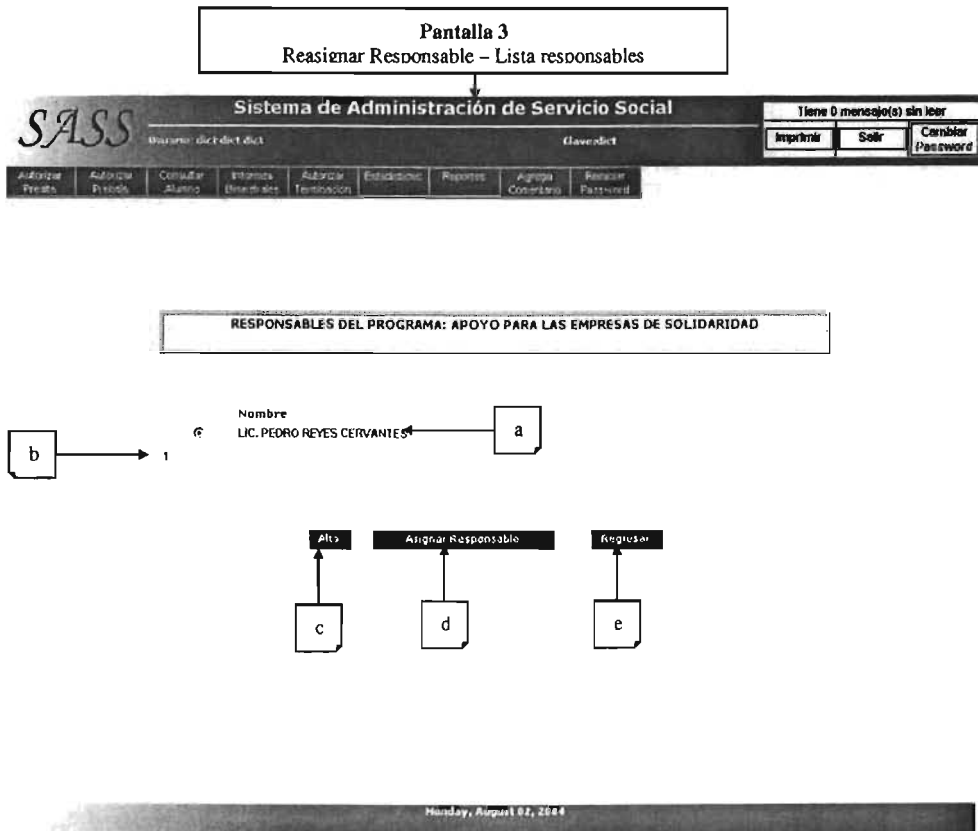
- a. Fecha inicial de SS, la cual podrá ser modificada en caso de que sea incorrecta.

Pantalla 2
Prealta - Información de la Dependencia

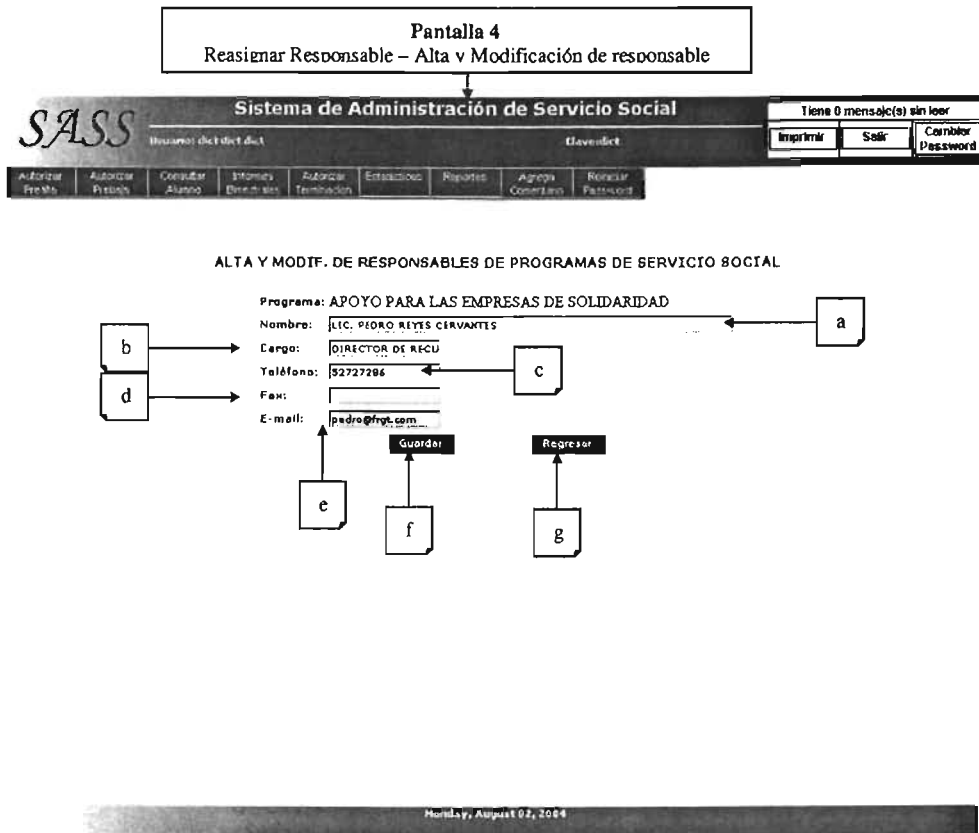
SASS										Tiene 0 mensaje(s) sin leer		
Sistema de Administración de Servicio Social										<input type="button" value="Imprimir"/> <input type="button" value="Salir"/> <input type="button" value="Cambiar Password"/>		
Usuario: dicit dicit Clave: dicit												
Autorizar Prealta	Autorizar Prealta	Consultar Alumnos	Informes Base de datos	Avanzar Terminación	Estadísticas	Reportes	Agregar Comentario	Eliminar Password				
DATOS DE LA DEPENDENCIA												
Nombre de la Dependencia		COMISION FEDERAL DE ELECTRICIDAD										
Subdirección o Departamento		GERENCIA DE CAPACITACION										
Oficina o Sección		ADMINISTRATIVA										
Dirección		THERS N. 251 2º PISO					Colonia		CASABLANCA			
C.P.		11590		Del o Mpo		M. HIDALGO						
Estado		D.F.		Teléfono(s)		5 531 1962						
DATOS DEL PROGRAMA												
Nombre del Programa		CREACION DE LA CARTERA DE PROYECTOS HIDROELECTRICOS CON ESTUDIOS A NIVEL DE FACTIBILIDAD										
Clave DOOSE		2002 - 3 / 54 - 3885										
Responsable del Programa		LIC. SYLVIA PEREZ CAMPUZANO					<input type="button" value="Reasignar responsable"/>					
Jefe Inmediato		LIC. MARCELA SANTOSCOY GOMEZ										
Cargo		JEFE DE LA OFICINA DE CAPACITACION Y DESARROLLO					Teléfono(s) 5 531 1962					
e-mail		=		Tipo de Programa		Investigación						
Motivos de Rechazo de Prealta o Alta:		<div style="border: 1px solid black; height: 20px; width: 100%;"></div>										
Sunday, August 01, 2004												



- a. Remite a la Pantalla 3, se accede a la funcionalidad de reasignación de responsable a un alumno para el ejercicio de su SS.
- b. Motivo por el cual se rechazará el trámite de prealta del alumno.
- c. Autoriza la prealta del alumno en el programa de SS, remite a la pantalla informativa.
- d. Le envía un mensaje al alumno indicándole que debe modificar los datos de su registro, remite a la pantalla informativa.
- e. Rechaza el movimiento del alumno, remite a la pantalla informativa.
- f. Remite a la pantalla 1.



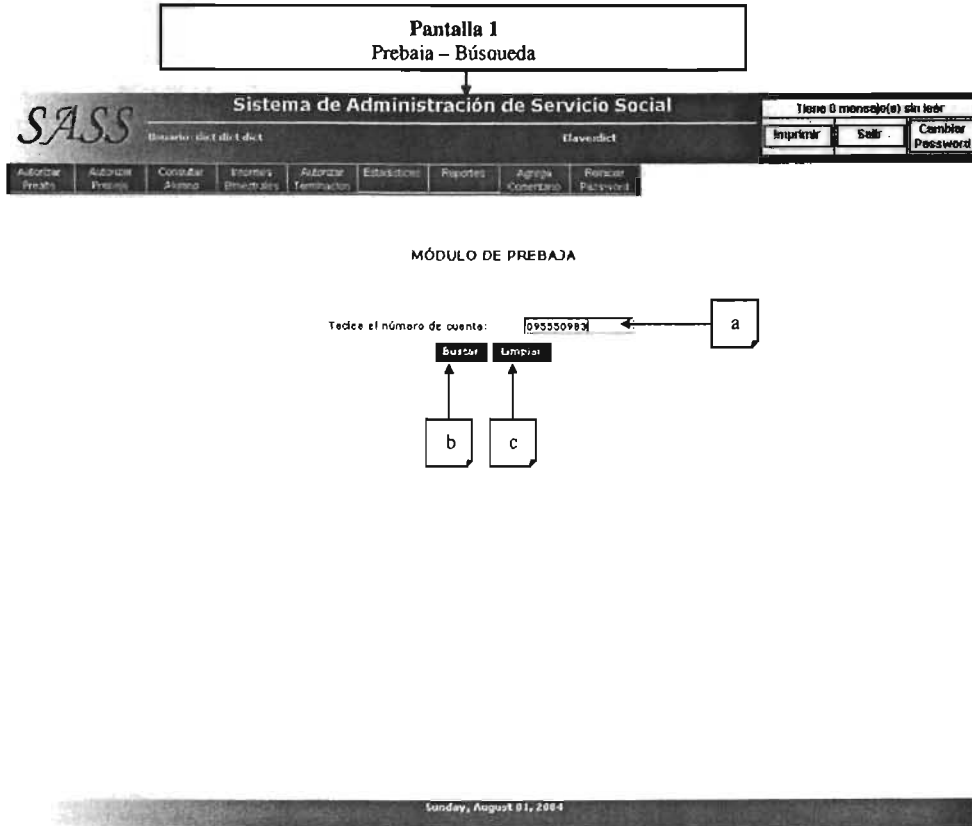
- a. Remite a la pantalla de alta y modificación de datos del responsable, pantalla 4.
- b. Número de pantallas con información, regresa a la pantalla 3 con los registros del número de pantalla seleccionada.
- c. Remite a la pantalla de alta y modificación de datos del responsable, pantalla 4.
- d. Remite a la pantalla 2 asignándole al alumno el responsable elegido.
- e. Remite a la pantalla 2, sin realizar cambios en la solicitud.



- a. Nombre del Responsable.
- b. Cargo del responsable.
- c. Teléfono de la oficina.
- d. Fax.
- e. Email.
- f. Actualiza la información del responsable, remite a la pantalla 3.
- g. Remite a la pantalla 3, sin realizar cambios a la información del responsable.

Prebaja del Alumno

Para autorizar la prebaja a un programa de SS a un alumno, deberá elegir la opción “Autorizar Prebaja” del menú para el perfil de coordinador de SS.



- a. Número de cuenta del alumno.
- b. Obtiene la información del registro del alumno en un programa de SS, remite a la pantalla 2.
- c. Limpia la caja de texto del número de cuenta del alumno.

Pantalla 2
Prebaia - Información del Alumno

SASS

Sistema de Administración de Servicio Social
Usuario: oled dict dict E:lavendict

Tiene 0 mensaje(s) sin leer

Autoriza Prebaia
Autoriza Prebaia
Consulta Alumno
Informes Estadísticos
Autorizar Terminación
Estadísticas
Reportes
Agrega Constante
Reinicia Password

SOLICITUD DE AUTORIZACIÓN DE PRESTACIÓN DEL SERVICIO SOCIAL E INFORMACIÓN ESTADÍSTICA

DATOS DEL ALUMNO

Número de cuenta: 095550983		Clave de la carrera: 113		Ingreso a la F.I.: 0	
Nombre:		Océlica	González	Mizam	
	<small>Ap. Paterno</small>		<small>Ap. Materno</small>	<small>Nombre(s)</small>	
Edad: 28		Sexo: M			
Calle: Calle 6		Número: 13	Colonia: Bocadillos 3		
C.P.: 54020		Estado: Mexico		Del. o Mpo.: Mexico	
Tel. particular: 54356787		Tel. celular: 0445554678976		Localizador: 78987738 PIN: 98	
Créditos aprobados: 0		Avance: 0.00		Promedio: 0.00	
Fecha de inicio: 2004/08/01		Duración en Meses: 6		Horas/Semana:	
Forma de remuneración: 6		Percepción mensual			

DATOS DE LA DEPENDENCIA

Nombre de la Dependencia	COMISION FEDERAL DE ELECTRICIDAD
Subdirección o Departamento	GERENCIA DE CAPACITACION

Sunday, August 01, 2004

Pantalla 2
Prebaia - Información de la Dependencia

SASS

Sistema de Administración de Servicio Social

Usuario: Clave:

Tiene 0 mensajes en leer

Imprimir	Salir	Cambiar Password
----------	-------	------------------

[Ayuda](#) [Perfiles](#) [Autorizar Perfiles](#) [Consultar Alumno](#) [Informes Estadísticos](#) [Actualizar Terminación](#) [Estadísticas](#) [Reportes](#) [Agregar Contacto](#) [Eliminar Perfil](#)

DATOS DE LA DEPENDENCIA

Nombre de la Dependencia	COMISION FEDERAL DE ELECTRICIDAD		
Subdirección o Departamento	GERENCIA DE CAPACITACION		
Oficina o Sección	ADMINISTRATIVA		
Dirección	THIERS N. 251 2º PISO	Colonia	CASABLANCA
C.P.	11590	Del o Mpo	M. HIDALGO
Estado	D.F.	Teléfono(s)	5 531 1962

DATOS DEL PROGRAMA

Nombre del Programa	CREACION DE LA CARTERA DE PROYECTOS HIDROELECTRICOS CON ESTUDIOS A NIVEL DE FACTIBILIDAD		
Clave DOOSE	2002 - 3 / 54 - 3885		
Responsable del Programa	LIC. SYLVIA PEREZ CAMPUZANO		
Jefe Inmediato	LIC. MARCELA SANTOSCOY GOMEZ		
Cargo	JEFE DE LA OFICINA DE CAPACITACION Y DESARROLLO	Teléfono(s)	5 531 1962
e-mail	* Tipo de Programa	Investigacion	
Motivos de Rechazo de Prebaia o Otra:			

Sunday, August 01, 2004

Pantalla 2
Prebaja – Datos del Programa

SASS

Sistema de Administración de Servicio Social

Numero de Usuario: clave:dict clave:dict

Tiene 0 mensaje(s) sin leer

Imprimir Salir Cambiar Password

Autorizar Prebaja Autorizar Prebaja Consultar Alumno Informes (Estructuras)

Autorizar Terminación Estadísticas Reportes Agregar Contratos Renovar Pasaporte

Subdirección o Departamento: **GERENCIA DE CAPACITACION**

Oficina o Sección: **ADMINISTRATIVA**

Dirección: **TIERS N. 251 2º PISO** Colonia: **CASABLANCA**

C.P.: **11590** Del o Mpo: **M. HIDALGO**

Estado: **D.F.** Teléfono(s): **5 531 1962**

DATOS DEL PROGRAMA

Nombre del Programa: **CREACION DE LA CARTERA DE PROYECTOS HIDROELECTRICOS CON ESTUDIOS A NIVEL DE FACTIBILIDAD**

Clave DOOSE: **2002 - 3 / 54 - 3885**

Responsable del Programa: **LIC. SYLVIA PEREZ CAMPUZANO**

Jefe Inmediato: **LIC. MARCELA SANTOSCOY GOMEZ**

Cargo: **JEFE DE LA OFICINA DE CAPACITACION Y DESARROLLO** Teléfono(s): **5 531 1962**

e-mail: Tipo de Programa: **Investigación**

Motivos de Rechazo de Prebaja o Alta:

a →

b se me autorice cumplir con la prestación de Servicio Social en el programa cado.

México D.F. a 01/08/2004

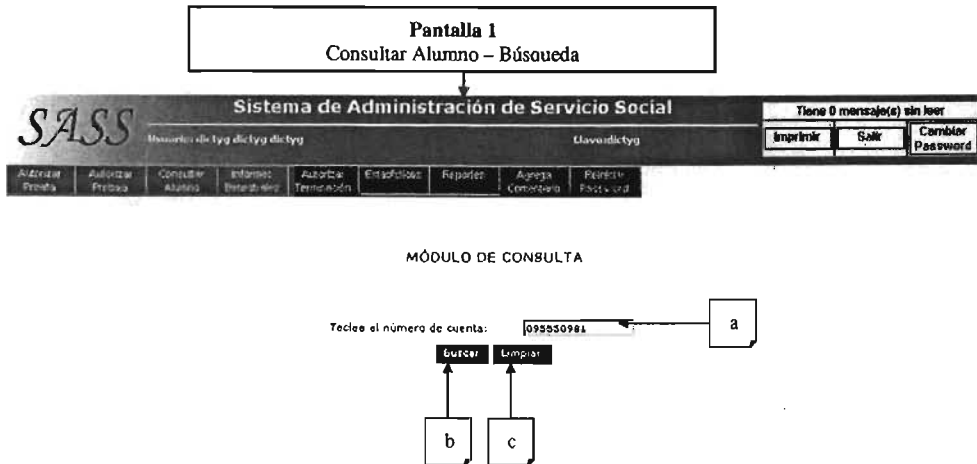
Autoriza Enviar a Cambios Rechaza Cancela

Sunday, August 01, 2004

- a. Motivo por el cual se desea dar la prebaja al alumno.
- b. Autoriza la prebaja del alumno en el programa de SS, remite a la pantalla informativa.
- c. Remite a la pantalla 1.

Consultar Movimientos del Alumno

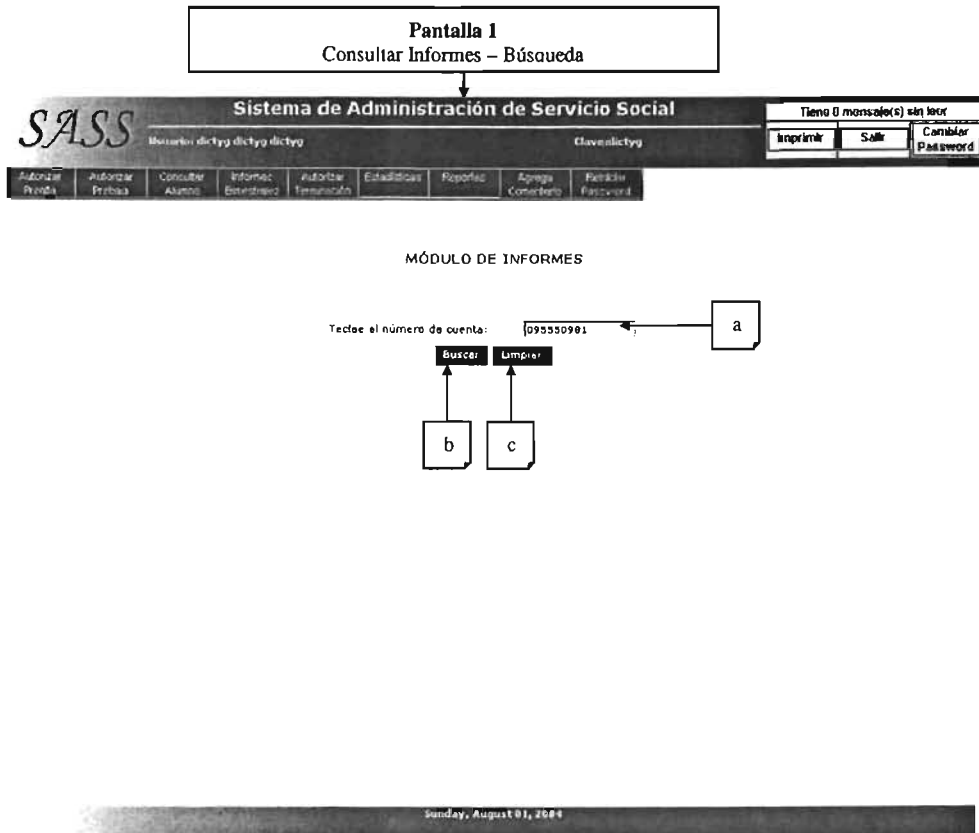
Para permitir al coordinador, consultar los movimientos del alumno durante el ejercicio de su SS, deberá elegir la opción “Consultar Alumno” del menú para el perfil del Coordinador.



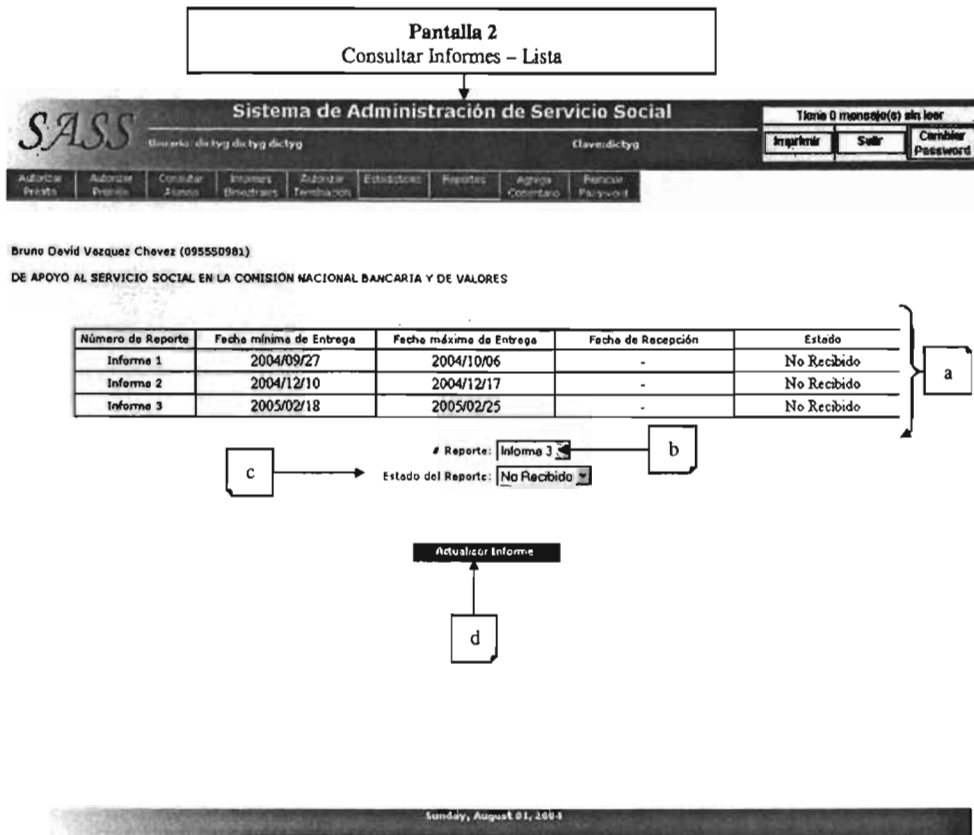
- a. Número de cuenta del alumno.
- b. Obtiene la información de los movimientos del alumno realizados durante el ejercicio de su SS, remite a la pantalla 1 del módulo de Consulta de Movimientos del Alumno.
- c. Limpia el contenido de la casilla de Número de Cuenta del alumno.

Actualizar Informes del Alumno

Para permitir al coordinador, actualizar los informes bimestrales del alumno durante el ejercicio de su SS, deberá elegir la opción “**Informes Bimestrales**” del menú para el perfil del Coordinador.



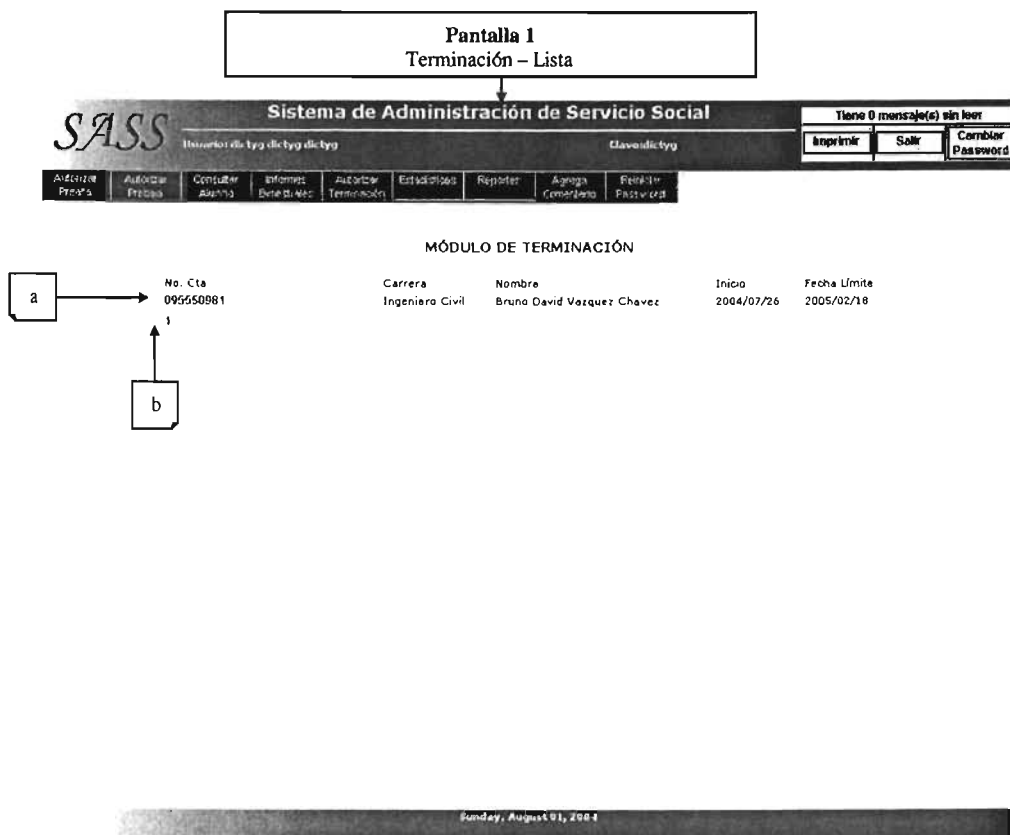
- a. Número de cuenta del alumno.
- b. Obtiene la información de los informes bimestrales del alumno realizados y programados durante el ejercicio de su SS, remite a la pantalla 2.
- c. Limpia el contenido de la casilla de Número de Cuenta del alumno.



- a. Información sobre el estado y fechas del número de informes entregados y por entregar.
- b. Permite seleccionar el número de informe que se desea actualizar.
- c. Permite seleccionar el estado con el que se calificará al informe entregado por el alumno.
- d. Actualiza la fecha de recepción y el estado del informe entregado por el alumno, remite a la pantalla 2 con la información del informe actualizado.

Terminación del Alumno

Para permitir al coordinador, autorizar la terminación de un alumno en un programa de SS, deberá elegir la opción “Autorizar Terminación” del menú para el perfil del Coordinador.



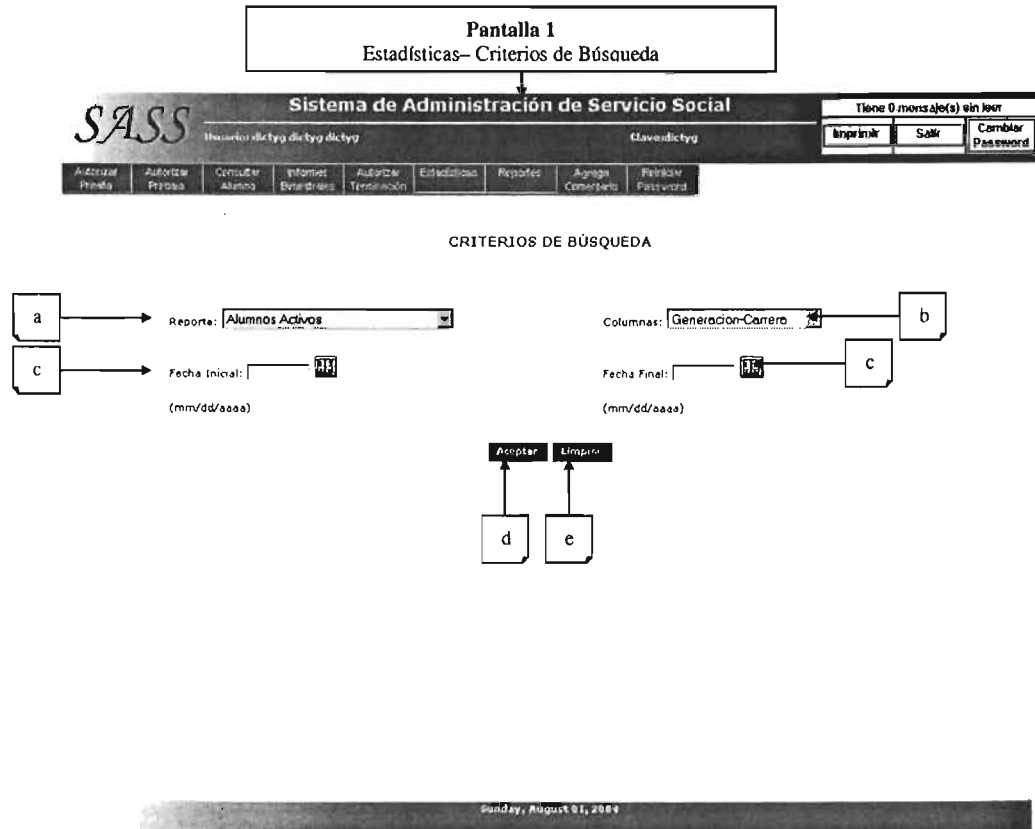
- a. Obtiene la información de los movimientos durante el ejercicio del SS de un alumno, remite a la pantalla 2.
- b. Número de pantallas con información, regresa a la pantalla 1 con los registros del número de pantalla seleccionada.

Pantalla 2
 Terminación – Movimientos del Alumno

- a. Actualiza el estado del alumno a Terminación, remite a la pantalla 1 con los alumnos susceptibles a Terminación.

Estadísticas

Para permitir al coordinador obtener los reportes estadísticos de los alumnos pertenecientes a su división dentro del ejercicio de su servicio social, deberá elegir la opción “Estadísticas” del menú para el perfil del Coordinador.



- a. Tipo de Reporte.
- b. Información que contendrá el reporte.
- c. Rango de fechas como criterio de búsqueda para el tipo de reporte.
- d. Obtiene la información referente al tipo de reporte, remite a la pantalla 2.
- e. Limpia las casillas del rango de fechas.

Pantalla 2
Estadísticas- Reporte estadístico

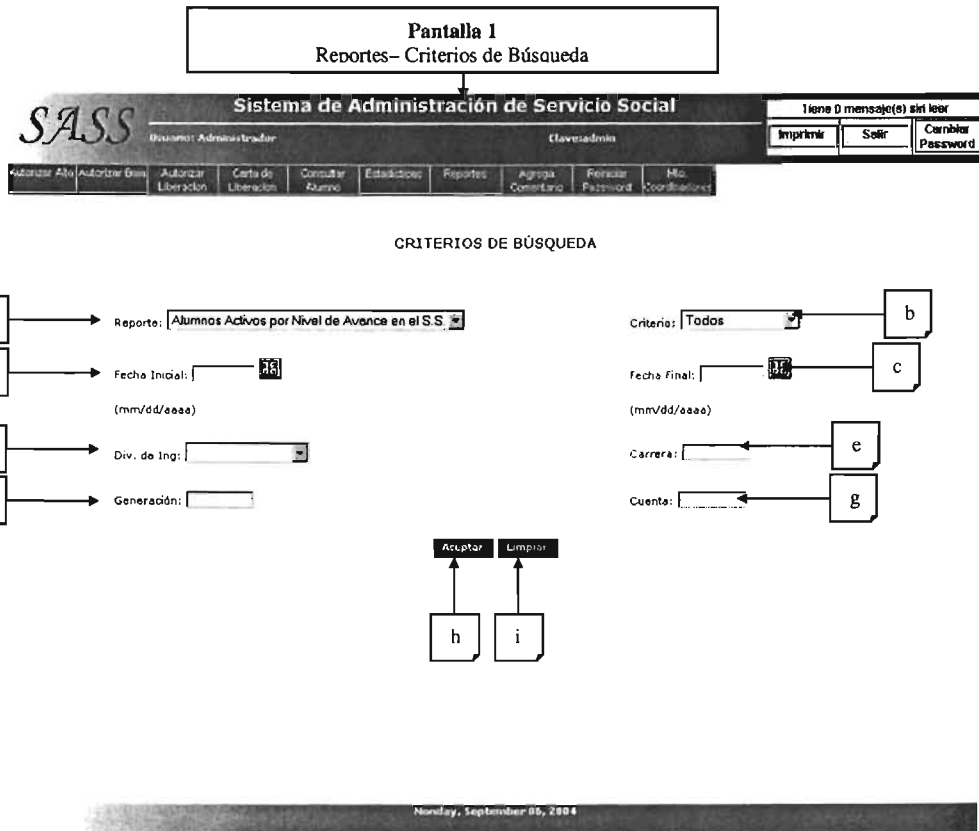


Sistema de Administración de Servicio Social
Fecha de Generación: 2004-09-06 14:28:41.346
Reporte: Alumnos Actives
Período del 2003-09-02 al 2004-09-06
Generado por: Administrador
Estatus/Situación: Generación-Carrera

Generación	Carrera	Cantidad
2000	108	1

Reportes

Para permitir al coordinador obtener los reportes informativos de los alumnos pertenecientes a su división dentro del ejercicio de su servicio social, deberá elegir la opción “Reportes” del menú para el perfil del Coordinador.



- a. Tipo de Reporte.
- b. Información que contendrá el reporte.
- c. Rango de fechas como criterio de búsqueda para el tipo de reporte.
- d. Filtro por división de Ingeniería.
- e. Filtro por carrera.
- f. Filtro por generación.
- g. Filtro por número de cuenta.
- h. Obtiene la información referente al tipo de reporte, remite a la pantalla 2.
- i. Limpia las casillas de los criterios del reporte.

Pantalla 2
Reportes- Reporte informativo

SASS Sistema de Administración de Servicio Social

Usuario: Administrador Clave: admin

Tiene 0 mensajes sin leer

Imprimir Salir Cambiar Password

Inicio de Sesión Avanzar Licencias Calle de Liberación Consulta Avance Estadísticas Reportes Agrupa Consultas Registrar Password Mi Perfil

Sistema de Control de Servicio Social

Fecha de Generación: 2004-09-06 14:45:23.91
 Reporte: Alumnos Aprobados por Nivel de Avance en el S.S.
 Período del 2003-09-01 al 2004-09-06

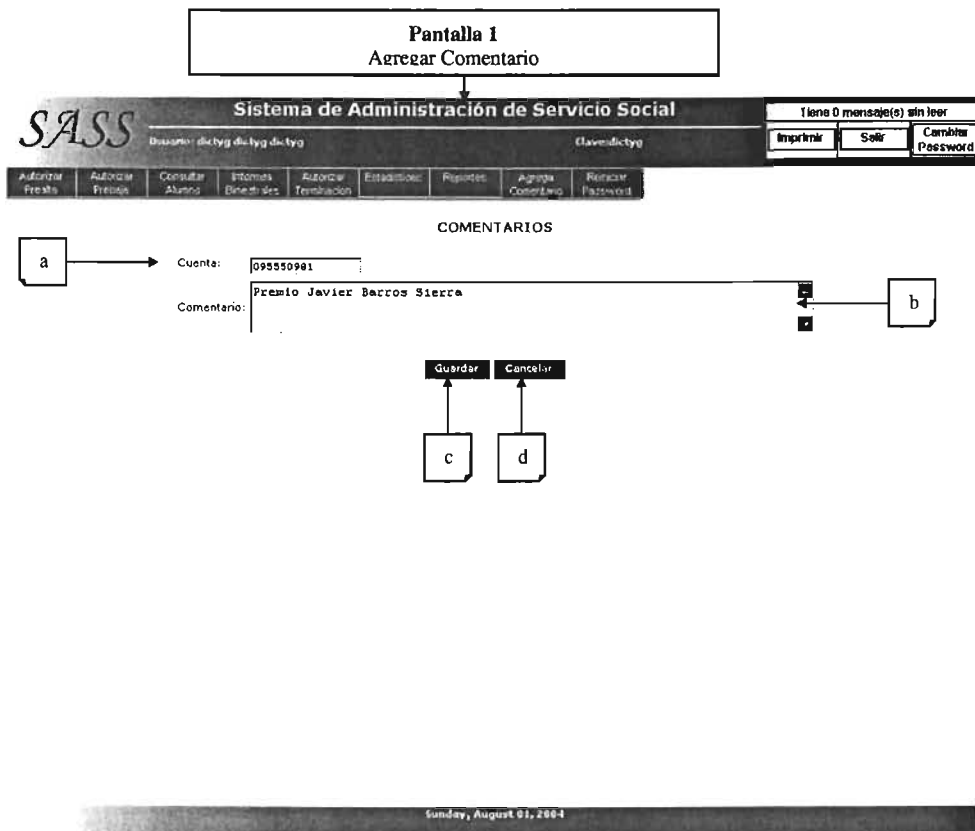
Generado por: Administrador
 Estatus/Situación: Todos

Generación	Dirección	Carrera	Cuarta	Nombre	Situación
DICTYO	107	09330981	Bruno David Vazquez Chavez	Entregada	

Monday, September 06, 2004

Agregar comentario

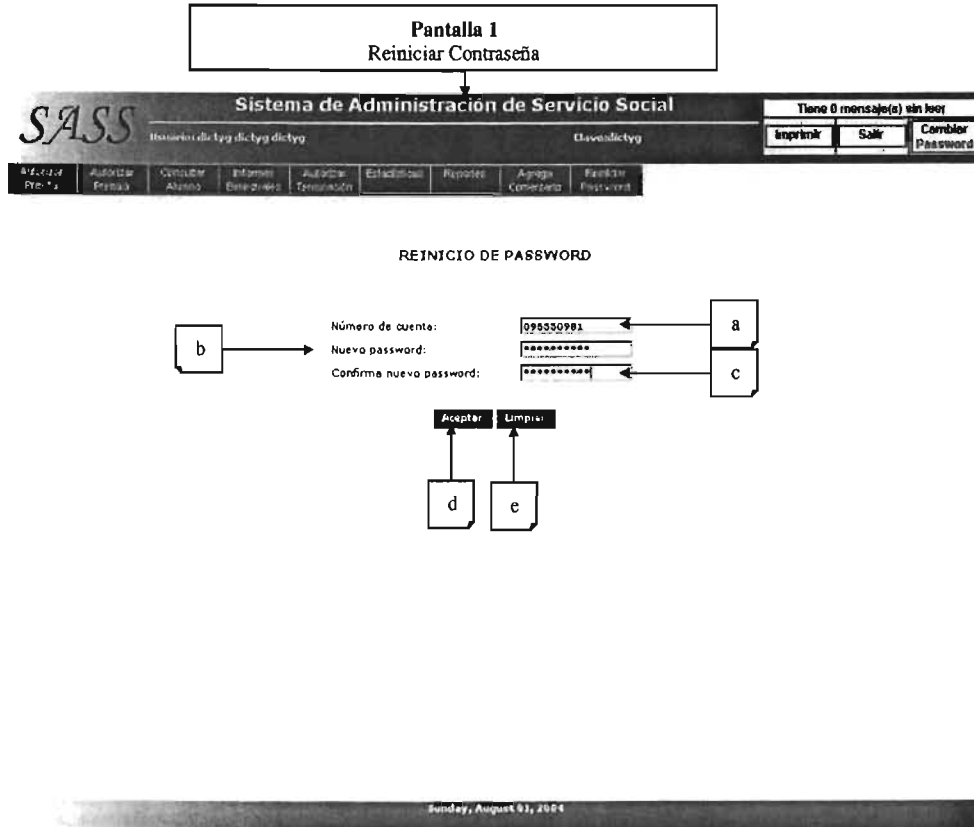
Para permitir al coordinador asociar un comentario o una calificación a un alumno perteneciente a su división dentro del ejercicio de su servicio social, deberá elegir la opción “Agregar Comentario” del menú para el perfil del Coordinador.



- a. Número de cuenta del alumno.
- b. Comentario asociado al alumno.
- c. Guarda el comentario del alumno, remite a la pantalla informativa.
- d. Limpia las casillas del número de cuenta y del comentario asociado al alumno.

Reiniciar contraseña

Para permitir al coordinador reiniciar la contraseña de un alumno dentro del ejercicio de su SS, deberá elegir la opción “**Reiniciar Contraseña**” del menú para el perfil del Coordinador.

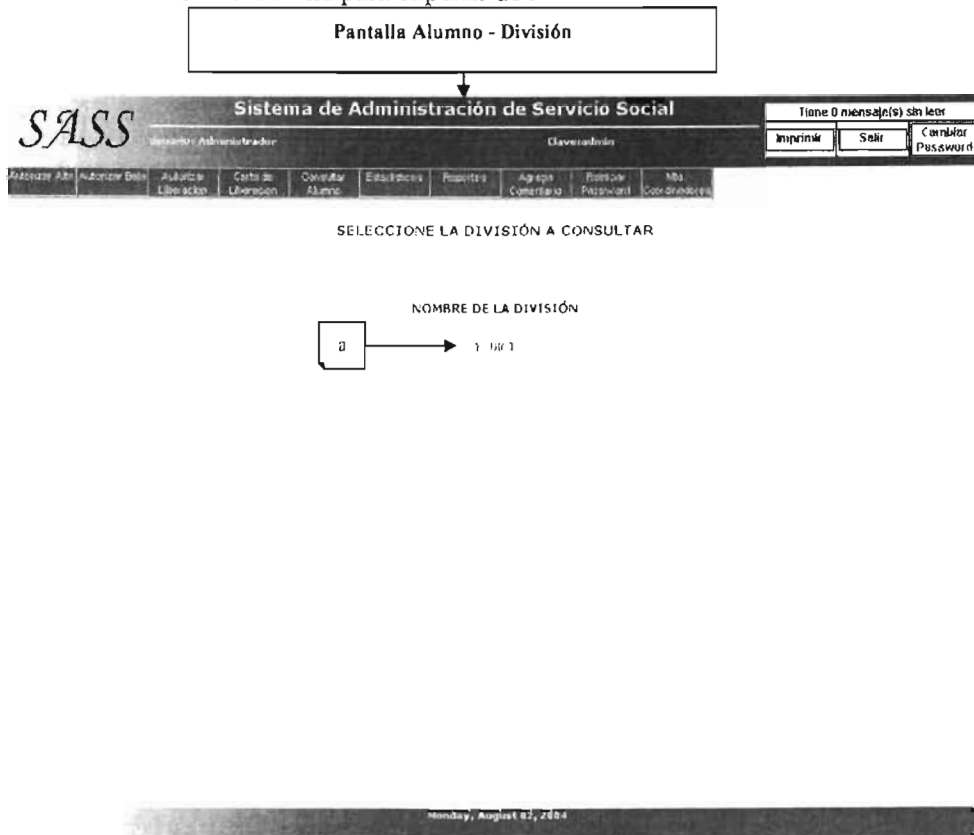


- a. Número de cuenta del alumno.
- b. Contraseña nueva.
- c. Confirmación de la contraseña nueva.
- d. Efectúa el cambio de contraseña del usuario, envía la pantalla general: pantalla informativa.
- e. Limpia las cajas de texto de los campos de contraseña.

4.2.2.4. Flujo de Pantallas para el Jefe de SS de la FI

Pantalla genérica para consulta de alumnos por División

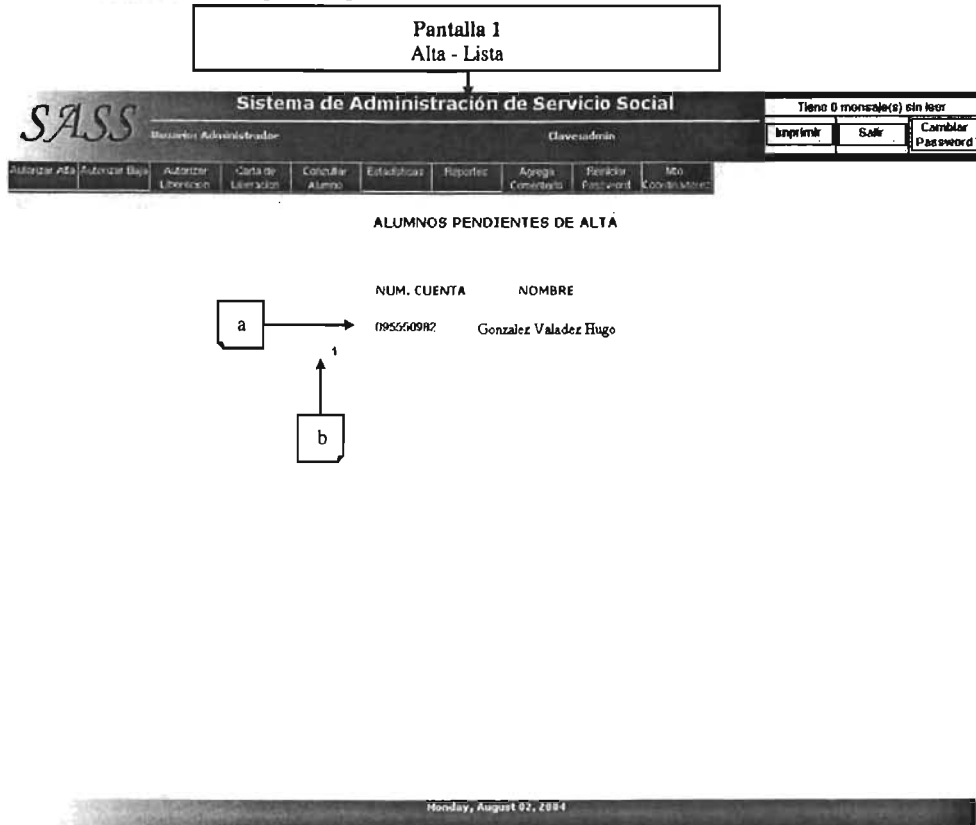
Pantalla genérica para consultar a los alumnos por División que tienen procesos pendientes por autorizar del Jefe de SS, al acceder a los módulos de “**Autorizar Alta**”, “**Autorizar Baja**” y “**Autorizar Liberación**” del menú para el perfil de Jefe.



- a. Número de alumnos - Nombre de la división, muestra información general de los alumnos pertenecientes al módulo de Alta, Baja o Liberación, remite a la pantalla 1 (Autorizar Alta - Lista), remite a la pantalla 1(Autorizar Baja - Lista), remite a la pantalla 1(Autorizar Liberación - Lista)

Alta de un Alumno

Para permitir al jefe autorizar el alta en un programa de SS a un alumno, deberá elegir la opción “Autorizar Alta” del menú para el perfil del Jefe.



- a. Obtiene la información de la solicitud del alumno, remite a la pantalla 2.
- b. Número de pantallas con información, regresa a la pantalla 1 con los registros del número de pantalla seleccionada.

Pantalla 2
Alta - Datos del Alumno

a. Fecha de Inicio de SS.

Pantalla 2
Alta – Datos de la Dependencia

SASS

Sistema de Administración de Servicio Social
 Usuario: Administrador Clave: admin

Tiene 0 mensaje(s) sin leer

Autorizar Alta Autorizar Baja Autorizar Liberación Consultar Estado Estadísticas Reportes Agenda Registrar Mo. Coordinación

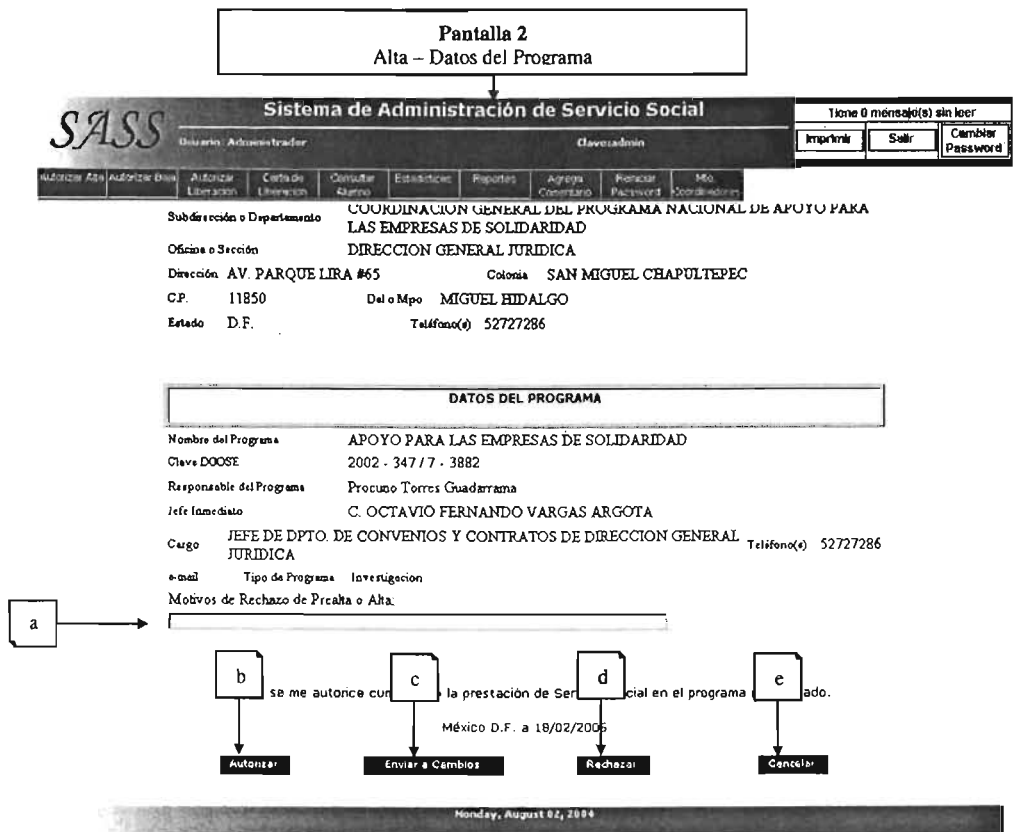
DATOS DE LA DEPENDENCIA

Nombre de la Dependencia	SECRETARIA DE ECONOMIA		
Subdirección o Departamento	COORDINACION GENERAL DEL PROGRAMA NACIONAL DE APOYO PARA LAS EMPRESAS DE SOLIDARIDAD		
Oficina o Sección	DIRECCION GENERAL JURIDICA		
Dirección	AV. PARQUE LIRA #65	Colonia	SAN MIGUEL CHAPULTEPEC
C.P.	11850	Del o Mpo	MIGUEL HIDALGO
Estado	D.F.	Teléfono(s)	52727286

DATOS DEL PROGRAMA

Nombre del Programa	APOYO PARA LAS EMPRESAS DE SOLIDARIDAD		
Clave DOOSE	2002 - 347 / 7 - 3882		
Responsable del Programa	Procuno Torres Guadarrama		
Jefe inmediato	C. OCTAVIO FERNANDO VARGAS ARGOTA		
Cargo	JEFE DE DPTO. DE CONVENIOS Y CONTRATOS DE DIRECCION GENERAL JURIDICA	Teléfono(s)	52727286
e-mail	Tipo de Programa	Investigacion	
Motivos de Rechazo de Preacta o Aña:			

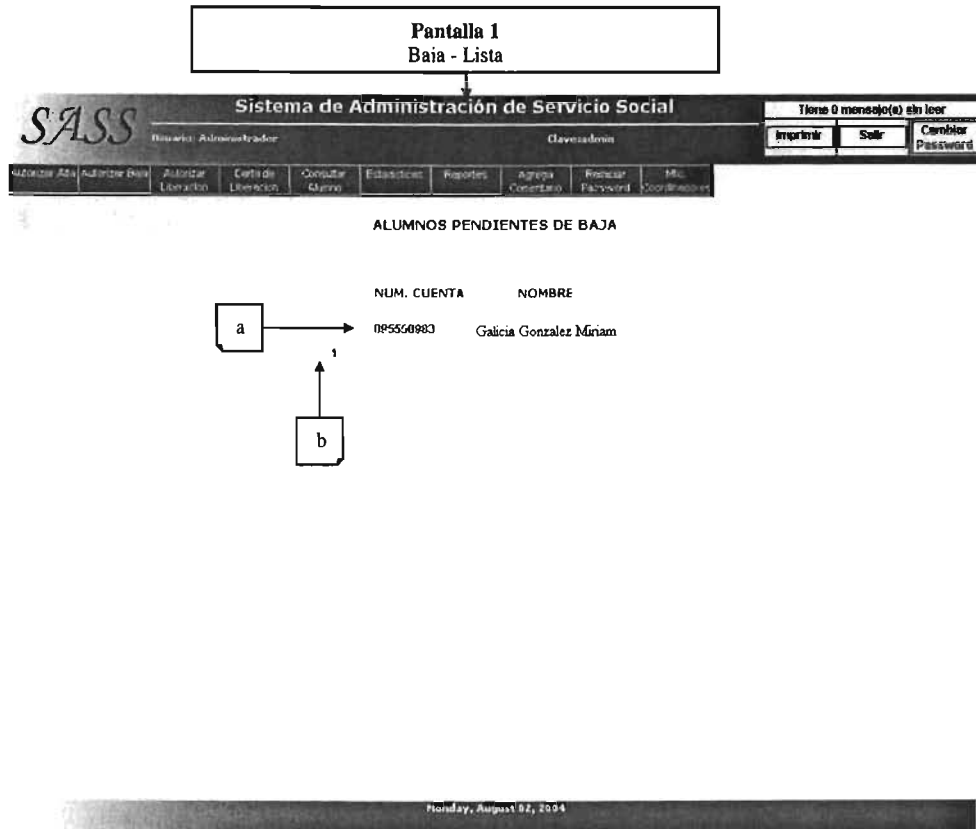
Monday, August 02, 2004



- a. Motivo por el cual se rechazará el trámite de alta del alumno.
- b. Autoriza el alta del alumno en el programa de SS, remite a la pantalla informativa.
- c. Le envía un mensaje al alumno indicándole que debe modificar los datos de su registro, remite a la pantalla informativa.
- d. Rechaza el movimiento del alumno, remite a la pantalla informativa.
- e. Remite a la pantalla 1.

Baja de un Alumno

Para permitir al jefe autorizar la baja en un programa de SS a un alumno, deberá elegir la opción “Autorizar Baja” del menú para el perfil del Jefe.



- a. Obtiene la información de la solicitud del alumno, remite a la pantalla 2.
- b. Número de pantallas con información, regresa a la pantalla 1 con los registros del número de pantalla seleccionada.

Pantalla 2
Baía - Datos del Alumno

SASS

Sistema de Administración de Servicio Social
Módulo Administrador Claveadmin

Tiene 0 mensaje(s) en leer

Autorizar Alta
Autorizar Baja
Autorizar Liberación
Carta de Liberación
Consultar Alumno
Estadísticas
Reportes
Agrupar Consultas
Relación Padres
Año

**SOLICITUD DE AUTORIZACIÓN DE PRESTACIÓN DEL
SERVICIO SOCIAL E INFORMACIÓN ESTADÍSTICA**

DATOS DEL ALUMNO

Número de cuenta: 095550982	Clave de la carrera: 108	Ingreso a la FI.: 0
Nombre:	Gonzalez	Valedet
	Ap. Patero	Ap. Materno
Edad: 24		Sexo: M
Calle: Calle	Número: Numero 3	Colonia: Colonia Tlacotalera
C.P.: 9879	Estado: Mexico	Del o Mpo.: Tlaxtepania
Tel. particular: 3456789	Tel. celular: 76456787	Localizador: 987987 PIN: 878
Créditos aprobados: 0	Avance: 0.00	Promedio: 0.00
Fecha de inicio: 2004/08/01	Duración en Meses: 6	Horas/Semana:
Forma de remuneración: 4	Percepción mensual: 2500	

DATOS DE LA DEPENDENCIA

Nombre de la Dependencia:	SECRETARIA DE ECONOMIA
Subdirección o Departamento:	COORDINACION GENERAL DEL PROGRAMA NACIONAL DE APOYO PARA LAS EMPRESAS PYMES

Monday, August 02, 2004

Pantalla 2
Baja – Datos de la Dependencia

SASS										Tiene 0 mensajes(s) sin leer		
Sistema de Administración de Servicio Social												
Usuario: Administrador					Clave: admin							
									<input type="button" value="Imprimir"/> <input type="button" value="Salir"/> <input type="button" value="Cambiar Password"/>			
Autónoma	Alta	Autorización	Asistencia	Cartas de Liberación	Consultar	Estadísticas	Reportes	Agencia	Financiera	Módulos		
			Liberación	Alumno				Consultas	Facturas	Administración		

DATOS DE LA DEPENDENCIA

Nombre de la Dependencia SECRETARIA DE ECONOMIA
Subdirección o Departamento COORDINACION GENERAL DEL PROGRAMA NACIONAL DE APOYO PARA LAS EMPRESAS DE SOLIDARIDAD
Oficina o Sección DIRECCION GENERAL JURIDICA
Dirección AV. PARQUE LIRA #65 Colonia SAN MIGUEL CHAPULTEPEC
C.P. 11850 Dal o Mpo MIGUEL HIDALGO
Estado D.F. Telefono(s) 52727286

DATOS DEL PROGRAMA

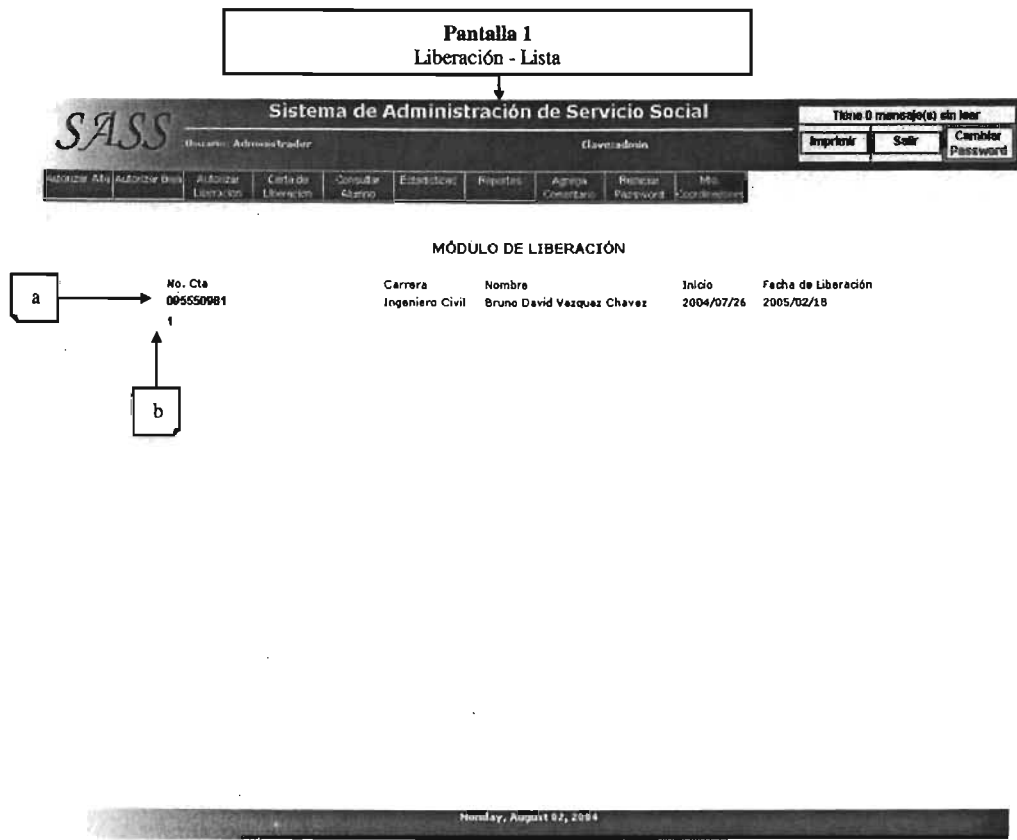
Nombre del Programa APOYO PARA LAS EMPRESAS DE SOLIDARIDAD
Clave DOOSE 2002 - 347 / 7 - 3882
Responsable del Programa Procuno Torres Guadarrama
Jefe Inmediato C. OCTAVIO FERNANDO VARGAS ARGOTA
Cargo JEFE DE DPTO. DE CONVENIOS Y CONTRATOS DE DIRECCION GENERAL JURIDICA Telefono(s) 52727286
e-mail Tipo de Programa Investigación
Motivos de Rechazo de Preacta o Alta:

Pantalla 2
 Baja – Datos del Programa

- a. Motivo por el cual se rechazará la baja del alumno.
- b. Autoriza la baja del alumno en el programa de SS, remite a la pantalla informativa.
- c. Rechaza el movimiento del alumno, remite a la pantalla informativa.
- d. Remite a la pantalla 1.

Liberación de un Alumno

Para permitir al jefe autorizar la liberación en un programa de SS a un alumno, deberá elegir la opción **“Autorizar Liberación”** del menú para el perfil del Jefe.



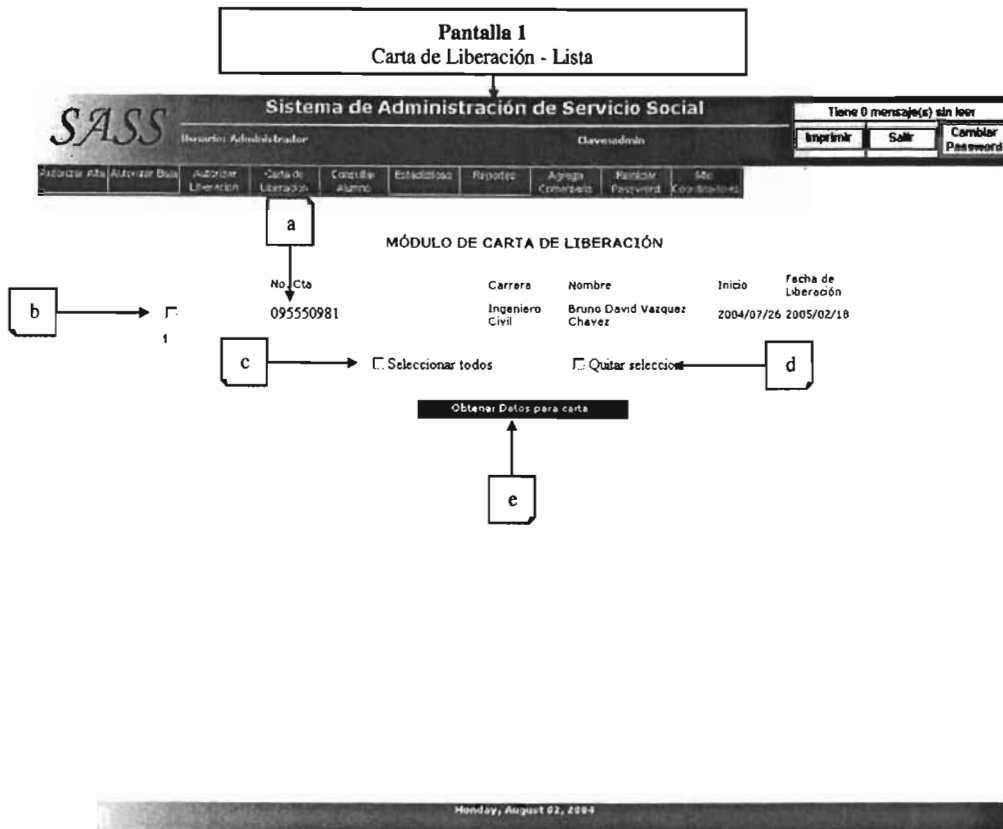
- a. Obtiene la información de los movimientos durante el ejercicio del SS de un alumno, remite a la pantalla 2.
- b. Número de pantallas con información, regresa a la pantalla 1 con los registros del número de pantalla seleccionada.

Pantalla 2
 Liberación - Consulta

- a. Actualiza el estado del alumno a Liberación, remite a la pantalla 1 con los alumnos susceptibles a Liberación.
- b. Rechaza el movimiento del alumno, remite a la pantalla 1 con los alumnos susceptibles a Liberación.
- c. Motivo para rechazar la liberación.

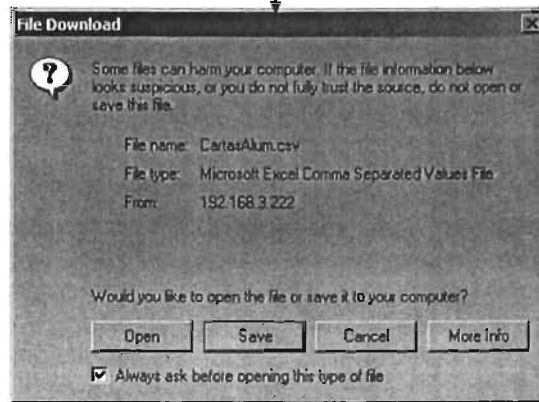
Carta de Liberación de un Alumno

Para permitir al jefe obtener la información sobre la carta de liberación de un alumno, deberá elegir la opción “**Carta de Liberación**” del menú para el perfil del Jefe.



- a. Obtiene la información de los movimientos durante el ejercicio del SS de un alumno, remite a la pantalla 2
- b. Permite seleccionar diferentes alumnos para obtener la información para la impresión de la carta de liberación.
- c. Selecciona a todos los alumnos susceptibles a impresión de carta de Liberación.
- d. Quita la selección de los alumnos seleccionados para obtener la información para la impresión de la carta de liberación.
- e. Exporta la información de la carta de liberación del alumno, remite a la pantalla 2.

Pantalla 2
Carta de Liberación - Exportación



Guarda la información de la carta del alumno en un archivo de excel en la máquina del Jefe, para su posterior explotación.

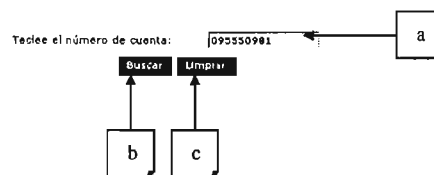
Consultar Movimientos del Alumno

Para permitir al jefe, consultar los movimientos del alumno durante el ejercicio de su SS, deberá elegir la opción “Consultar Alumno” del menú para el perfil del Jefe.

Pantalla 1
Consultar Alumno – Búsqueda



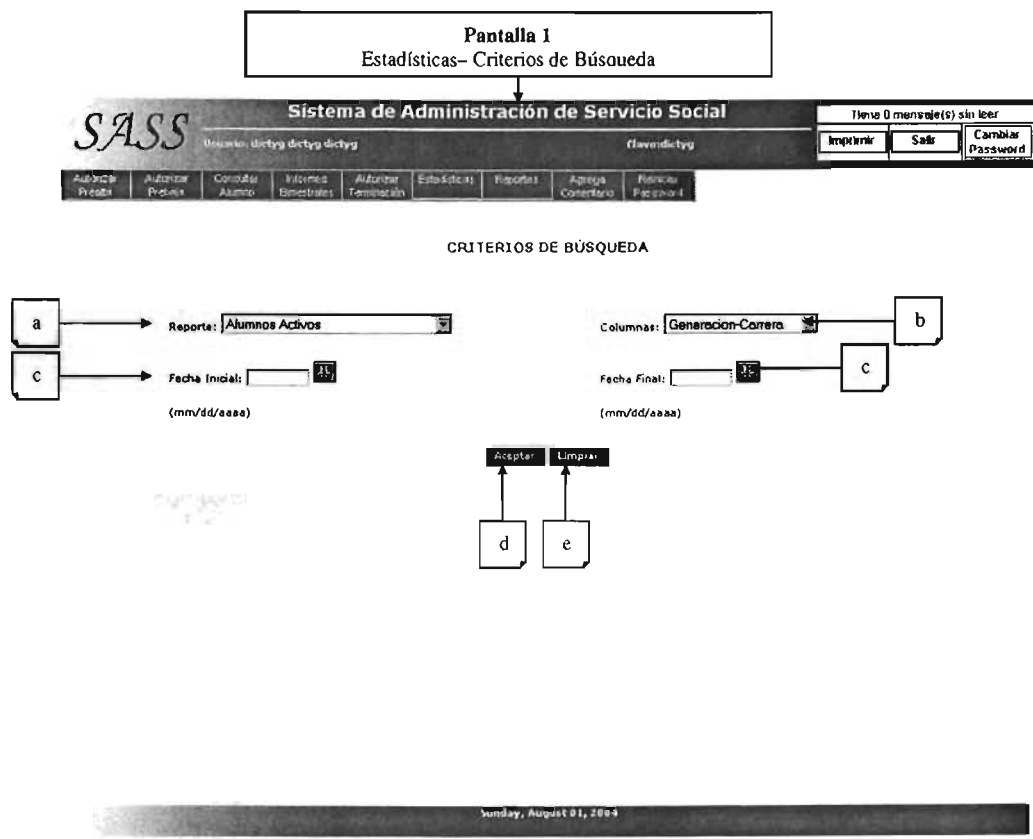
MÓDULO DE CONSULTA



- a. Número de cuenta del alumno.
- b. Obtiene la información de los movimientos del alumno realizados durante el ejercicio de su SS, remite a la pantalla 1 del módulo de Consulta de Movimientos del Alumno.
- c. Limpia el contenido de la casilla de Número de Cuenta del alumno.

Estadísticas

Para permitir al jefe obtener los reportes estadísticos de los procesos de los alumnos durante del ejercicio de su servicio social, deberá elegir la opción “**Estadísticas**” del menú para el perfil del Jefe.



- a. Tipo de Reporte.
- b. Información que contendrá el reporte.
- c. Rango de fechas como criterio de búsqueda para el tipo de reporte.
- d. Obtiene la información referente al tipo de reporte, remite a la pantalla 2.
- e. Limpia las casillas del rango de fechas.

Pantalla 2
Estadísticas- Reporte estadístico

SASS

Sistema de Administración de Servicio Social
Usuario: Administrador Clave: admin

Tiene 0 mensaje(s) sin leer

Autorizar AltaAutorizar BajaAutorizar LiberaciónCarta de LiberaciónConsulta AlumnoEstadísticasReportesAlgebra ContableRenovar PasswordMío

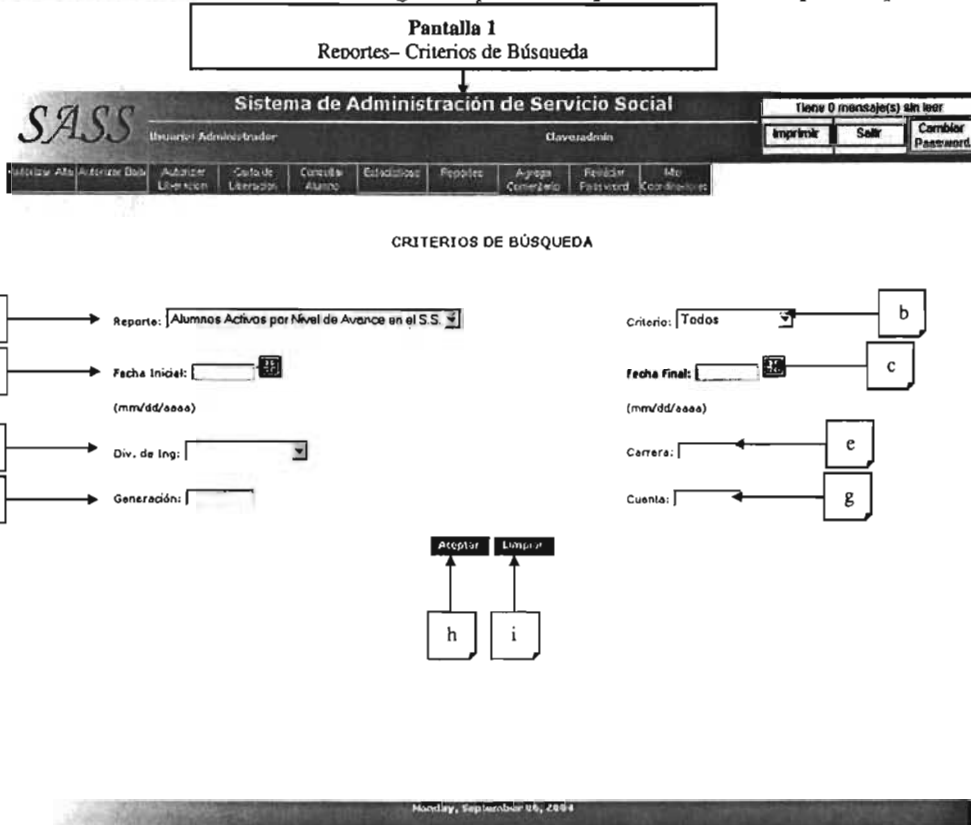
Sistema de Administración de Servicio Social

Fecha de Generación: 2004-09-06 14:28:41.346 Generado por: Administrador
 Reporte: Alumnos Activos Estatus/Situación: Generacion-Carrera
 Periodo del 2003-09-02 al 2004-09-06

Generación	Carrera	Cantidad
2000	108	1

Reportes

Para permitir al jefe obtener los reportes informativos de los procesos de los alumnos durante del ejercicio de su servicio social, deberá elegir la opción “Reportes” del menú para el perfil del Jefe.



- a. Tipo de Reporte.
- b. Información que contendrá el reporte.
- c. Rango de fechas como criterio de búsqueda para el tipo de reporte.
- d. Filtro por división de Ingeniería.
- e. Filtro por carrera.
- f. Filtro por generación.
- g. Filtro por número de cuenta.
- h. Obtiene la información referente al tipo de reporte, remite a la pantalla 2.
- i. Limpia las casillas de los criterios del reporte.

Pantalla 2
Reportes- Reporte informativo

Sistema de Control de Servicio Social

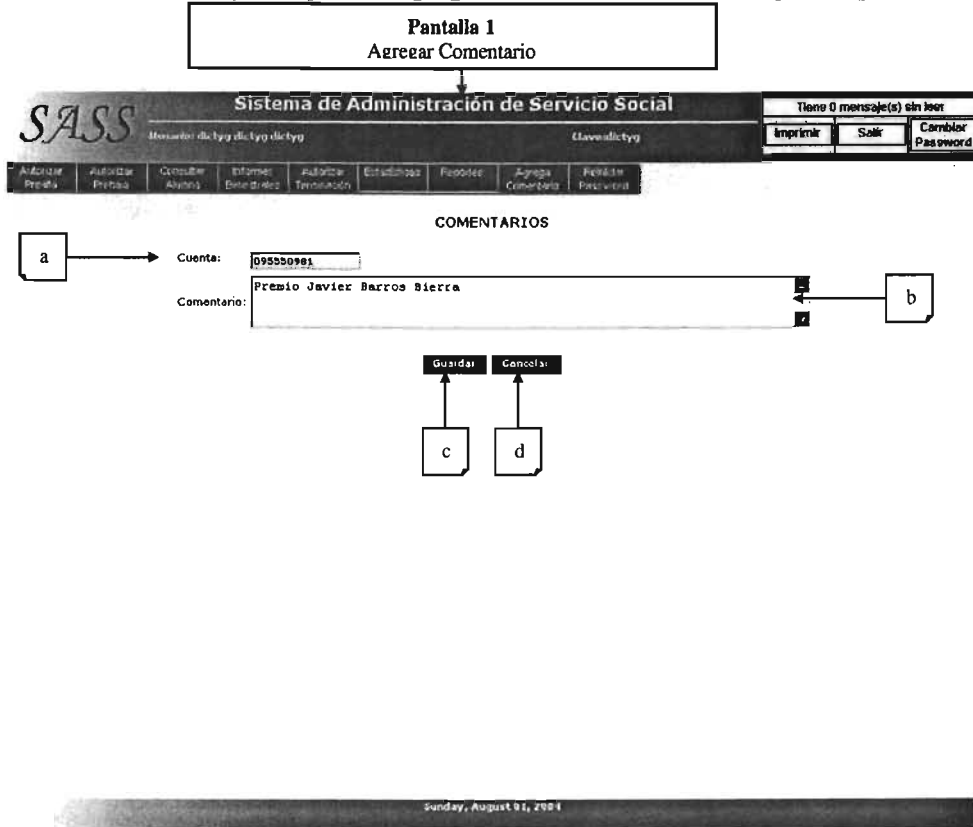
Fecha de Generación: 2004-09-06 14:45:23.91
 Reporte: Alumnos Activos por Nivel de Avance en el S.S.
 Período del 2003-09-01 al 2004-09-06

Generado por: Administrador
 Estatus/Situación: Todos

Generación	Dirección	Carrera	Cuenta	Nombre	Situación
DICTYO	107	09-5530981	Bruno David Vazquez Chaver	Entregada	

Agregar Comentario

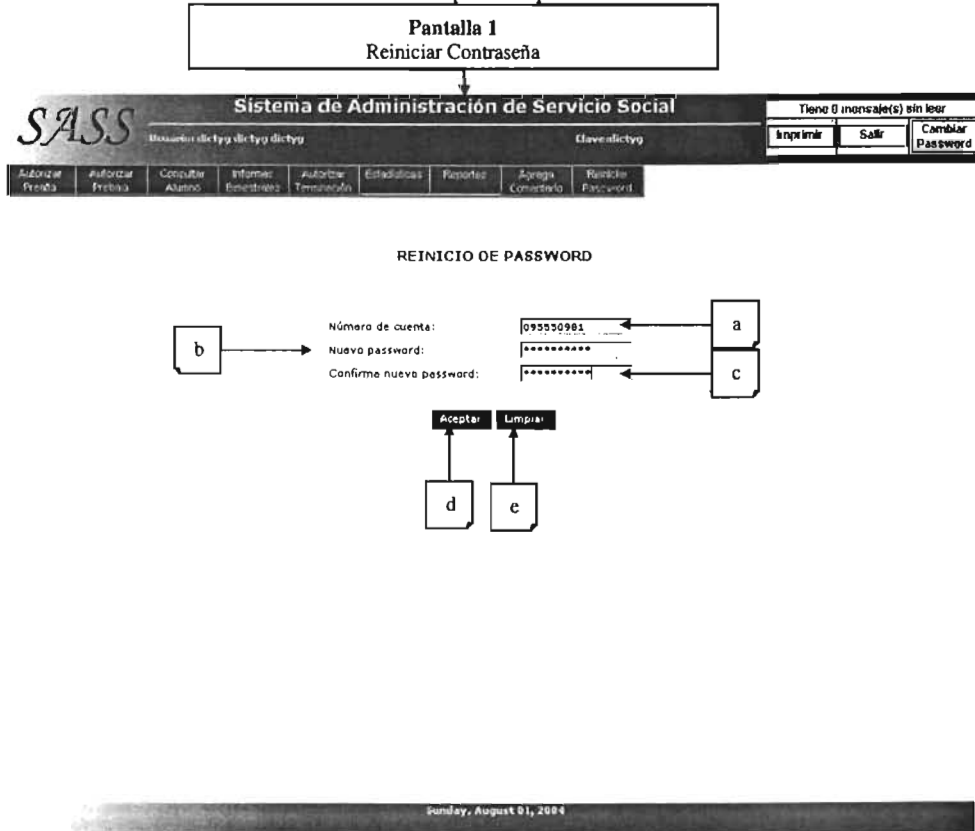
Para permitir al jefe asociar un comentario o una calificación a un alumno dentro del ejercicio de su servicio social, deberá elegir la opción **“Agregar Comentario”** del menú para el perfil del jefe.



- a. Número de cuenta del alumno.
- b. Comentario asociado al alumno.
- c. Guarda el comentario del alumno, remite a la pantalla informativa.
- d. Limpia las casillas del número de cuenta y del comentario asociado al alumno.

Reiniciar Contraseña

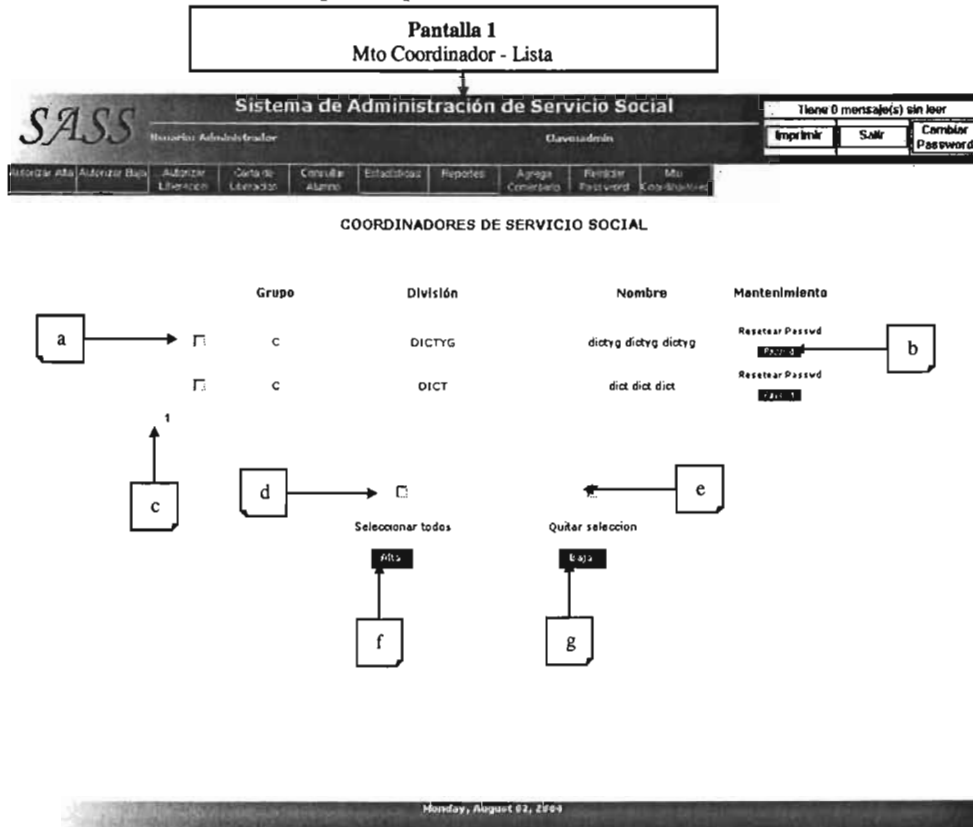
Para permitir al jefe reiniciar la contraseña de un alumno dentro del ejercicio de su SS, deberá elegir la opción “Reiniciar Contraseña” del menú para el perfil del Jefe.



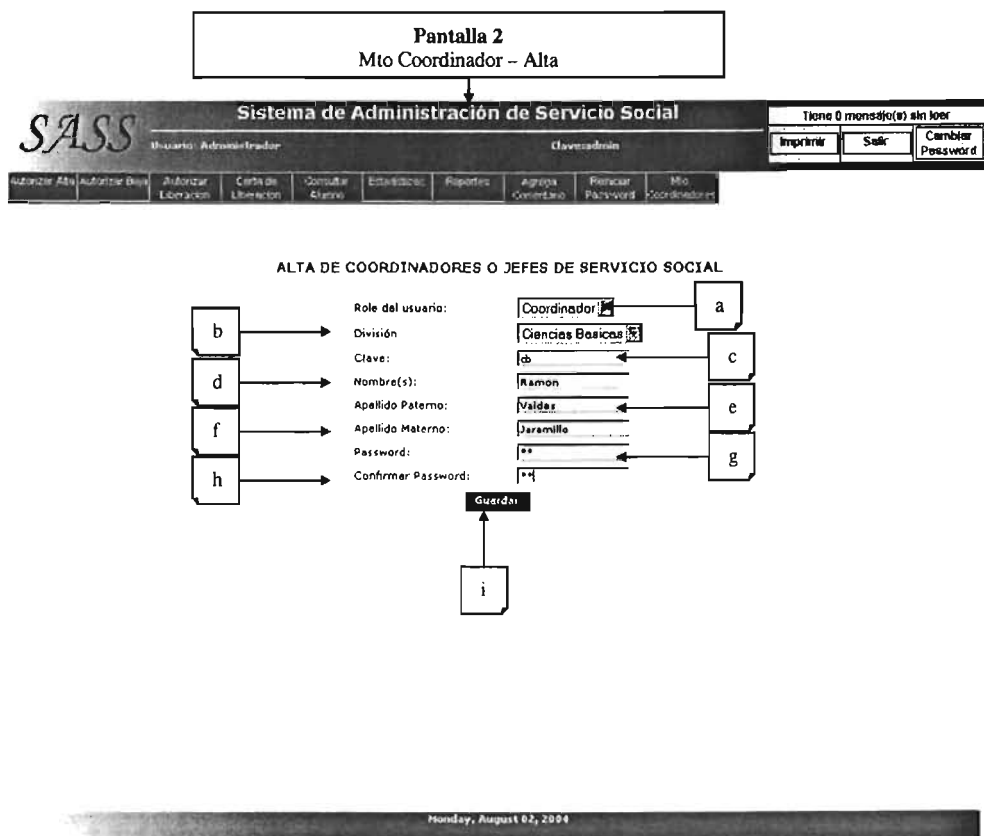
- a. Número de cuenta del alumno.
- b. Contraseña nueva.
- c. Confirmación de la contraseña nueva.
- d. Efectúa el cambio de contraseña del usuario, envía la pantalla general: pantalla informativa.
- e. Limpia las cajas de texto de los campos de contraseña.

Mantenimiento a Coordinadores

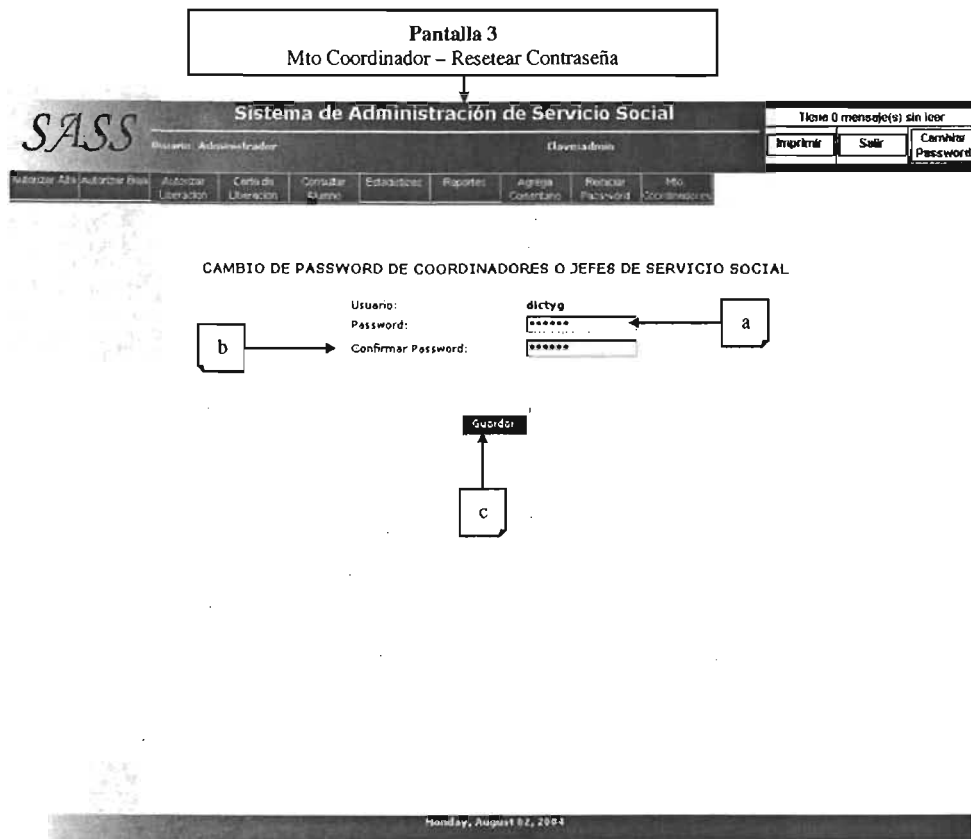
Para permitir al jefe dar mantenimiento a las cuentas de los coordinadores, deberá elegir la opción “Mto Coordinadores” del menú para el perfil del Jefe.



- a. Permite seleccionar los registros de los coordinadores o jefes para darlos de baja.
- b. Permite cambiar la contraseña de un coordinador o un jefe, remite a la pantalla 3.
- c. Número de pantallas con información, regresa a la pantalla 1 con los registros del número de pantalla seleccionada.
- d. Permite seleccionar todos los registros de los coordinadores o jefes.
- e. Permite quitar la selección de todos los registros de los coordinadores o jefes.
- f. Permite dar de alta un nuevo coordinador o jefe, remite a la pantalla 2.
- g. Permite dar de baja a un coordinador o un jefe, remite a la pantalla 1 sin que aparezca en la lista el coordinador o jefe borrado.



- a. Role del usuario, que puede ser coordinador o jefe.
- b. División de Ingeniería.
- c. Clave del sistema para el coordinador o jefe.
- d. Nombre del Coordinador o Jefe.
- e. Apellido Paterno del coordinador o jefe.
- f. Apellido materno del coordinador o jefe.
- g. Contraseña del sistema para el coordinador o jefe.
- h. Confirmación de la contraseña del coordinador o jefe.
- i. Permite guardar la información del nuevo coordinador o jefe, remite a la pantalla 1, con el nuevo coordinador o jefe dado de alta.



- a. Contraseña actual del coordinador o jefe al que se le desea cambiar la contraseña.
- b. Confirmación de la contraseña del coordinador o jefe.
- c. Guarda la nueva contraseña del coordinador o jefe, remite a la pantalla 1.

4.2.3. Diagrama de Clases

De acuerdo al análisis de los objetos que constituyen el dominio, se obtiene el siguiente diagrama de clases.

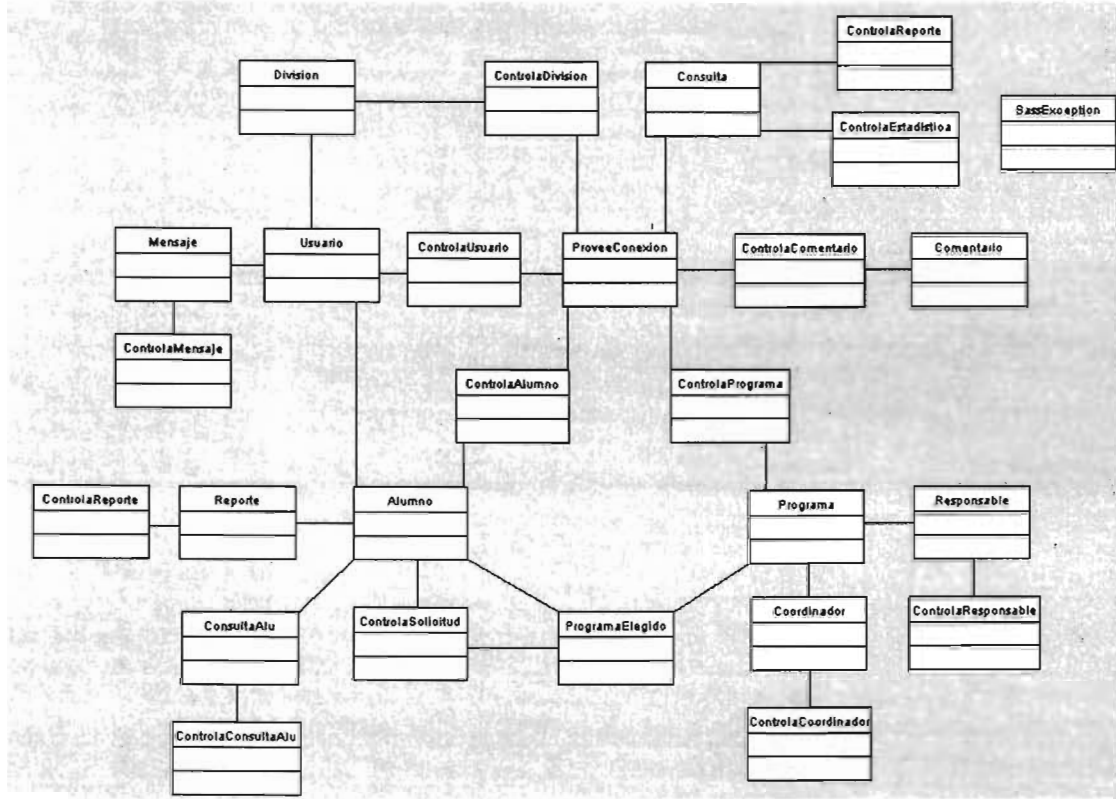


Figura 4.4 Diagrama General de Clases.

4.2.4. Diagramas de Secuencias

Para los casos de uso definidos con anterioridad en este capítulo, se presentan a continuación los diagramas de secuencias del sistema, en los cuales se muestra la interacción de los objetos con los mensajes enviados entre ellos durante el paso del tiempo en el cual se provee la funcionalidad al usuario.

Los diagramas se relacionan con los casos de uso por medio de la clave y el nombre asignado a cada uno de éstos.

Clave: SASS01AU.

Caso de uso: Validar Usuario.

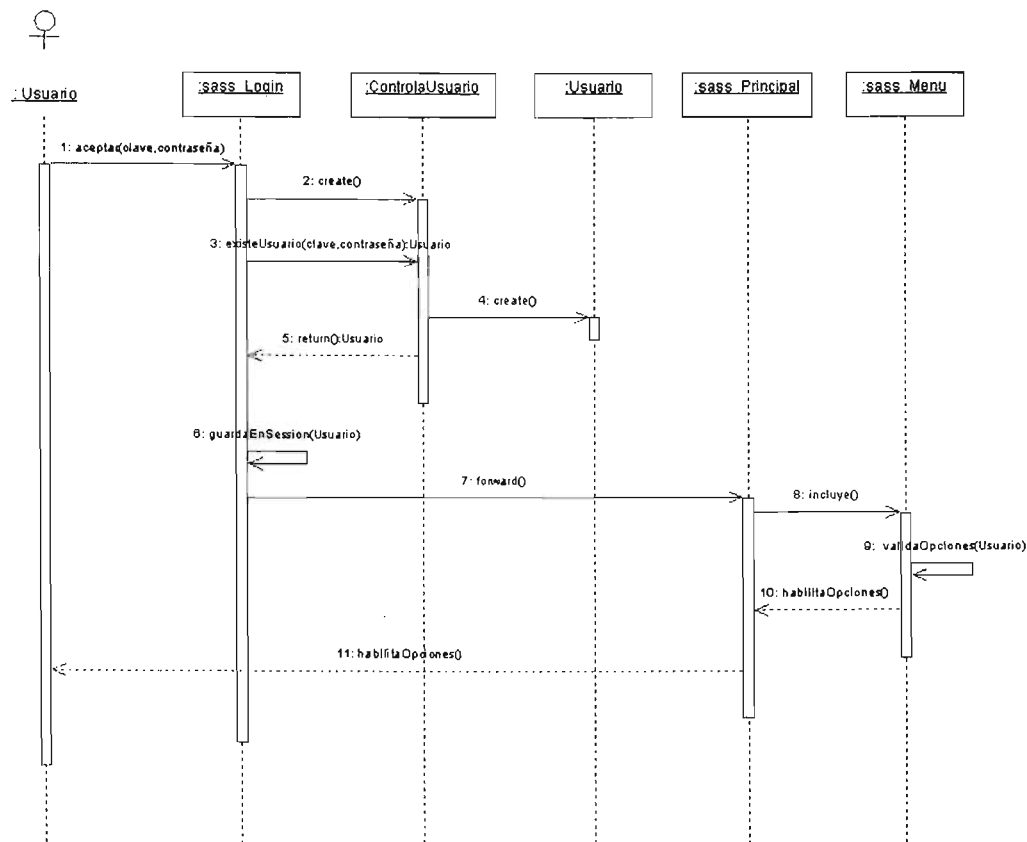


Figura 4.5 Diagrama de Secuencias .Caso de Uso SASS01AU.

Clave: SASS02R.

Caso de uso: Registrar un Alumno en el Sistema de Administración de SS.

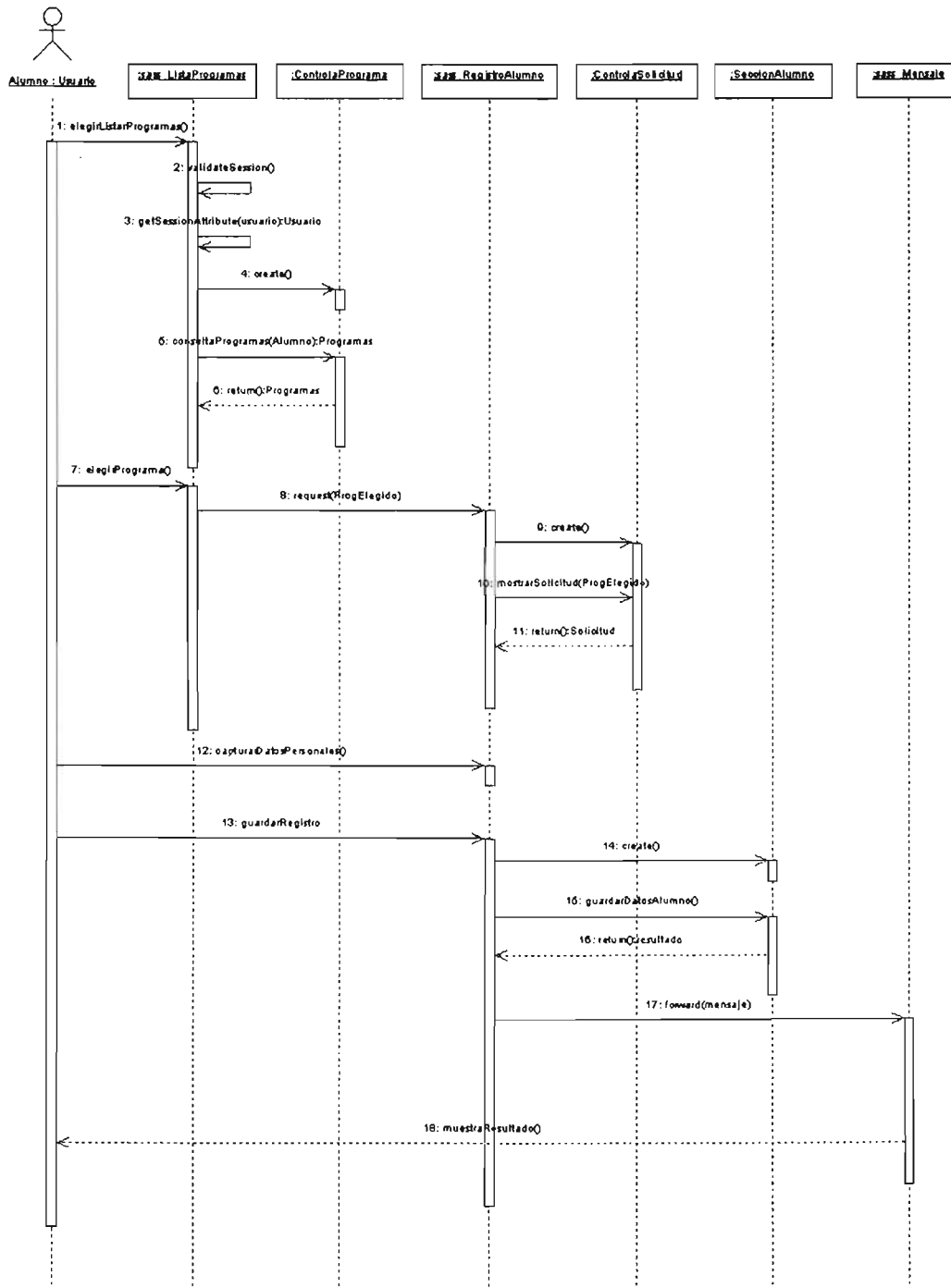


Figura 4.6a Diagrama de Secuencias. Caso de Uso SASS02R (Modalidad: Listar Programas).

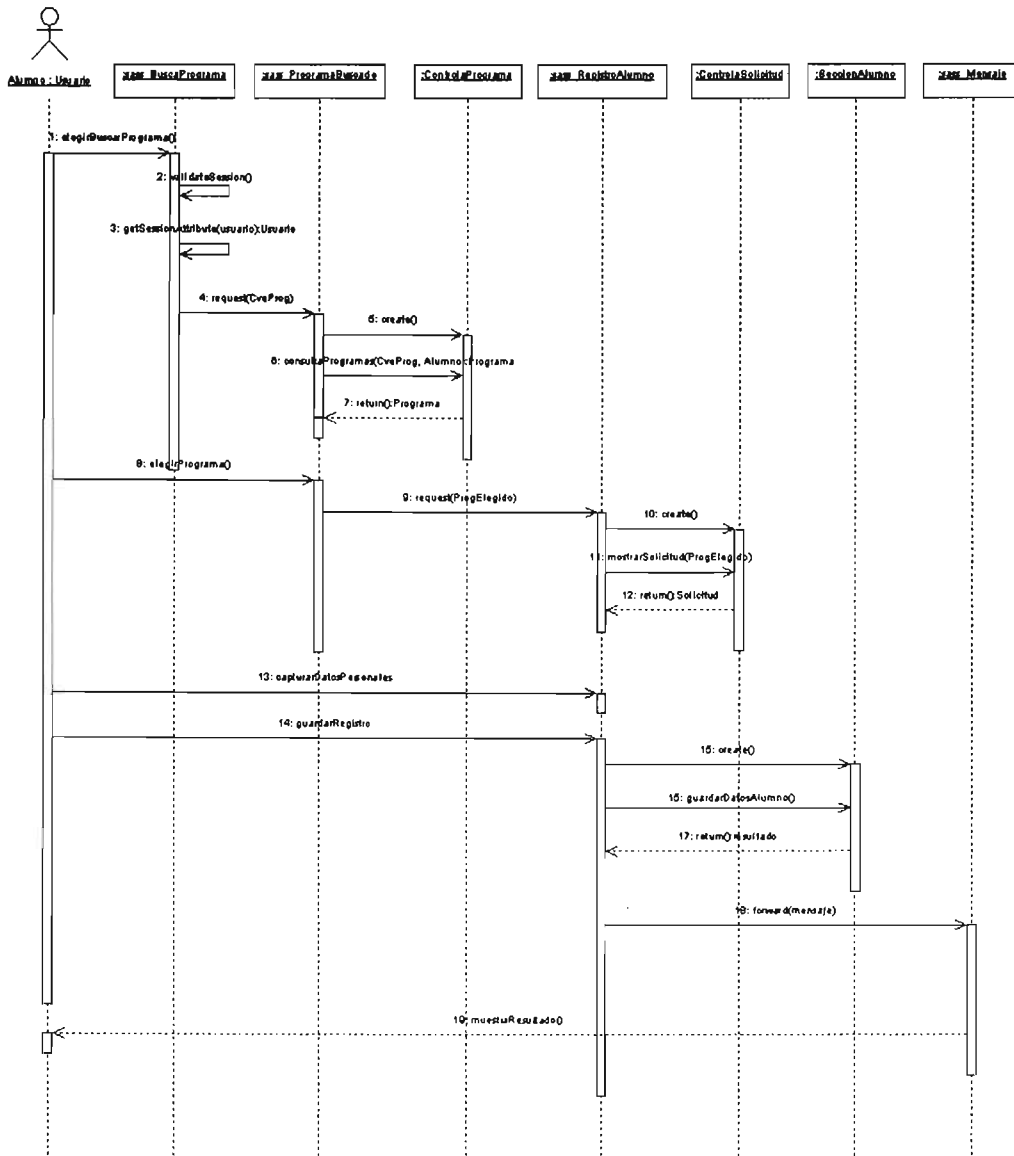


Figura 4.6b Diagrama de Secuencias. Caso de Uso SASS02R (Modalidad: Buscar Programa).

Clave: SASS03PA.

Caso de uso: Autorizar Preacta de un Alumno en el Sistema de Administración del SS.

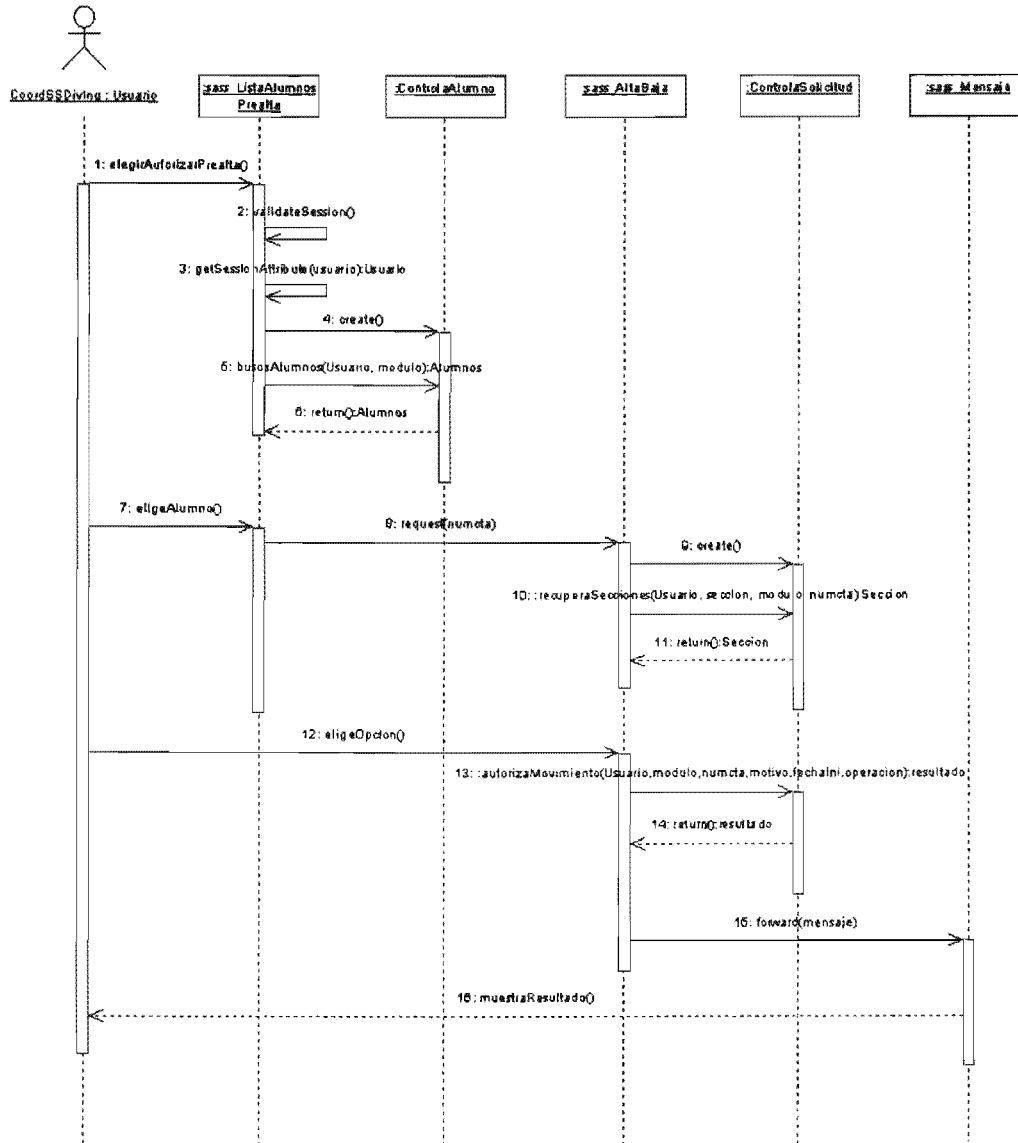


Figura 4.7 Diagrama de Secuencias. Caso de Uso SASS03PA.

Clave: SASS04A.

Caso de uso: Autorizar Alta de un Alumno en el Sistema de Administración del SS.

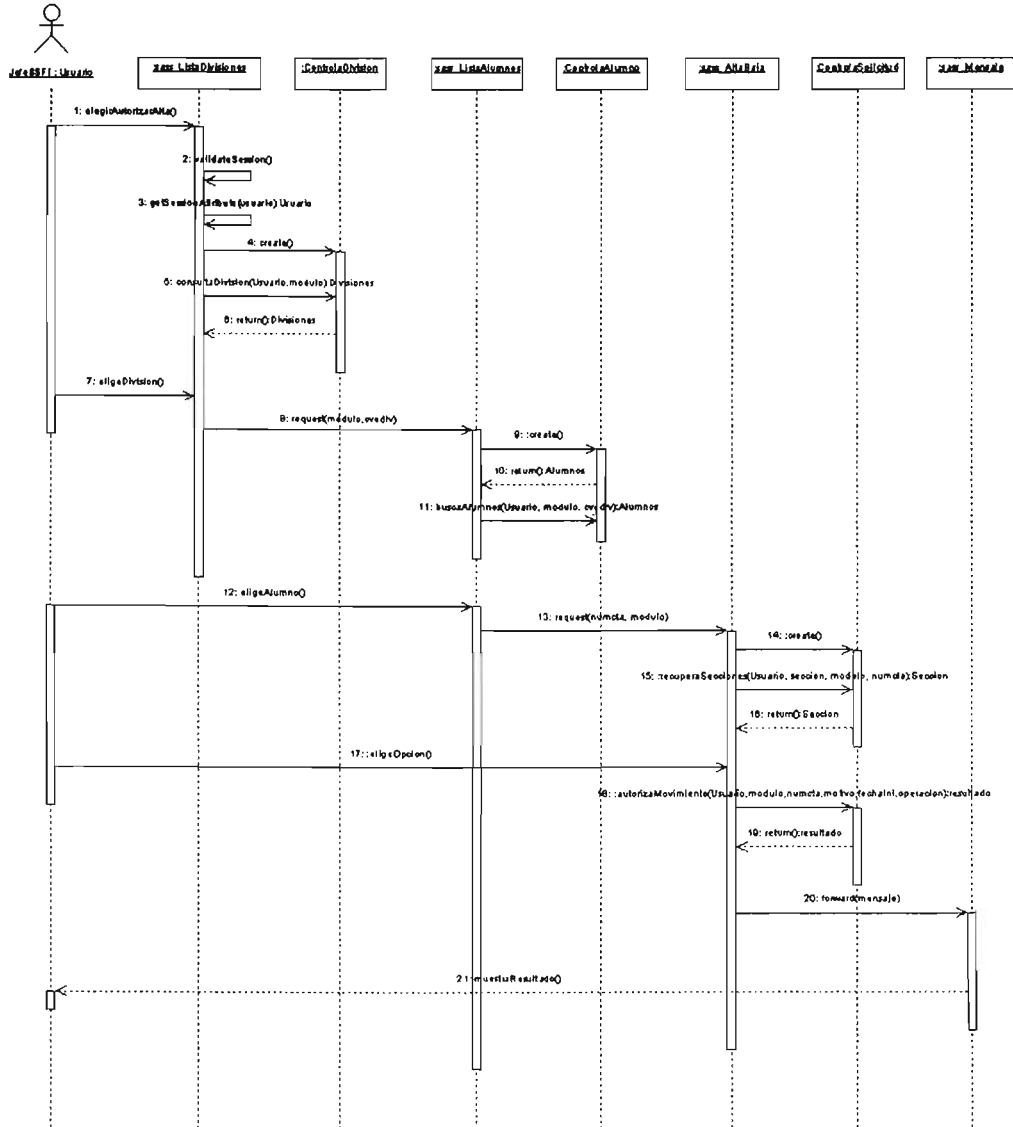


Figura 4.8 Diagrama de Secuencias. Caso de Uso SASS04A.

Clave: SASS05PB.

Caso de uso: Realizar Prebaja de un Alumno en el Sistema de Administración del SS.

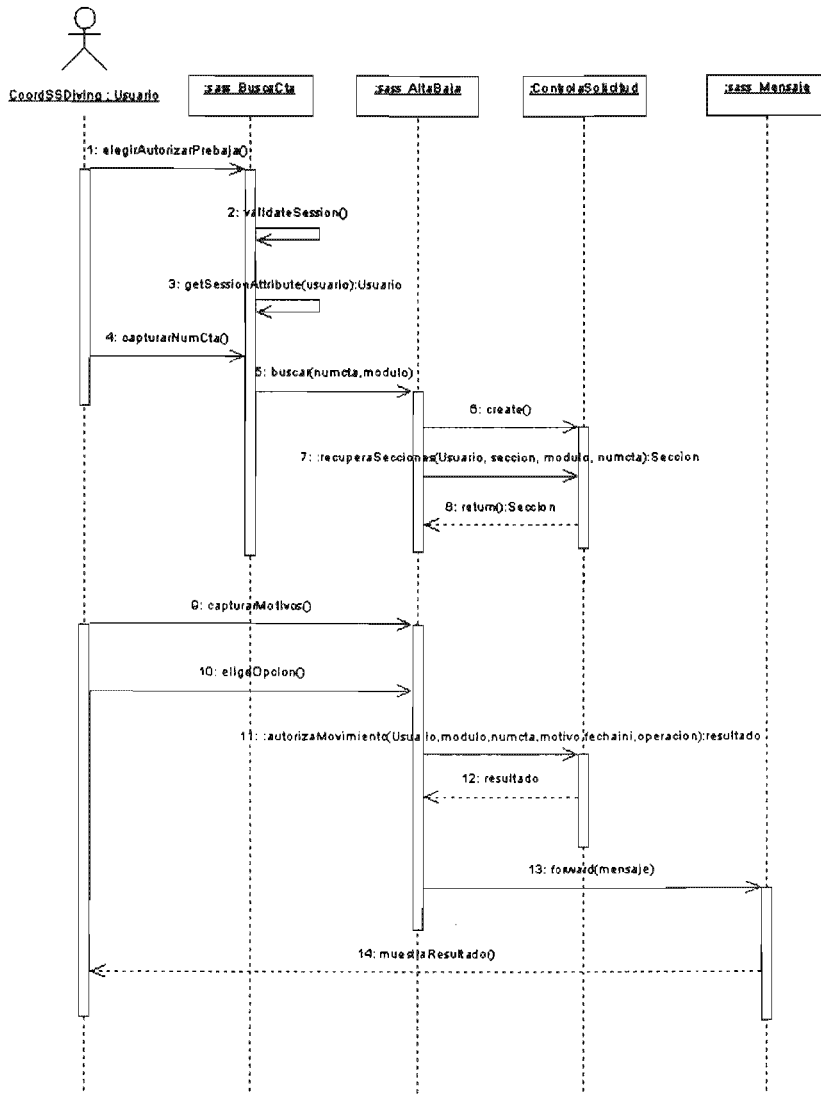


Figura 4.9 Diagrama de Secuencias. Caso de Uso SASS05PB.

Clave: SASS06B.

Caso de uso: Autorizar Baja de un Alumno en el Sistema de Administración de SS.

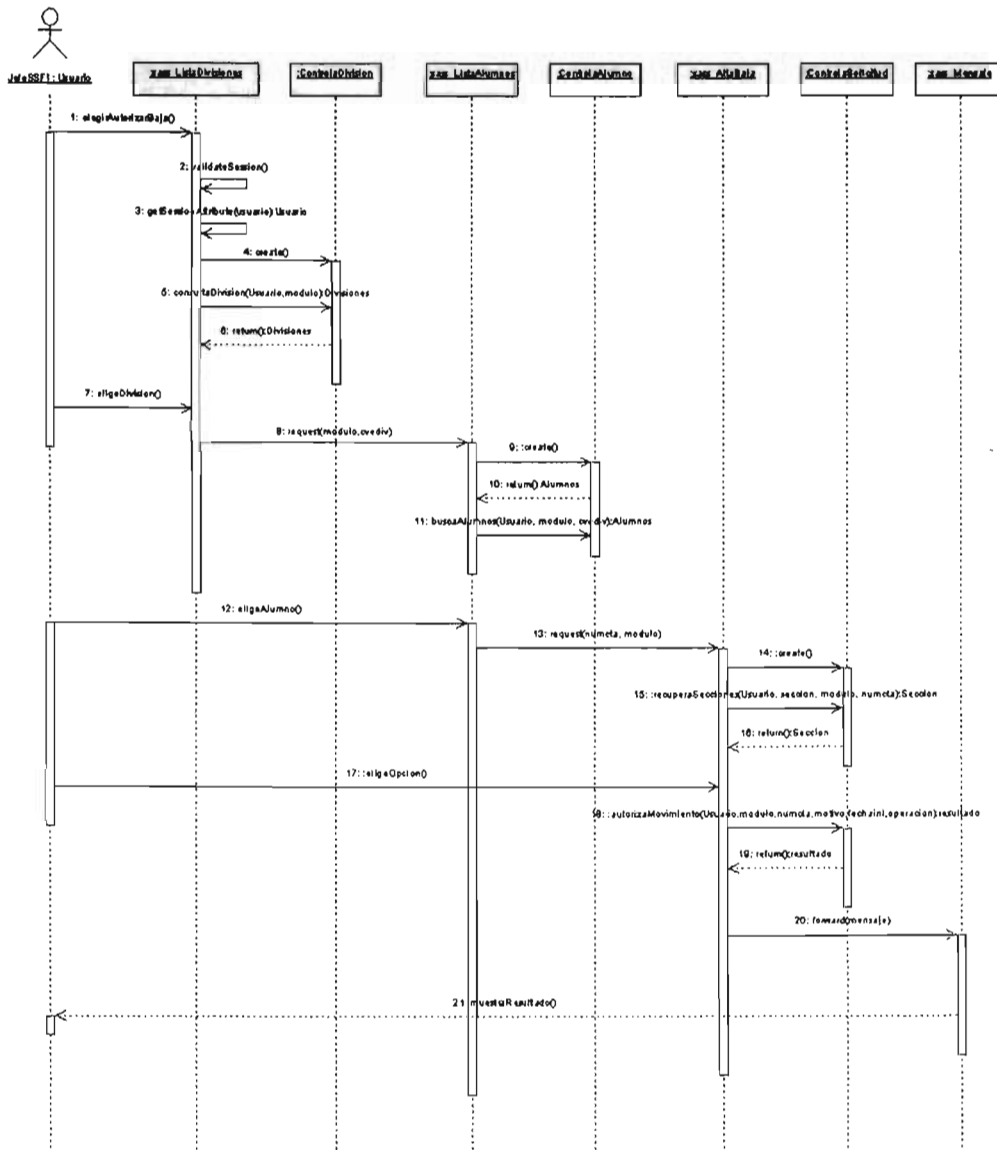


Figura 4.10 Diagrama de Secuencias. Caso de Uso SASS06B.

Clave: SASS07IC.

Caso de uso: Ingresar Comentarios de un Alumno en el Sistema de Administración del SS.

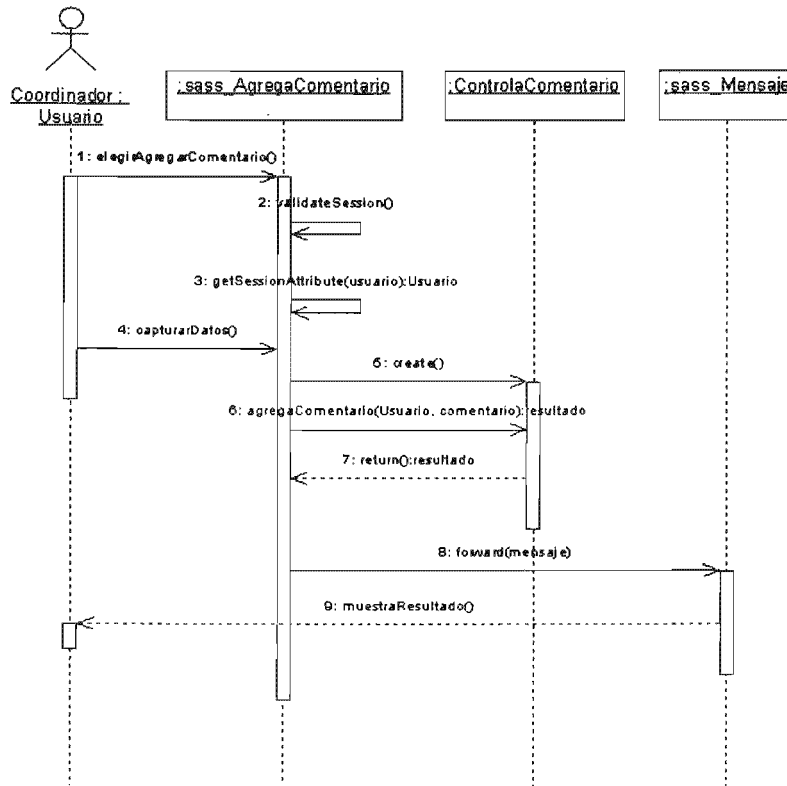


Figura 4.11 Diagrama de Secuencias. Caso de Uso SASS07IC.

Clave: SASS08GE.

Caso de uso: Generar Estadísticas.

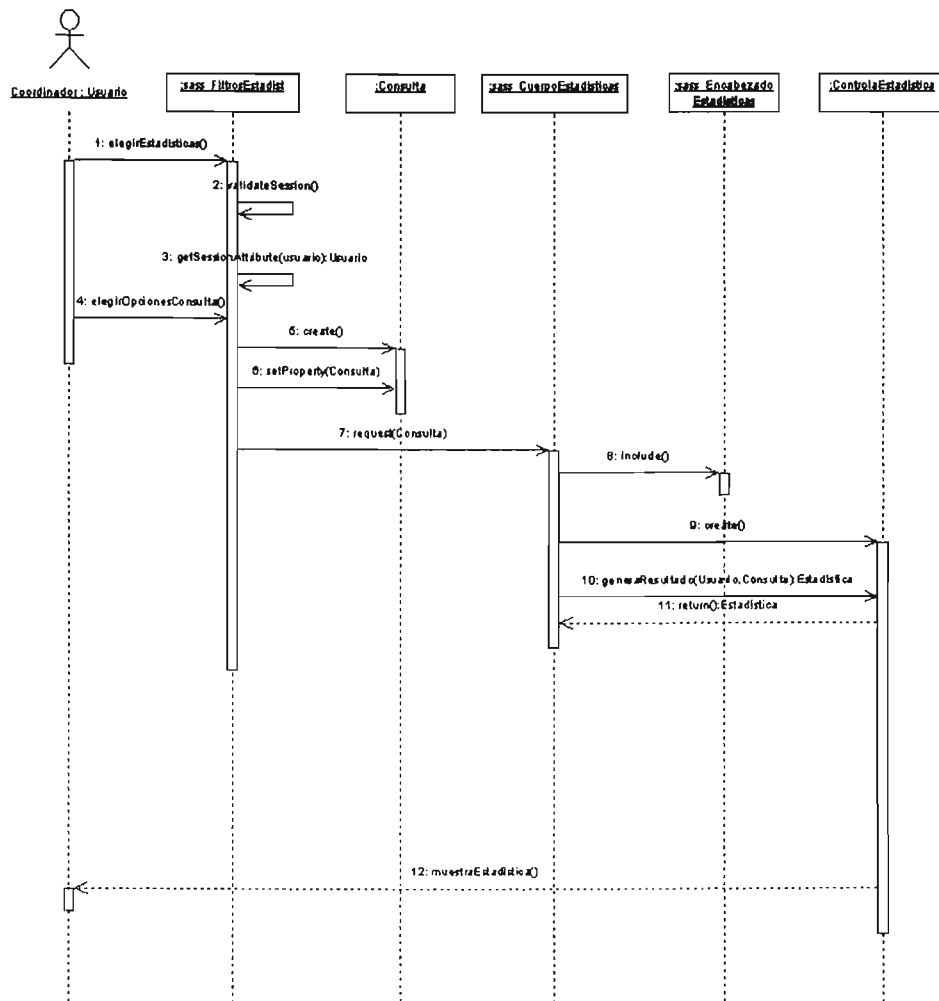


Figura 4.12 Diagrama de Secuencias. Caso de Uso SASS08GE.

Clave: SASS09GR.

Caso de uso: Generar Reportes.

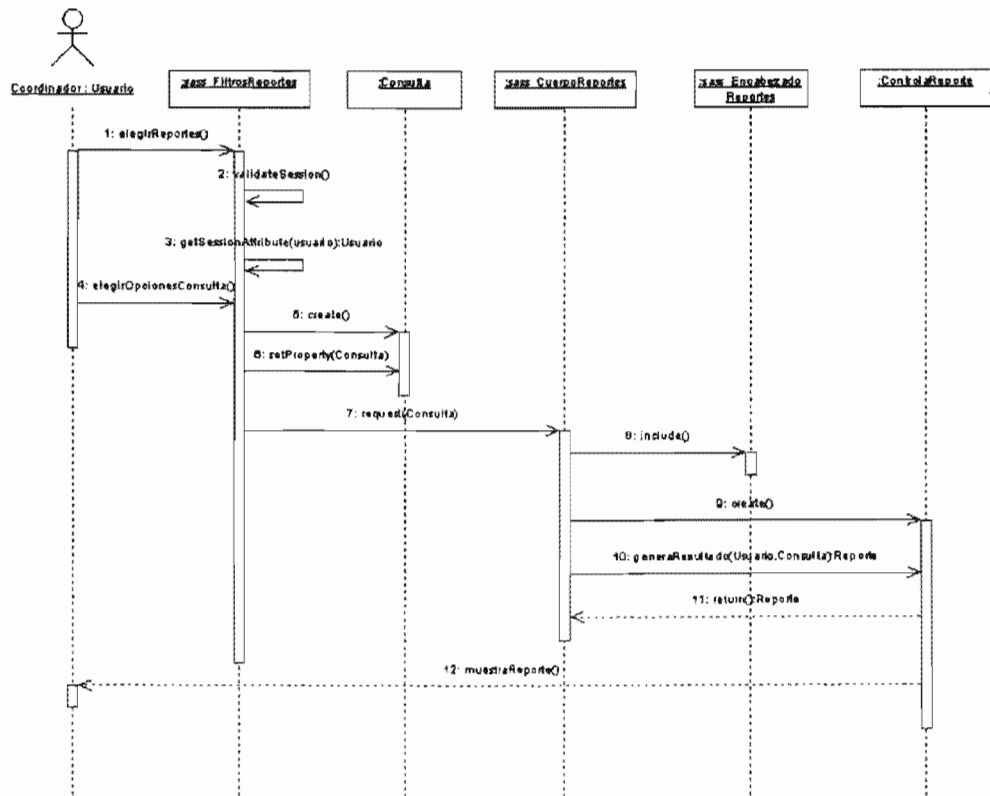


Figura 4.13 Diagrama de Secuencias. Caso de Uso SASS09GR.

Clave: SASS10AD.

Caso de uso: Actualizar Datos Personales del Alumno.

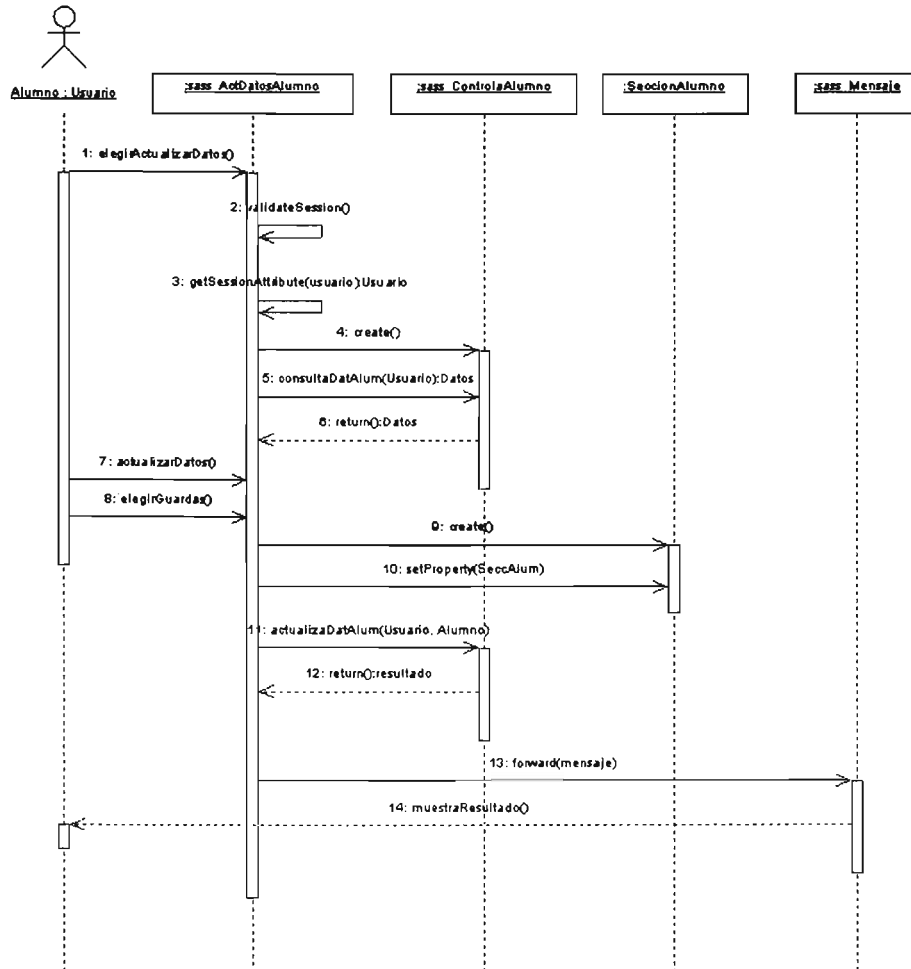


Figura 4.14 Diagrama de Secuencias. Caso de Uso SASS10AD.

Clave: SASS11RC.

Caso de uso: Reiniciar Contraseña del Alumno.

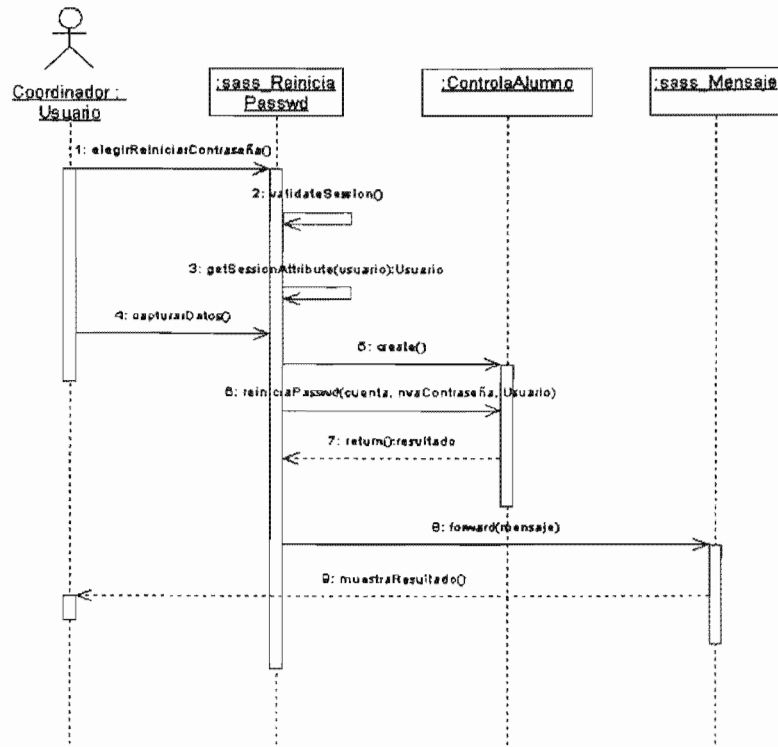


Figura 4.15 Diagrama de Secuencias. Caso de Uso SASS11RC.

Clave: SASS12CC.

Caso de uso: Cambiar Contraseña del Usuario.

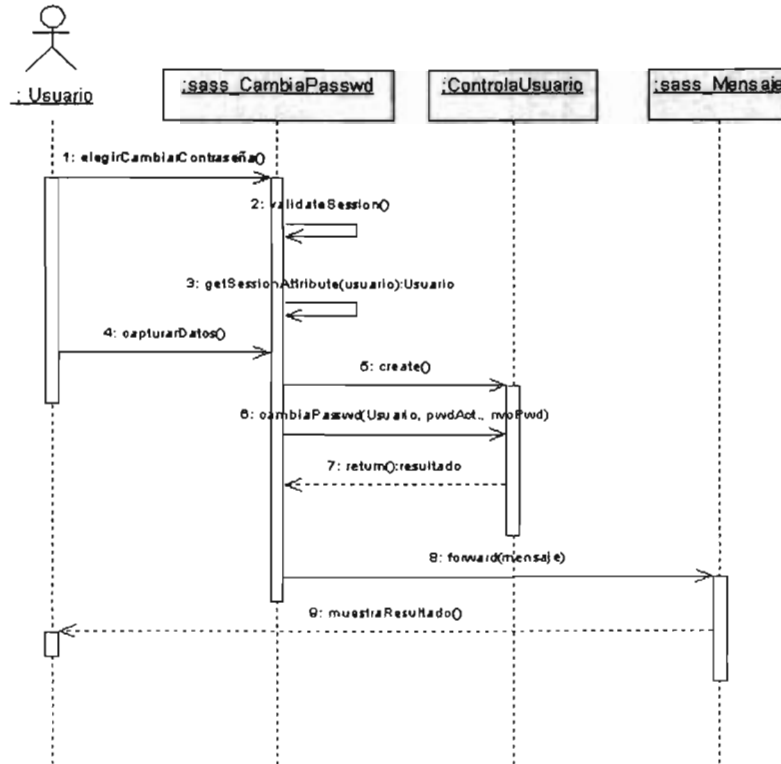


Figura 4.16 Diagrama de Secuencias. Caso de Uso SASS12CC.

Clave: SASS13T.

Caso de uso: Autorizar Terminación de un Alumno en el Sistema de Administración del SS.

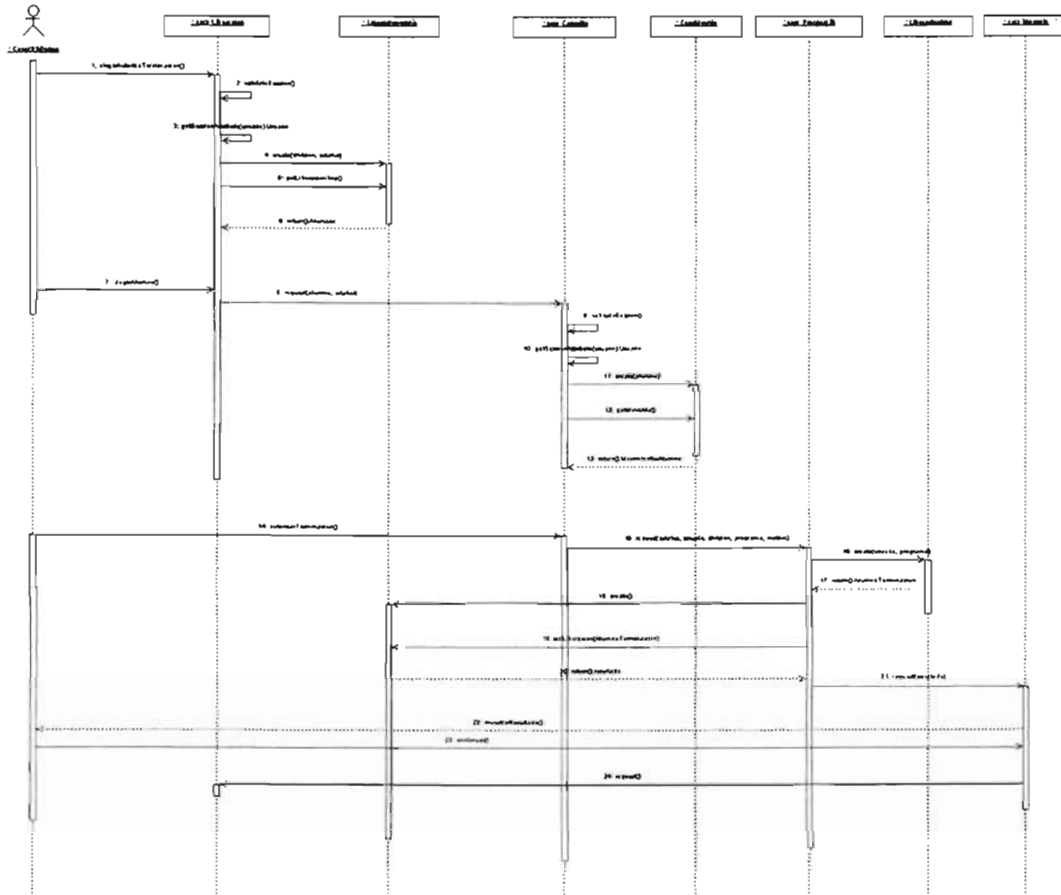


Figura 4.17 Diagrama de Secuencias. Caso de Uso SASS13T.

Clave: SASS14L.

Caso de uso: Autorizar Liberación de un Alumno en el Sistema de Administración del SS.

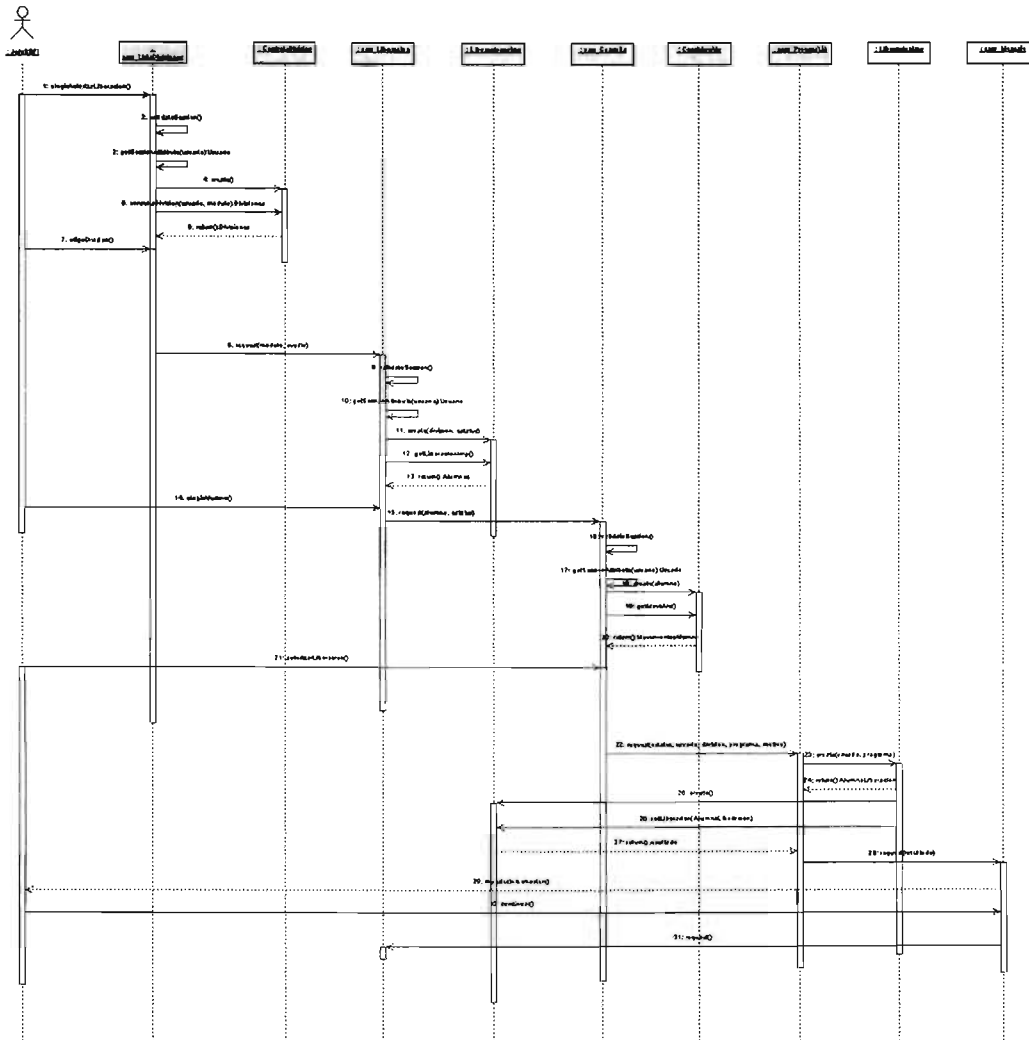


Figura 4.18 Diagrama de Secuencias. Caso de Uso SASS14L.

Clave: SASS15CA.

Caso de uso: Consultar Avance de un Alumno en el Sistema de Administración del SS.

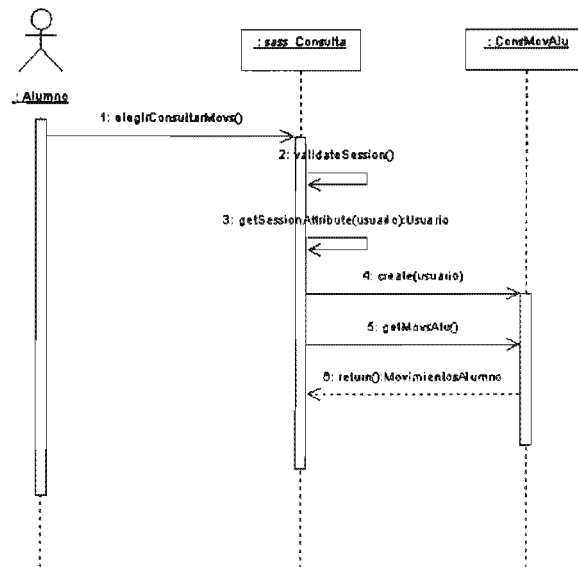


Figura 4.19 Diagrama de Secuencias. Caso de Uso SASS15CA.

Clave: SASS16CM.

Caso de uso: Consultar Mensajes en el Sistema de Administración del SS.

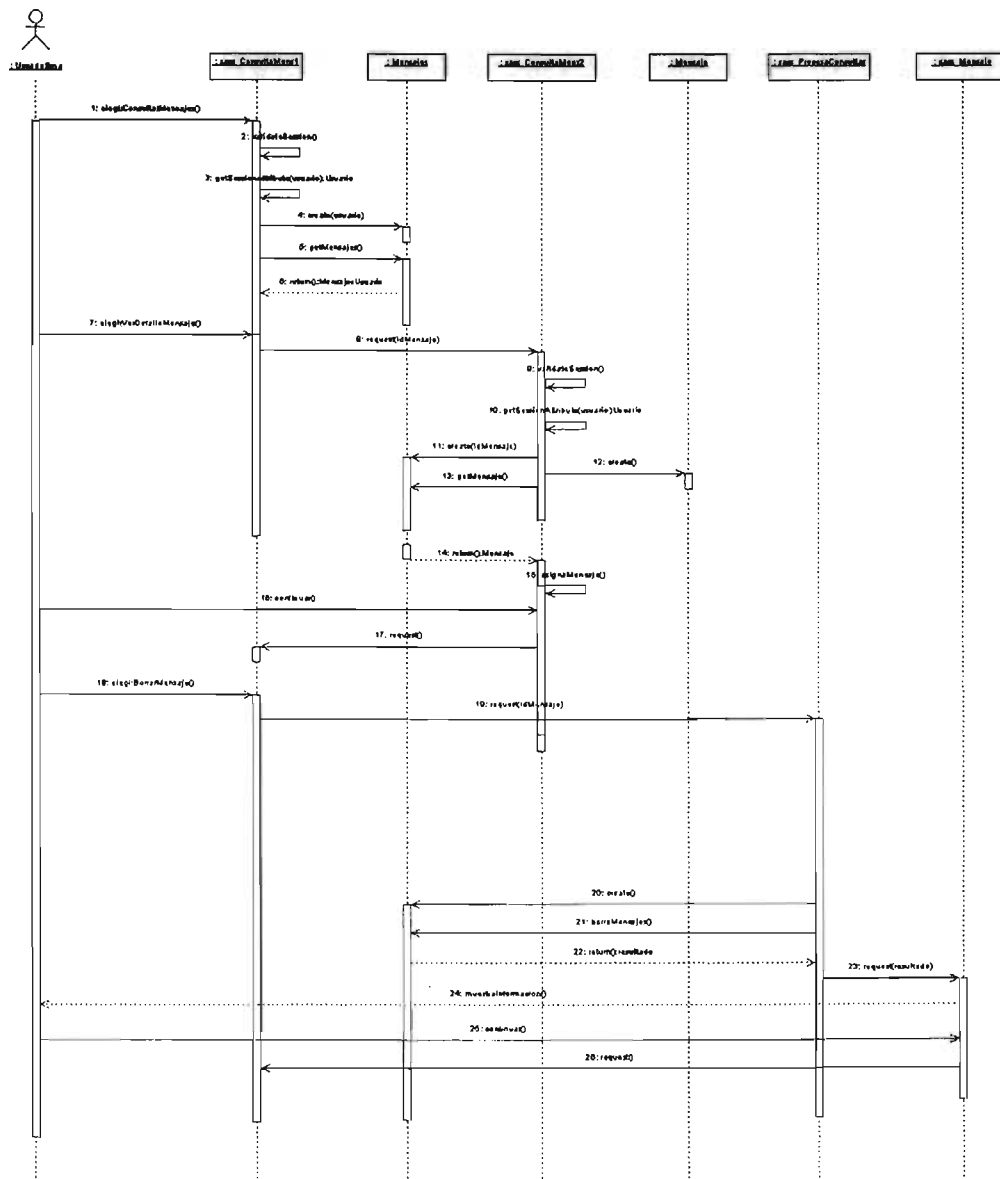


Figura 4.20 Diagrama de Secuencias. Caso de Uso SASS16CM.

Clave: SASS17AI.

Caso de uso: Actualizar el Estado de los Informes Bimestrales de un Alumno en el Sistema de Administración del SS.

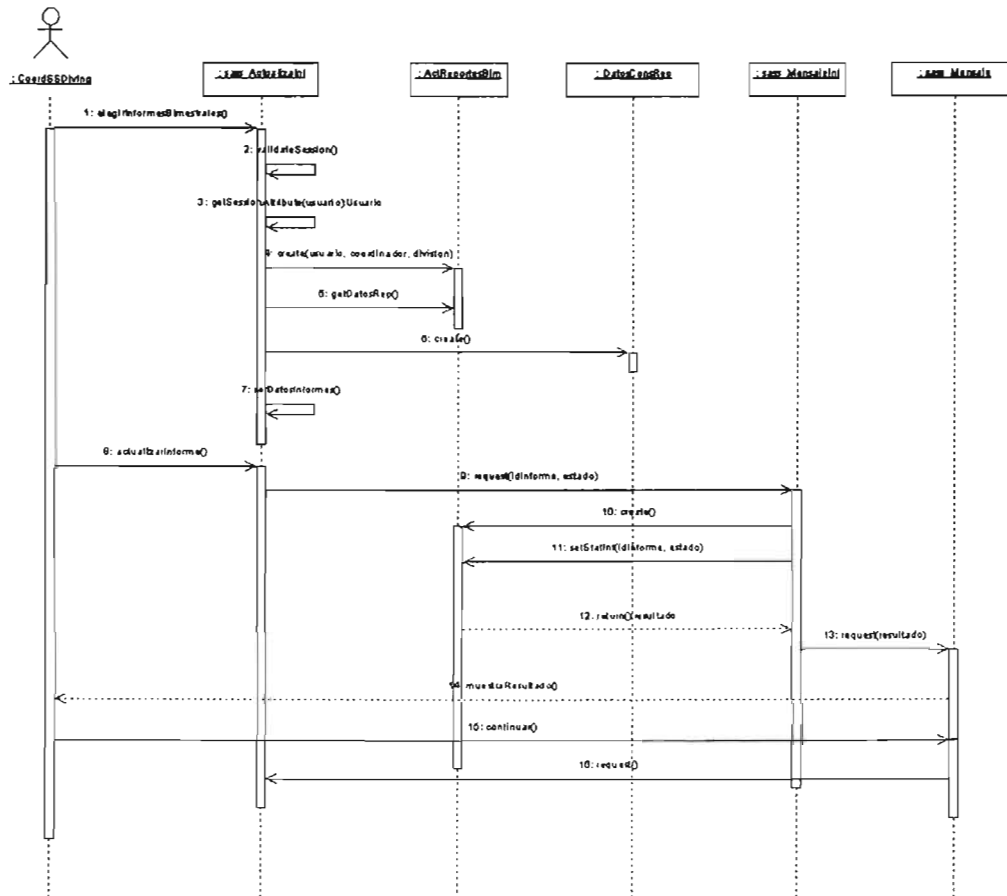


Figura 4.21 Diagrama de Secuencias. Caso de Uso SASS17AI.

Clave: SASS18AR.

Caso de uso: Asociar Nuevo Responsable de Programa de SS a un Alumno.

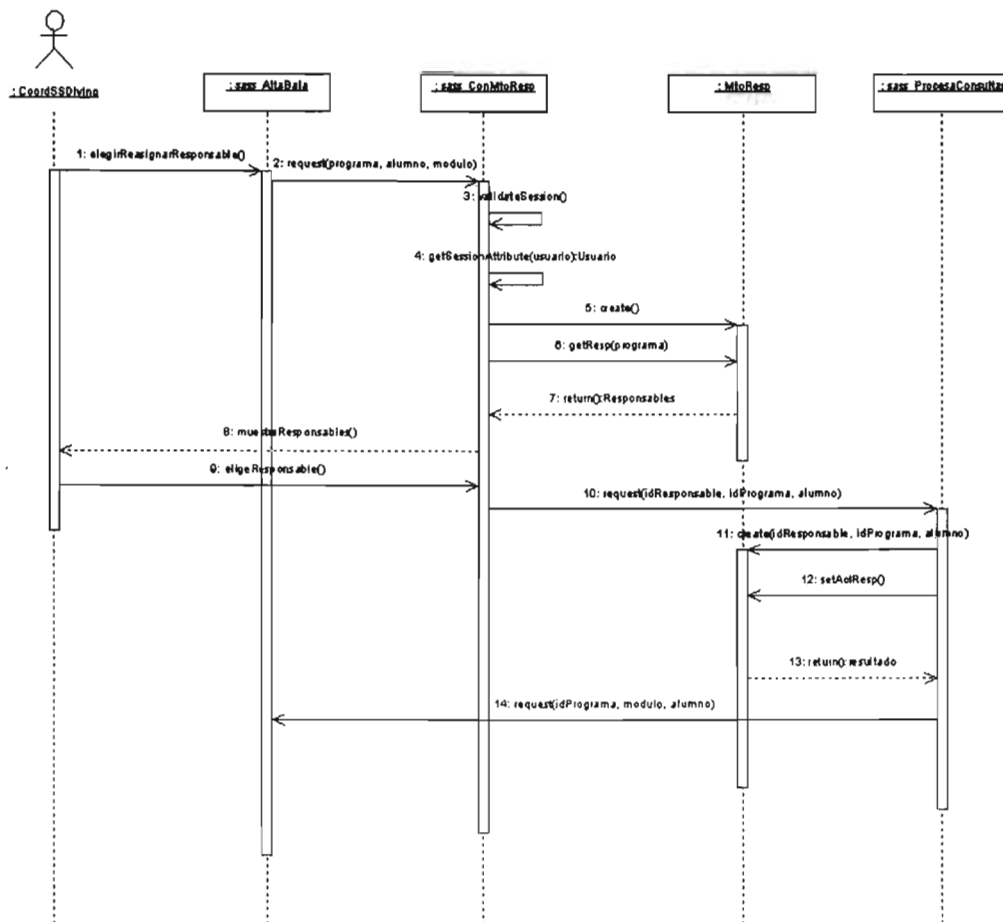


Figura 4.22 Diagrama de Secuencias. Caso de Uso SASS18AR.

Clave: SASS19CL.

Caso de uso: Descargar Datos para la Carta de Liberación.

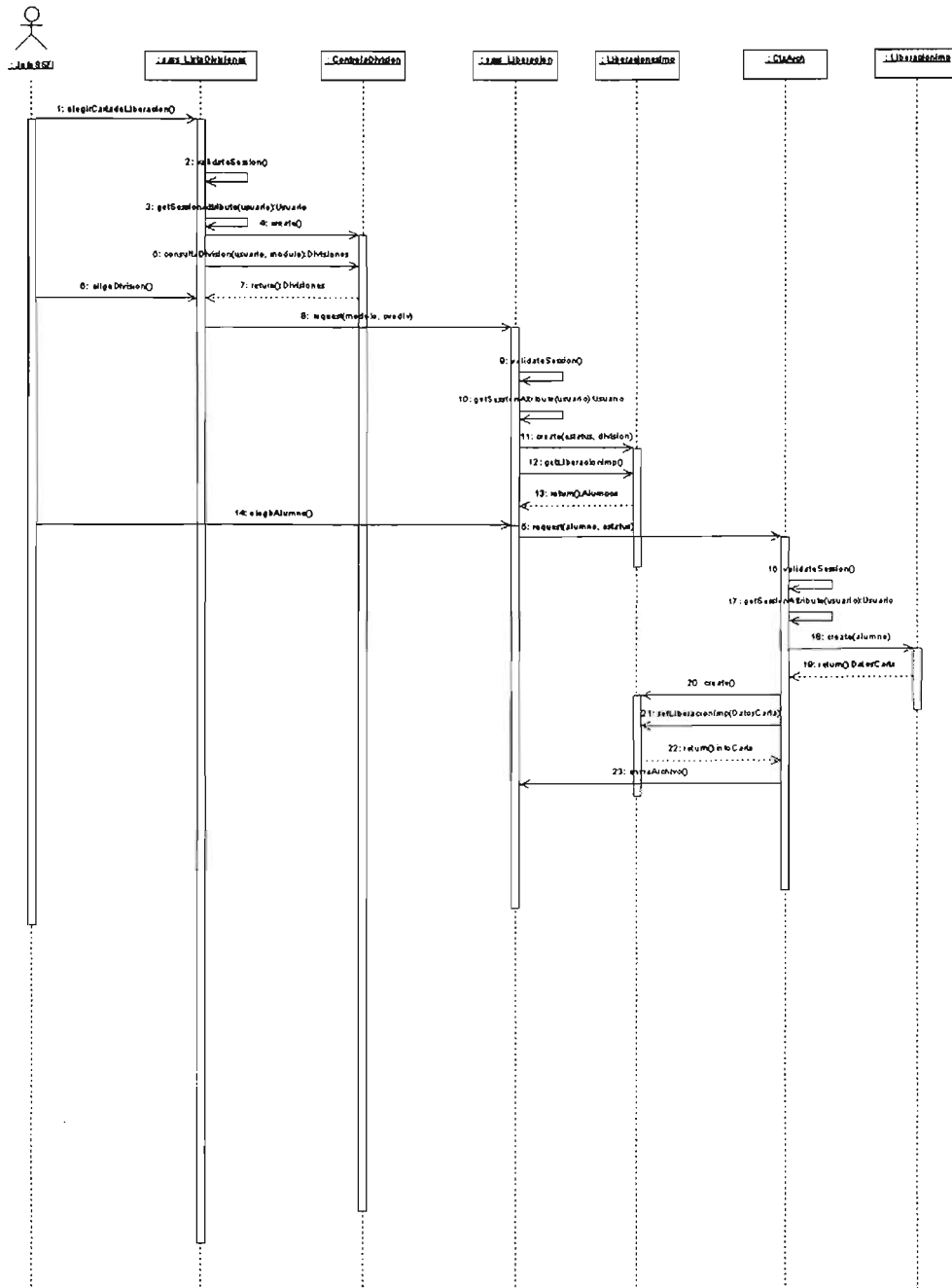


Figura 4.23 Diagrama de Secuencias. Caso de Uso SASS19CL.

Clave: SASS20MC.

Caso de uso: Dar mantenimiento de Cuentas de Coordinadores de SS.

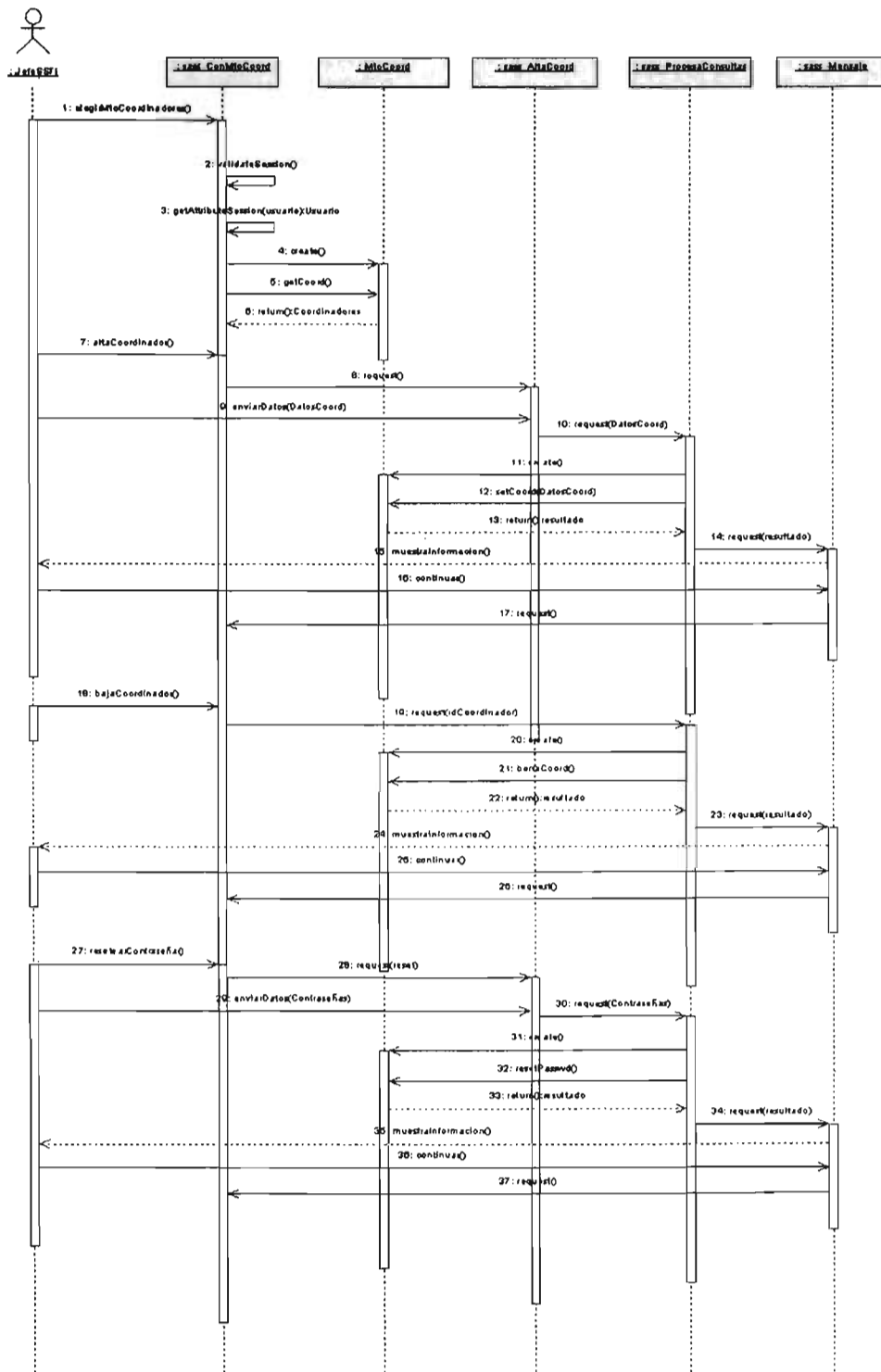


Figura 4.24 Diagrama de Secuencias. Caso de Uso SASS20MC.

Clave: SASS21GA.

Caso de uso: Generar Archivos con los Programas de SS Correspondientes a la Facultad de Ingeniería.

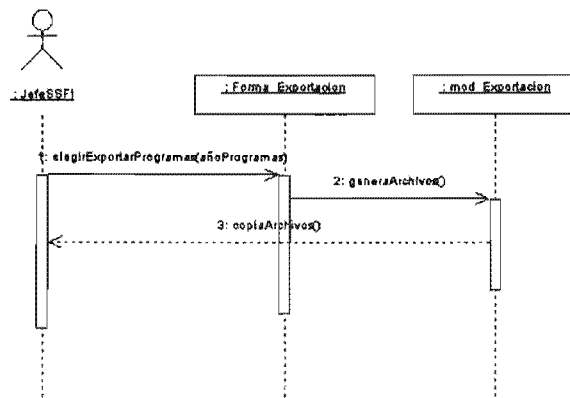


Figura 4.25 Diagrama de Secuencias. Caso de Uso SASS21GA.

4.2.5. Diagrama E/R

Con base en el análisis del dominio del sistema y en la identificación de las entidades que intervienen en el contexto del problema, se diseñó la base de datos cuyo diagrama Entidad/Relación se muestra a continuación.

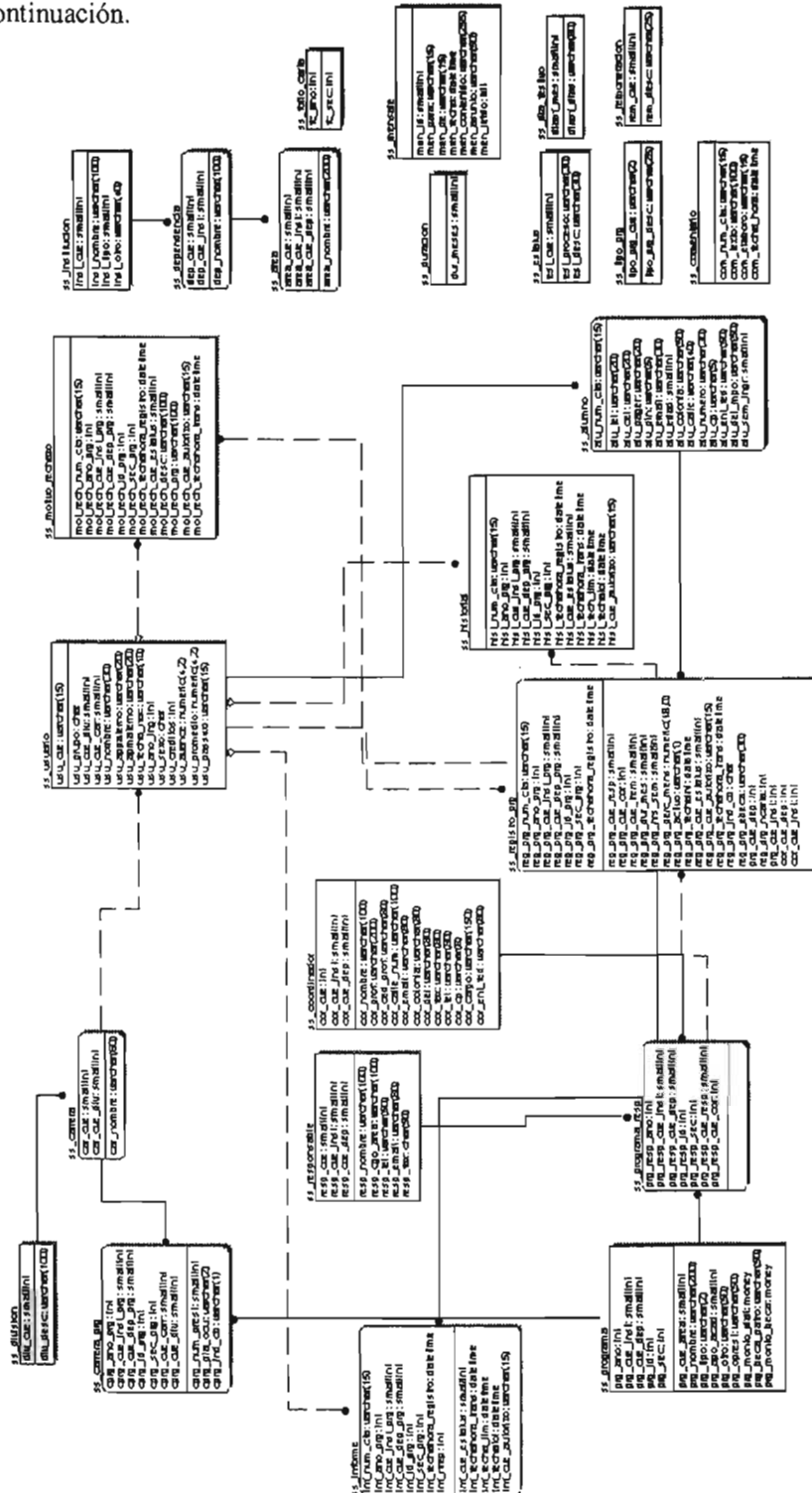


Figura 4.26 Diagrama E/R de la Base de Datos del SASS.

4.2.6. Diccionario de Datos

4.2.6.1. Tablas de Trabajo

SS_ALUMNO

Tabla que contiene la información referente a los datos personales del alumno.

Campo	Tipo de Dato	Descripción
alu_num_cta(PK, FK)	varchar(15)	Llave primaria que identifica a un alumno. Llave foránea de la tabla SS_USUARIO, que identifica a un alumno.
alu_tel	varchar(20)	Número telefónico del Alumno.
alu_cel	varchar(20)	Número celular del Alumno.
alu_pager	varchar(20)	Número de pager del Alumno
alu_pin	varchar(5)	Número de pin del pager del Alumno
alu_email	varchar(30)	Dirección de correo electrónico del Alumno.
alu_edad	smallint	Edad del Alumno.
alu_colonia	varchar(50)	Colonia del domicilio del Alumno.
alu_calle	varchar(40)	Calle del domicilio del Alumno.
alu_numero	varchar(30)	Número del domicilio del Alumno.
alu_cp	varchar(5)	Código postal del domicilio del Alumno.
alu_ent_fed	varchar(50)	Entidad federativa del domicilio del Alumno.
alu_del_mpo	varchar(50)	Delegación del domicilio del Alumno.
alu_sem_ingr	smallint	Semestre de ingreso del Alumno a la Facultad.

SS_HISTORIAL

Tabla que contiene la bitácora de los estatus por los que ha pasado un alumno durante el ejercicio de su servicio social.

Campo	Tipo de Dato	Descripción
hist_num_cta(FK)	varchar(15)	Llave foránea de la tabla SS_REGISTRO_PRG que identifica los movimientos de un registro.
hist_ano_prg(FK)	int	
hist_cve_inst_prg(FK)	smallint	
hist_cve_dep_prg(FK)	smallint	
hist_id_prg(FK)	int	
hist_sec_prg(FK)	int	
hist_fechahora_registro(FK)	datetime	
hist_cve_estatus	smallint	Clave del estatus asignado al Alumno en el movimiento en cuestión.
hist_fechahora_trans	datetime	Fecha y hora del movimiento o transacción.
hist_fech_lim	datetime	Fecha inicial del periodo de entrega de documentos correspondientes a los procesos del alumno sin penalización.
hist_fechatol	datetime	Fecha limite del periodo de entrega de documentos correspondientes a los procesos del alumno sin penalización.
hist_cve_autorizo(FK)	varchar(15)	Llave foránea de la tabla SS_USUARIO que identifica quien autorizo el movimiento.

SS_INFORME

Tabla que contiene los estatus de cada uno de los informes entregados por el Alumno.

Campo	Tipo de Dato	Descripción	
inf_num_cta(PK, FK)	varchar(15)	Llave primaria compuesta que asocia la clave de una dependencia a la clave de un registro con el número de reporte correspondiente.	Llave foránea de la tabla SS_REGISTRO_PRG que identifica los informes para el registro de un alumno
inf_ano_prg(PK, FK)	int		
inf_cve_inst_prg(PK, FK)	smallint		
inf_cve_dep_prg(PK; FK)	smallint		
inf_id_prg(PK, FK)	int		
inf_sec_prg(PK, FK)	int		
inf_fechahora_registro(PK, FK)	datetime		
inf_nrep(PK)	int	Indica el número de reporte.	
inf_cve_estatus	smallint	Clave del estatus asignado al informe.	
inf_fechahora_trans	datetime	Fecha y hora del movimiento o transacción.	

Inf_fecha_lim	datetime	Fecha límite para entrega del reporte que debe ser cumplida para que el Alumno no sea acreedor al corrimiento de fechas de sus próximos reportes.
Inf_fecha_tol	datetime	Fecha de tolerancia que debe ser cumplida por el Alumno para que este no se haga acreedor a una baja definitiva.
Inf_cve_autorizo(FK)	varchar(15)	Llave foránea de la tabla SS_USUARIO que identifica quien autorizó el informe.

SS_MENSAJE

Tabla que contiene los mensajes que se envían entre los coordinadores o de coordinadores a alumnos para mantenerse informados sobre rechazos de algún estado del alumno durante el ejercicio de su servicio social.

Campo	Tipo de Dato	Descripción
men_id(PK)	smallint	Llave primaria que permite identificar un mensaje.
men_para	varchar(15)	Clave del coordinador o alumno a quien va dirigido el mensaje.
men_de	varchar(15)	Clave del coordinador que está enviando el mensaje.
men_fecha	datetime	Fecha en que se envía el mensaje.
men_contenido	varchar(255)	Cuerpo del mensaje.
men_asunto	varchar(50)	Título del mensaje.
men_leido	Bit	Bandera que indica si el mensaje ya ha sido leído.

SS_MOTIVO_RECHAZO

Tabla que contiene los motivos de rechazo por proceso del alumno durante el ejercicio de su servicio social, incluyendo bajas de alumnos.

Campo	Tipo de Dato	Descripción
mot_rech_num_cta(PK, FK)	varchar(15)	Llave foránea de la tabla SS_REGISTRO_PRG que identifica los movimientos de rechazo de un registro.
mot_rech_ano_prg(PK, FK)	int	
mot_rech_cve_inst_prg(PK, FK)	smallint	
mot_rech_cve_dep_prg(PK, FK)	smallint	
mot_rech_id_prg(PK, FK)	int	
mot_rech_sec_prg(PK, FK)	int	
mot_rech_fechahora_registro(PK, FK)	datetime	
mot_rech_cve_estatus	smallint	Clave del estatus asignado al movimiento rechazado del Alumno.

mot_rech_desc	varchar(100)	Motivo de Rechazo del proceso.
mot_rech_prg	varchar(100)	Rechazo del programa elegido por el alumno.
mot_rech_cve_autorizo(FK)	Varchar(15)	Llave foránea de la tabla SS_USUARIO que identifica quien autorizo el rechazo del movimiento.
mot_rech_fecha_hora_trans	Datetime	Fecha y hora en que se realizó el rechazo.

SS_REGISTRO_PRG

Tabla que contiene los datos del registro del Alumno a un programa de SS.

Campo	Tipo de Dato	Descripción
reg_prg_fecha_hora_registro(PK)	datetime	Fecha y hora del registro.
reg_prg_num_cta (PK, FK)	smallint	Llave primaria compuesta que asocia la clave de un alumno con la clave de un programa y la fecha en la que se realiza el registro.
reg_prg_ano_prg (PK, FK)	int	Llave foránea de la tabla SS_USUARIO que identifica a un alumno.
reg_prg_cve_inst_prg (PK, FK)	smallint	Llave foránea de la tabla SS_PROGRAMA_RESP, que indica en qué programa se inscribió el Alumno.
reg_prg_cve_dep_prg (PK, FK)	smallint	
reg_prg_id_prg (PK, FK)	int	
reg_prg_sec_prg (PK, FK)	int	
reg_prg_cve_resp(FK)	smallint	
reg_prg_cve_cor(FK)	int	
reg_prg_cve_trem	smallint	Clave del tipo de Remuneración del Programa.
reg_prg_dur_mes	smallint	Duración, en meses, del Programa de SS
reg_prg_hrs_sem	smallint	Horas a la semana en las que el Alumno ejercerá su SS.
reg_prg_perc_mens	numeric (18,0)	Monto de la percepción mensual, en caso de existir.
reg_prg_activo	varchar(1)	Indica el estatus de actividad en el que se encuentra el registro. A=Activo, P=Prebaja, B=Baja, C=Cambiar Datos.
reg_prg_fecha_ini	datetime	Fecha de Inicio del ejercicio de SS.
reg_prg_cve_estatus	smallint	Clave del estatus de avance durante el ejercicio del SS del Alumno.
reg_prg_cve_autorizo(FK)	smallint	Llave foránea de la tabla SS_USUARIO que identifica quien

		autorizo el movimiento.
reg_prg_fecha_hora_trans	datetime	Fecha y hora en que se realizó la transacción (Registro, Preacta, Alta, Prebaja, etc.)
reg_prg_ind_cb	char	Identificador para saber si el programa en el que se registró el Alumno es de Ciencias Básicas o no.
reg_prg_ebeca	varchar(30)	NO ME ACUERDO
reg_prg_ncarta	int	Número de carta de liberación de SS asignada al Alumno.

SS_COMENTARIO

En esta tabla se almacenan los comentarios o notas adicionales sobre los Alumnos que los Coordinadores ingresan en el sistema.

Campo	Tipo de Dato	Descripción
com_num_cta	varchar(15)	Número de cuenta del alumno al que se le agrega algún comentario.
com_fecha_hora	datetime	Fecha y hora en la que se editó el comentario.
com_elaboro	varchar(15)	Clave del Coordinador que elaboró el comentario.
com_texto	varchar(100)	Texto del comentario.

4.2.6.2. Catálogos

SS_AREA

Tabla que contiene los nombres de las áreas de las dependencias.

Campo	Tipo de Dato	Descripción
area_cve(PK)	smallint	Llave primaria
area_cve_inst(PK,FK)	smallint	compuesta que asocia la clave de un área a la clave de una dependencia.
area_cve_dep(PK,FK)	smallint	
area_nombre	varchar(200)	Nombre descriptivo del área.

SS_CARRERA

Tabla que contiene el catálogo de las carreras que se imparten en la Facultad de Ingeniería.

Campo	Tipo de Dato	Descripción
car_cve(PK)	smallint	Llave primaria compuesta que
car_cve_div(PK, FK)	smallint	asocia la clave de una carrera a la clave de una división.
car_nombre	varchar(60)	Nombre descriptivo de la carrera.

SS_CARRERA_PRG

Tabla que contiene la relación de los programas con las carreras de la Facultad de Ingeniería.

Campo	Tipo de Dato	Descripción
cprg_ano_prg(PK, FK)	int	Llave primaria compuesta que asocia la clave de un programa a la clave de una carrera.
cprg_cve_inst_prg(PK, FK)	smallint	
cprg_cve_dep_prg(PK, FK)	smallint	
cprg_id_prg (PK, FK)	int	
cprg_sec_prg (PK, FK)	int	
cprg_cve_carr(PK, FK)	smallint	Llave foránea de la tabla SS_PROGRAMA, que permite identificar al programa asociado a una carrera.
cprg_cve_div(PK, FK)	smallint	Llave foránea de la tabla SS_CARRERA, que permite identificar a la carrera asociada a un programa.
cprg_num_prest	smallint	Indica el número de alumnos de la carrera que se necesitan para prestar el servicio social en el programa asociado.
cprg_pla_ocu	varchar(2)	Indica el número de alumnos de la carrera que están actualmente prestando el servicio social en el programa asociado.
cprg_ind_cb	varchar(1)	Indicador para saber si el programa es o no de Ciencias Básicas.

SS_COORDINADOR

Tabla que contiene la información general de los coordinadores de las dependencias involucradas en los programas de servicio social.

Campo	Tipo de Dato	Descripción
cor_cve(PK)	int	

cor_cve_inst(PK)	smallint	Llave primaria compuesta que asocia la clave de un coordinador a la clave de una dependencia.
cor_cve_dep(PK)	smallint	
cor_nombre	varchar(100)	Nombre del coordinador.
cor_prof	varchar(200)	Profesión del coordinador.
cor_ced_prof	varchar(80)	Clave de la cédula profesional del coordinador.
cor_cargo	varchar(150)	Cargo del coordinador.
cor_email	varchar(80)	E-mail del coordinador.
cor_calle_num	varchar(100)	Calle y número donde se ubica el coordinador
cor_colonia	varchar(80)	Colonia donde se ubica el coordinador
cor_del	varchar(80)	Delegación o municipio en donde se ubica el coordinador
cor_ent_fed	varchar(80)	Entidad federativa donde se encuentra el coordinador
cor_cp	char(5)	Código postal donde se ubica el coordinador.
cor_tel	varchar(80)	Teléfonos donde se puede localizar al coordinador.
cor_fax	varchar(80)	Fax de la oficina del coordinador.

SS_DEPENDENCIA

Tabla que contiene el catálogo de dependencias por institución involucradas en los programas de servicio social.

Campo	Tipo de Dato	Descripción	
dep_cve(PK)	smallint	Llave primaria	Clave que identifica una dependencia.
dep_cve_inst(PK, FK)	smallint	compuesta que asocia la clave de una dependencia a la clave de una institución.	Llave foránea de la tabla SS_INSTITUCIÓN que permite relacionar una institución con una dependencia
dep_nombre	varchar(100)	Nombre de la Dependencia	

SS_DIA_FESTIVO

Tabla que contiene los días festivos dentro del ciclo escolar.

Campo	Tipo de Dato	Descripción
diaf_mes(PK)	smallint	Llave primaria que contiene un número que identifica un mes del año. (0-Enero...11-Diciembre)

diaf_dia	varchar(80)	Contiene los días festivos separados por comas.
----------	-------------	---

SS_DIVISION

Tabla que contiene el catálogo de divisiones que comprende la Facultad de Ingeniería.

Campo	Tipo de Dato	Descripción
div_cve(PK)	smallint	Llave primaria que identifica la clave de la División.
div_des	varchar(100)	Nombre de la División de la Facultad de Ingeniería.

SS_DURACION

Tabla que contiene el catálogo de duraciones en meses para la realización del servicio social.

Campo	Tipo de Dato	Descripción
dur_meses (PK)	smallint	Llave primaria que identifica la duración en meses válida para la prestación de SS.

SS_ESTATUS

Tabla que contiene el catalogo de estatus (combinación de un proceso con un estado) en los que se puede encontrar un alumno durante el ejercicio de su SS.

Campo	Tipo de Dato	Descripción
est_cve(PK)	smallint	Llave primaria que identifica el estatus en el que se encuentra el alumno durante la prestación del SS.
est_proceso	varchar(30)	Descripción del proceso asociado a la clave del estatus. Los procesos son: Registro, Prealta, Alta, Prebaja, Baja, Informe, Terminación, Liberación, Carta de Terminación.
est_desc	varchar(30)	Descripción del estado asignado al proceso del alumno.

Valores del catalogo:

Est_cve	Est_proceso	Est_desc
0	Registro	
1	Prealta	Rechazada
2	Prealta	Aceptada
3	Alta	Rechazada
4	Alta	Aceptada
5	Prebaja	
6	Baja	
7	Prebaja	Revertida
8	Baja	Revertida
9	Informe	Aceptado
10	Informe	Rechazado
11	Informe	No Recibido
12	Terminación	Aceptada
13	Liberación	Aceptada
14	Carta de Liberación	Entregada
15	Liberación	Estimada

SS_FOLIO_CARTA

Tabla que contiene la relación de l número de folios asociados a la impresión de cartas de Liberación en el año.

Campo	Tipo de Dato	Descripción
fc_ano(PK)	int	Llave primaria que identifica el año en que se imprimen cartas de liberación.
fc_sec	int	Ultimo folio impreso en el año asociado.

SS_INSTITUCIÓN

Tabla que contiene el catálogo de instituciones involucradas en los programas de servicio social.

Campo	Tipo de Dato	Descripción
inst_cve(PK)	smallint	Llave primaria que identifica la clave de una institución.
inst_nombre	varchar(100)	Nombre de la Institución
inst_tipo	smallint	Indica si la institución es de tipo UNAM (1), Sector Público (2,3), Sector Social (4), Organismo autónomo (6) u otra (5).
inst_otro	varchar(40)	Cuando el tipo de institución es otro aquí se describe el tipo.

SS_PROGRAMA

Tabla que contiene el catálogo de programas de SS.

Campo	Tipo de Dato	Descripción
prg_ano(PK)	int	Llave primaria compuesta que identifica un programa de SS.
prg_cve_inst(PK,)	smallint	
prg_cve_dep(PK)	smallint	
prg_id(PK)	int	
prg_sec(PK)	int	
prg_cve_area	smallint	Área del Programa.
prg_nombre	varchar(200)	Nombre del Programa
prg_tipo	varchar(2)	Indicador del tipo de programa (investigación, académico, servicios, etc.).
prg_apo_acad	smallint	Cuando el programa presta apoyo académico al alumno, los cuales pueden ser: Cursos de capacitación, Cursos complementarios, Desarrollo de Tesis u otro. Cuando el programa presta apoyo económico, los cuales pueden ser: viáticos o beca. Cuando el programa da otras prestaciones, las cuales pueden ser: Servicio de Comedor, Atención Médica, Impresión de Tesis u otro.
prg_otro	varchar(50)	Cuando el apoyo académico es otro, se especifica cual es.
prg_oprest	varchar(50)	Cuando el apoyo de otra prestación es otro, se especifica cual es.
prg_monto_viat	money	Cuando el apoyo económico es viáticos, se especifica el monto.
prg_beca_patro	varchar(50)	Cuando el apoyo es económico, se especifica el patrocinador.
prg_monto_beca	Money	Cuando el apoyo económico es beca, se especifica el monto.

SS_PROGRAMA_RESP

Tabla que contiene el catálogo de programas de SS y su respectivo responsable.

Campo	Tipo de Dato	Descripción	
prg_resp_ano(PK, FK)	int	Llave primaria compuesta	Llave foránea de la tabla SS_PROGRAMA que identifica un programa de servicio social
prg_resp_id(PK, FK)	int		
prg_resp_sec(PK, FK)	int		

prg_resp_cve_resp(PK, FK)	smallint	que asocia la clave de un programa con la clave de un responsable y un coordinador.	Llave foránea de la tabla SS_RESPONSABLE que identifica un responsable.	
prg_resp_cve_inst(PK, FK)	smallint			
prg_resp_cve_dep(PK, FK)	smallint			
prg_resp_cve_cor(PK, FK)	int			Llave foránea de la tabla SS_COORDINADOR que identifica un coordinador.

SS_REMUNERACION

En esta tabla se almacena el catálogo de remuneración u otras ayudas que otorga el Programa de SS.

Campo	Tipo de Dato	Descripción
rem_cve(PK)	smallint	Llave primaria que identifica a un tipo de remuneración.
rem_desc	varchar(25)	Descripción de la remuneración

Valores del catalogo:

Rem_cve	Rem_desc
1	Sueldo
2	Honorarios
3	Ayuda Económica
4	Beca
5	Otro
6	No remunerado

SS_RESPONSABLE

Tabla que contiene el catálogo de los responsables de los programas de servicio social.

Campo	Tipo de Dato	Descripción
resp_cve(PK)	smallint	Llave primaria compuesta que asocia la clave de un responsable a la clave de una dependencia.
resp_cve_inst(PK)	smallint	
resp_cve_dep(PK)	smallint	
resp_nombre	varchar(100)	Nombre del Jefe inmediato responsable del Programa.
resp_cgo_area	varchar(100)	Cargo del Jefe Inmediato responsable del programa.
resp_tel	varchar(50)	Teléfono del Jefe inmediato

		responsable del programa.
resp_fax	varchar(50)	Fax del Jefe Inmediato responsable del programa.
resp_email	varchar(80)	e-mail del Jefe inmediato responsable del programa.

SS_TIPO_PRG

Tabla que contiene los tipos en los que se clasifican los Programas de SS.

Campo	Tipo de Dato	Descripción
tipo_prg_cve(PK)	varchar(2)	Llave primaria que identifica una clave para un tipo de programa.
tipo_prg_desc	varchar(25)	Descripción del tipo de Programa.

Valores del catalogo:

Tipo_prg_cve	Tipo_prg_desc
1	Investigación
2	Academico-Administrativo
3	Servicios
4	Admon. Publica
5	Docencia

SS_USUARIO

Tabla que contiene la información de los usuarios del sistema (Alumno, Coordinador, Administrador) de la Facultad de Ingeniería.

Campo	Tipo de Dato	Descripción
usu_cve(PK)	varchar(15)	Llave primaria que identifica una clave para un coordinador.
usu_grupo	char(1)	Clave del grupo al que pertenece el usuario.
usu_cve_div(FK)	Smallint	Llave foránea de la tabla SS_CARRERA, que permite identificar a la carrera asociada al alumno.
usu_cve_car(FK)	Smallint	
usu_nombre	varchar(30)	Nombre del Usuario.
usu_appaterno	varchar(20)	Apellido paterno del Usuario.
usu_apmaterno	varchar(20)	Apellido materno del Usuario.
usu_fecha_nac	varchar(10)	Fecha de nacimiento del usuario.
usu_ano_ing	Int	Año de ingreso a la Facultad. Este campo tiene un valor sólo si el usuario es un Alumno.

usu_sexo	char(1)	Indicador del sexo del usuario. F=Femenino, M=Masculino.
usu_creditos	Int	Créditos de la carrera acumulados Este campo tiene un valor sólo en el caso en que se trate de un Alumno.
usu_avance	numeric(4,2)	Porcentaje de avance en la carrera. Este campo tiene un valor sólo en el caso en que se trate de un Alumno.
usu_promedio	numeric(4,2)	Promedio de calificaciones. Este campo tiene un valor sólo en el caso en que se trate de un Alumno.
usu_passwd	varchar(15)	Contraseña para el acceso de los usuarios al sistema.

4.2.7. Lista de Personas

En la fase de diseño, el equipo de trabajo está conformado por:

Proveedor

- ◆ Galicia González Gpe. Miriam
- ◆ Vázquez Chávez Bruno David

Construcción del Sistema

CAPÍTULO 5

CONSTRUCCIÓN DEL SISTEMA

5.1 Análisis del Tiempo

Con la finalidad de identificar las actividades a realizar y el tiempo que se emplea en la fase de construcción del sistema, se elabora el análisis del tiempo, en el cual se especifica el tiempo que se asigna para la realización de cada una de estas actividades.

La elaboración de las siguientes tablas de asignación de tiempos se basa en la condición de que el equipo de trabajo de esta tesis está en posibilidades de invertir dos días a la semana, con un aproximado de 7 horas cada día.

5.1.1. Asignación de Tiempos para la Instalación del Ambiente de Desarrollo

Actividad	Horas	Semanas
Instalación del Sistema Operativo Red Hat Linux 7.0	10	0.714285714
Instalación de Java 2 SDK 1.4	4	0.285714286
Instalación del Motor de JSP y Servlet TomCat 1.4	6	0.428571429
Instalación del Manejador de Base de Datos Sybase 11.0	10	0.714285714
Total	30	2.142857143

5.1.2. Asignación de Tiempos para la Construcción de la Base de Datos

Actividad	Horas	Semanas
Elaboración de Scripts de la Estructura de la BD	14	1
Construcción de la BD (pruebas y correcciones)	7	0.5
Construcción de scripts para el Proceso de Carga de Programas de SS (DGOSE) a la Base de Datos del SASS.	28	2
Generar archivos con los programas de SS correspondientes a la FI (pruebas y correcciones)	20	1.428571
Cargar los Archivos Generados sobre los Programas de SS en el SASS (pruebas y correcciones)	20	1.428571
Pruebas y adecuaciones del proceso de Carga de Programas de SS de DGOSE a la BD deL SASS.	25	1.785714
Total	114	8.142857

5.1.3. Asignación de Tiempos para la Construcción de la Aplicación

Actividad	Horas	Semanas
Validar Usuario	15	1.071429
Registrar un Alumno en el SASS	28	2
Autorizar Prealta de un Alumno en el SASS	18	1.285714

Autorizar Alta de un Alumno SASS	9	0.642857
Autorizar Prebaja de un Alumno SASS	9	0.642857
Autorizar Baja de un Alumno SASS	9	0.642857
Ingresar Comentarios de un Alumno en el SASS	6	0.428571
Generar Estadísticas	45	3.214286
Generar Reportes	30	2.142857
Actualizar Datos Personales del Alumno	5	0.357143
Reiniciar Contraseña del Alumno	6	0.428571
Cambiar Contraseña del Usuario	5	0.357143
Autorizar Terminación de un Alumno en el SASS	21	1.5
Autorizar Liberación de un Alumno en el SASS	14	1
Actualizar el Estado de los Informes Bimestrales de un Alumno en el SASS	21	1.5
Consultar Avance de un Alumno en el SASS	15	1.071429
Consultar Mensajes en el SASS	15	1.071429
Descargar Datos para la Carta de Liberación	14	1
Dar mantenimiento de Cuentas de Coordinadores de Servicio Social	21	1.5
Asociar Nuevo Responsable de Programa de SS a un Alumno	15	1.071429
Proceso de Integración de Componentes	21	1.5
Total	342	24.42857

5.1.4. Asignación de Tiempos para el Diseño Gráfico de la Aplicación

Actividad	Horas	Semanas
Elaboración del logo del sistema	3	0.214286
Elaboración del fondo	2	0.142857
Definición de fuentes, tamaños y colores en la hoja de estilos	3	0.214286
Aplicación y revisión del formato en toda la aplicación	13	0.928571
Total	21	1.5

5.2 Fases de Construcción

El Sistema de Administración de Servicio Social es una aplicación construida bajo una arquitectura estricta de tres capas principales llamadas front-end, middleware y back-end. Sin embargo, con el fin de organizar los elementos del diseño orientado a objetos se distingue una subdivisión lógica de la capa de la lógica del negocio. Todo lo anterior se describe en las siguientes secciones.

5.2.1. Front-End o Capa de Presentación

El front-end o capa de presentación de la aplicación se encuentra constituida por los archivos:

- ◆ JSP: Objetos que forman las pantallas del sistema.
- ◆ CSS: Objetos que definen el formato y diseño de las páginas del sistema.

- ◆ JS: Objetos que proveen funcionalidad particular como validaciones o utilidades específicas como calendarios.
- ◆ GIF: Imágenes que se utilizan como parte de la presentación del sistema.

Cabe mencionar que el desarrollo de esta capa se inició desde la fase de diseño ya que para obtener el visto bueno del usuario se desarrolló, en código HTML, la secuencia de pantallas mostrada en el “Capítulo 4, Análisis y Diseño del Sistema” tomando este código HTML, como base para el desarrollo de los archivos JSP que constituyen la interfaz de usuario.

A continuación se muestra el inventario de componentes que constituyen la capa de presentación del sistema.

A r c h i v o s J S P		
Nombre	Caso de uso que lo utiliza	Funcionalidad
sass_ActDatosAlumno.jsp	Actualizar Datos Personales.	Proporciona la interfaz gráfica en la que el Alumno puede actualizar sus datos personales.
sass_ActualizaInf.jsp	Actualizar el estado de los Informes Bimestrales de un Alumno en el Sistema de Administración del Servicio Social.	Presenta la pantalla con la información de los informes del alumno.
sass_AgregaComentario.jsp	Ingresar Comentarios de un Alumno en el Sistema de Administración del Servicio Social.	Provee la pantalla en la el coordinador de la división agrega comentarios de un Alumno.
sass_AltaBaja.jsp	Autorizar Prealta de un Alumno en el Sistema de Administración del Servicio Social. Autorizar Alta de un Alumno en el Sistema de Administración del Servicio Social. Autorizar Prebaja de un Alumno en el Sistema de Administración del Servicio Social. Autorizar Baja de un Alumno en el Sistema de Administración del Servicio Social.	Provee la pantalla en la que se muestra la solicitud del Alumno para proceder a autorizar o rechazar la prealta, alta, prebaja o baja de dicho Alumno.
sass_AltaCoord.jsp	Dar mantenimiento de cuentas de Coordinadores de SS	Muestra el formato para dar de alta o modificar a un coordinador.
sass_AltaResp.jsp	Asociar nuevo responsable de programa de SS a un	Muestra el formato para dar de alta o modificar a un

	alumno.	responsable de Programa de SS
sass_BuscaCta.jsp	Autorizar Prebaja de un Alumno en el Sistema de Administración del Servicio Social. Actualizar el Estado de los Informes Bimestrales de un Alumno.	Constituye la pantalla en la que el coordinador captura el número de cuenta del Alumno que desea buscar.
sass_BuscaPrograma.jsp	Registrar un Alumno en el Sistema (En la modalidad de Buscar Programa).	Constituye la pantalla en la que el Alumno captura la clave del programa que desea buscar.
sass_ConMtoCoord.jsp	Dar mantenimiento de cuentas de Coordinadores de SS	Despliega la lista de los coordinadores.
sass_ConMtoResp.jsp	Asociar nuevo responsable de programa de SS a un alumno.	Muestra la lista de los responsables de un Programa de SS
sass_Consulta.jsp	Autorizar Terminación de un Alumno en el Sistema de Administración del Servicio Social. Autorizar Liberación de un Alumno en el Sistema de Servicio Social. Muestra la información relacionada a los movimientos del Alumno.	Pantalla en la cual se muestran los resultados de la Consulta de Avance de un Alumno en el Sistema de Administración del Servicio Social.
sass_CambiaPasswd.jsp	Cambiar Contraseña del Usuario.	Proporciona la pantalla en la que cualquier usuario del sistema puede cambiar su contraseña.
sass_ConsultaMens1.jsp	Consultar Mensajes en el Sistema de Administración del Servicio Social.	Despliega una pantalla con la lista de los mensajes de los usuarios.
sass_ConsultaMens2.jsp	Consultar Mensajes en el Sistema de Administración del Servicio Social.	Muestra el contenido de un mensaje en específico del usuario.
sass_CuerpoEstadisticas.jsp	Generar Estadísticas.	Conforma la parte inferior de la pantalla en la que se despliega una estadística. Este componente proporciona el detalle de la información que proporciona la estadística generada por el usuario.
sass_Cuerpo_Rpt.jsp	Generar Reportes.	Conforma la parte inferior de la pantalla en la que se despliega un reporte. Este

		componente proporciona el detalle de la información que proporciona el reporte generado por el usuario.
sass_EncabezadoEstadisticas.jsp	Generar Estadísticas.	Conforma la parte superior de la pantalla en la que se despliega una estadística. Este componente proporciona el encabezado de la estadística generada por el usuario.
sass_Encabezado_Rpt.jsp	Generar Reportes.	Conforma la parte superior de la pantalla en la que se despliega un reporte. Este componente proporciona el encabezado del reporte generado por el usuario.
sass_FiltrosEstadistCoord.jsp	Generar Estadísticas.	Proporciona la interfaz gráfica en la que el Coordinador de la división selecciona los parámetros con los cuales desea generar la estadística.
sass_FiltrosEstadistJefe.jsp	Generar Estadísticas.	Proporciona la interfaz gráfica en la que el Jefe de Servicio Social de la FI de la división selecciona los parámetros con los cuales desea generar la estadística.
sass_FiltrosReportes.jsp	Generar Reportes.	Proporciona la interfaz gráfica en la que el Coordinador de la división o el Jefe de SS seleccionan los parámetros con los cuales desea generar el reporte.
sass_Footer.jsp	Validar Usuario. (Forma parte de la página principal del sistema).	Este componente se utiliza para desplegar el contenido del frame superior de la pantalla principal del sistema.
sass_Liberacion.jsp	Autorizar Terminación de un Alumno en el Sistema de Administración del Servicio Social.	Muestra la lista de los alumnos en proceso de terminación, liberación u obtención de información para la carta de liberación. Autorizar Liberación de un Alumno en el Sistema de Servicio Social. Obtener información para la carta de liberación de un alumno.

sass_ListaAlumnosPrealta.jsp	Autorizar Prealta de un Alumno en el Sistema de Administración del Servicio Social.	Su función es desplegar la lista de Alumnos que se encuentran en espera de autorización de prealta.
sass_ListaDivisiones.jsp	Autorizar Alta de un Alumno en el Sistema de Administración del Servicio Social. Autorizar Baja de un Alumno en el Sistema de Administración del Servicio Social. Autorizar Liberación de un Alumno en el Sistema de Administración del Servicio Social.	Su función es desplegar la lista de Divisiones de Ingeniería que tienen Alumnos en un cierto estatus (en espera de: alta, baja o liberación).
sass_ListaProgramas.jsp	Registrar un Alumno en el Sistema (En la modalidad de Listar Programas).	Constituye la pantalla en la que se listan los programas en los que el Alumno puede realizar su servicio social.
sass_Login.jsp	Validar Usuario.	Componente que se utiliza para autenticar un usuario en el sistema y así permitir o negar el acceso al mismo.
sass_Logoff.jsp	Todos los casos de uso.	Componente no visible que se utiliza para invalidar la sesión de un usuario.
sass_Mensaje.jsp	Todos los casos de uso.	Proporciona una pantalla genérica que despliega los mensajes al usuario, ya sea de error, éxito o fracaso de la operación.
sass_MensajeInf.jsp	Actualizar el estado de los Informes Bimestrales de un Alumno en el Sistema de Administración del Servicio Social.	Constituye la pantalla en la cual se actualizan los informes bimestrales de un alumno.
sass_Menu.jsp	Validar Usuario. (Forma parte de la página principal del sistema).	Este componente se utiliza para visualizar las opciones del sistema que deben ser mostradas al usuario, esto de acuerdo al proceso de validación previamente realizado.
sass_PolíticasUso.jsp	Validar Usuario. (Forma parte de la página de inicio del sistema).	Su función es desplegar la lista de políticas de uso del sistema que todo usuario debe conocer.
sass_Principal.jsp	Validar Usuario. (Forma	Constituye el frame principal

	parte de la página de principal del sistema).	de la página Web, el cual a su vez se subdivide en otros frames que muestran otras secciones del sistema.
sass_ProcesaConsultas.jsp	Consultar Mensajes en el Sistema de Administración del Servicio Social. Dar mantenimiento de cuentas de Coordinadores de SS Asociar nuevo responsable de programa de SS a un alumno.	Este componente es provee la interfaz para realizar las operaciones de mantenimiento de mensajes, coordinadores y responsables de Programa de SS
sass_ProcesaLib.jsp	Autorizar Terminación de un Alumno en el Sistema de Administración del Servicio Social. Autorizar Liberación de un Alumno en el Sistema de Servicio Social.	Conforma la pantalla en la cual el usuario puede actualizar los estados de terminación y liberación del alumno.
sass_ProgramaBuscado.jsp	Registrar un Alumno en el Sistema (En la modalidad de Buscar Programa).	Su función es desplegar la información correspondiente al programa buscado por el Alumno.
sass_RegistroAlumno.jsp	Registrar un Alumno en el Sistema de Administración de Servicio Social.	Constituye la interfaz gráfica que despliega el formato de Solicitud de Registro de Servicio Social del Alumno.
sass_ReiniciaPasswd.jsp	Reiniciar Contraseña del Alumno.	Proporciona la pantalla en la que el Coordinador de División o el Jefe de SS pueden reiniciar la contraseña de un Alumno.
sass_Top.jsp	Validar Usuario. (Forma parte de la página principal del sistema).	Este componente se utiliza para desplegar el contenido del frame superior de la pantalla principal del sistema.

Archivos CSS		
Nombre	Caso de uso que lo utiliza	Funcionalidad
styles.css	Todos los casos de uso.	Componente que define el formato y diseño de las páginas del sistema.

A r c h i v o s J S		
Nombre	Caso de uso que lo utiliza	Funcionalidad
valida_Campos.js	<p>Registrar un Alumno en el Sistema de Administración del Servicio Social.</p> <p>Validar Preacta de un Alumno en el Sistema de Administración del Servicio Social.</p> <p>Validar Alta de un Alumno en el Sistema de Administración del Servicio Social.</p>	Realiza la validación de los datos capturados por los usuarios, verificando qué tipos de caracteres (números o letras) se capturan en los campos.
date_Picker.js	<p>Generar Estadísticas.</p> <p>Generar Reportes.</p>	Componente que provee una interfaz de calendario para la selección del periodo de fechas a consultar en estos casos de uso.

A r c h i v o s G I F		
Nombre	Caso de uso que lo utiliza	Funcionalidad
logo_sass.gif	Validar Usuario. (Forma parte de la página principal del sistema).	Imagen que muestra el logotipo del sistema.
fondo.gif	Validar Usuario. (Forma parte de la página principal del sistema).	Imagen que sirve de fondo en los frames superior e inferior de la página principal del sistema.
show_Calendar.gif	<p>Generar Estadísticas.</p> <p>Generar Reportes.</p>	Esta imagen se muestra en el componente que provee una interfaz de calendario.

5.2.2. Middleware o Capa de la Lógica del Negocio

El middleware o lógica del negocio de la aplicación se encuentra constituida por las dos capas siguientes:

- ◆ **Objetos del Dominio:** Clases que representan los conceptos del dominio.
- ◆ **Servicios:** Las clases controladoras que proveen servicios como la interacción con la base de datos, los reportes o alguna operación en el sistema.

A continuación se muestra el inventario de componentes que constituyen la capa de la lógica del negocio del sistema.

A r c h i v o s J A V A (B e a n s)		
Nombre	Caso de uso que lo utiliza	Funcionalidad
Alumno.class	<p>Registrar un Alumno en el Sistema de Administración del Servicio Social.</p> <p>Autorizar Preacta de un Alumno en el Sistema de Administración del Servicio Social.</p> <p>Autorizar Alta de un Alumno en el Sistema de Administración del Servicio Social.</p> <p>Autorizar Prebaja de un Alumno en el Sistema de Administración del Servicio Social.</p> <p>Autorizar Baja de un Alumno en el Sistema de Administración del Servicio Social.</p> <p>Actualizar el Estado de los Informes Bimestrales de un Alumno.</p> <p>Autorizar Liberación de un Alumno en el Sistema de Administración del Servicio Social.</p> <p>Actualizar Datos Personales.</p> <p>Reiniciar Contraseña del Alumno.</p>	Esta clase modela al objeto Alumno y contiene las propiedades que definen a dicho objeto.
Carta.class	Obtener información para la carta de liberación de un alumno.	Clase que contiene la información necesaria para la carta de liberación.
Comentario.class	Ingresar Comentarios de un Alumno en el Sistema de Administración del Servicio Social.	Clase que encapsula las propiedades que definen al objeto Comentario.
ConsAlu.class	Consultar Avance de un Alumno en el Sistema de	Modela la consulta de información de los

	<p>Administración del Servicio Social.</p> <p>Autorizar Terminación de un Alumno en el Sistema de Administración del Servicio Social.</p> <p>Autorizar Liberación de un Alumno en el Sistema de Servicio Social.</p>	<p>movimientos realizados por el alumno en el ejercicio de su SS</p>
Consulta.class	<p>Generar Reportes.</p> <p>Generar Estadísticas.</p>	<p>Clase que modela una consulta ya sea para estadísticas o reportes y por lo tanto encapsula las propiedades de la consulta definidas por el usuario.</p>
Coordinador.class	<p>Dar mantenimiento de cuentas de Coordinadores de SS.</p>	<p>Encapsula las propiedades del objeto coordinador.</p>
DatosConsRep.class	<p>Actualizar el estado de los Informes Bimestrales de un Alumno en el Sistema de Administración del Servicio Social.</p>	<p>Contiene la información llave para actualizar los informes bimestrales de un alumno.</p>
Division.class	<p>Autorizar Alta de un Alumno en el Sistema de Administración del Servicio Social.</p> <p>Autorizar Baja de un Alumno en el Sistema de Administración del Servicio Social.</p> <p>Autorizar Liberación de un Alumno en el Sistema de Administración del Servicio Social.</p>	<p>Esta clase encapsula las propiedades que conforman al objeto División.</p>
LiberacionImp.class	<p>Autorizar Terminación de un Alumno en el Sistema de Administración del Servicio Social.</p> <p>Autorizar Liberación de un Alumno en el Sistema de Servicio Social.</p>	<p>Contiene la información llave para actualizar la terminación y liberación del alumno.</p>
Mensaje.class	<p>Consultar Mensajes en el Sistema de Administración del Servicio Social.</p>	<p>Contiene la información de un mensaje en específico de un usuario.</p>

Programa.class	Registrar un Alumno en el Sistema de Administración del Servicio Social.	Componente que encapsula las propiedades que definen a un Programa de SS.
ProgramaElegido.class	Registrar un Alumno en el Sistema de Administración del Servicio Social.	Componente que contiene las propiedades necesarias para identificar el programa elegido por el Alumno para realizar el registro.
Responsable.class	Asociar nuevo responsable de programa de SS a un alumno.	Este componente encapsula las propiedades que definen a un responsable de Programa de SS
SeccionAlumno.class	Registrar un Alumno en el Sistema de Administración del Servicio Social. Actualizar Datos Personales del Alumno.	Clase que modela el conjunto de datos personales que deben ser proporcionados y actualizados por el Alumno.

Archivos JAVA (Proveedoras de Servicios)		
Nombre	Caso de uso que lo utiliza	Funcionalidad
ActReportesBim.class	Actualizar el estado de los Informes Bimestrales de un Alumno en el Sistema de Administración del Servicio Social.	Actualiza los informes bimestrales de un alumno.
ConsMovAlu.class	Consultar Avance de un Alumno en el Sistema de Administración del Servicio Social. Autorizar Terminación de un Alumno en el Sistema de Administración del Servicio Social. Autorizar Liberación de un Alumno en el Sistema de Servicio Social.	Contiene la información llave para obtener los movimientos del alumno.
ControlaAlumno.class	Registrar un Alumno en el Sistema de Administración del Servicio Social. Autorizar Prealta de un Alumno en el Sistema de Administración del Servicio Social. Autorizar Alta de un Alumno en el Sistema de	Esta clase controla las operaciones y búsquedas que se realizan sobre el objeto Alumno.

	<p>Administración del Servicio Social.</p> <p>Autorizar Prebaja de un Alumno en el Sistema de Administración del Servicio Social.</p> <p>Autorizar Baja de un Alumno en el Sistema de Administración del Servicio Social.</p> <p>Actualizar el Estado de los Informes Bimestrales de un Alumno.</p> <p>Autorizar Liberación de un Alumno en el Sistema de Administración del Servicio Social.</p> <p>Actualizar Datos Personales.</p> <p>Reiniciar Contraseña del Alumno.</p>	
ControlaComentario.class	Ingresar Comentarios de un Alumno en el Sistema de Administración del Servicio Social.	Clase que controla la acción de ingresar comentarios de un alumno en el sistema.
ControlaDivision.class	<p>Autorizar Alta de un Alumno en el Sistema de Administración del Servicio Social.</p> <p>Autorizar Baja de un Alumno en el Sistema de Administración del Servicio Social.</p> <p>Autorizar Liberación de un Alumno en el Sistema de Administración del Servicio Social.</p>	Componente que realiza las consultas de las divisiones que tienen alumnos en un estatus en particular.
ControlaEstadistica.class	Generar Estadísticas.	Clase controladora que genera las estadísticas de acuerdo a las propiedades definidas para el objeto Consulta.
ControlaPrograma.class	Registrar un Alumno en el	Componente que controla la

	Sistema de Administración del Servicio Social.	búsqueda de los programas permitidos para un Alumno.
ControlaReporte.class	Generar Reportes.	Clase controladora que genera los reportes de acuerdo a las propiedades definidas para el objeto Consulta.
ControlaSolicitud.class	<p>Registrar un Alumno en el Sistema de Administración del Servicio Social.</p> <p>Autorizar Prealta de un Alumno en el Sistema de Administración del Servicio Social.</p> <p>Autorizar Alta de un Alumno en el Sistema de Administración del Servicio Social.</p> <p>Autorizar Prebaja de un Alumno en el Sistema de Administración del Servicio Social.</p> <p>Autorizar Baja de un Alumno en el Sistema de Administración del Servicio Social.</p>	Este componente realiza las operaciones de autorización y rechazo de movimientos a un Alumno y que requieren de la revisión de la información capturada en la solicitud.
ControlaUsuario.class	<p>Validar Usuario.</p> <p>Cambiar Contraseña del Usuario.</p>	Clase que controla las acciones que corresponden al objeto Usuario, es decir a cualquier tipo de usuario del sistema.
FormatFecha.class	Actualizar el estado de los Informes Bimestrales de un Alumno en el Sistema de Administración del Servicio Social.	Valida que una fecha sea válida.
LiberacionesImp.class	<p>Autorizar Terminación de un Alumno en el Sistema de Administración del Servicio Social.</p> <p>Autorizar Liberación de un Alumno en el Sistema de Servicio Social.</p> <p>Obtener información para la carta de liberación de un alumno.</p>	Actualiza el estatus de terminación, liberación y carta de liberación de un alumno.

Mensajes.class	Consultar Mensajes en el Sistema de Administración del Servicio Social.	Obtiene los mensajes asociados a un usuario del Sistema.
MtoCoord.class	Dar mantenimiento de cuentas de Coordinadores de SS	Realiza las operaciones de alta, baja y modificaciones de coordinadores de SS
MtoResp.class	Asociar nuevo responsable de programa de SS a un alumno.	Realiza las operaciones de alta, baja y modificaciones de responsables de Programas de SS
ProveeConexion.class	Todos los casos de uso.	Esta clase se encarga de administrar la apertura, validación y terminación de la conexión de la base de datos, así como también provee otros servicios genéricos utilizados por todas las clases proveedoras de servicios.
RepAlu.class	Actualizar el estado de los Informes Bimestrales de un Alumno en el Sistema de Administración del Servicio Social.	Obtiene la información y estatus de los informes bimestrales del alumno.
SassException.class	Todos los casos de uso.	Este componente se encarga de levantar las excepciones personalizadas definidas para el sistema.

Servlets		
Nombre	Caso de uso que lo utiliza	Funcionalidad
CtaArch.class	Obtener información para la carta de liberación de un alumno.	Permite bajar la información de los alumnos que necesiten imprimir la carta de liberación.

5.2.3. Back-End o Capa de Almacenamiento

El back-end o capa de almacenamiento de la aplicación se encuentra conformada por la base de datos Sybase Enterprise versión 11.0.

Para la construcción de la base de datos, se creó un script que realiza las siguientes tareas:

- ◆ Creación de las tablas.
- ◆ Generación de *constraints*.
- ◆ Creación de índices.
- ◆ Carga de catálogos.

A continuación se muestran pequeños fragmentos de este script que ejemplifican el código que realiza las tareas listadas anteriormente, para mayor información sobre el procedimiento de creación de la base de datos, consulte el Capítulo 7 de esta tesis.

```
/*Crea la tabla ss_alumno*/
create table ss_alumno (
    alu_num_cta varchar (15) not null ,
    alu_tel varchar (20) null ,
    alu_cel varchar (20) null ,
    alu_pager varchar (20) null ,
    alu_pin varchar(5) null,
    alu_email varchar (30) null ,
    alu_edad smallint not null ,
    alu_colonia varchar (50) null .
    alu_calle varchar (40) null .
    alu_numero varchar (30) null ,
    alu_cp varchar (5) null ,
    alu_ent_fed varchar (50) null ,
    alu_del_mpo varchar (50) null ,
    alu_sem_ingr smallint not null
)
go

/*Generación del constraint de la tabla ss_alumno*/
alter table ss_alumno add
constraint fk_ss_alu_ss_usu foreign key
(
    alu_num_cta
) references ss_usuario (
    usu_cve
)
go

/*Creación de índices para la tabla ss_programa*/
/*Tabla de ss_programa(Tipo Programa)*/
if exists(select name from sysindexes where name = 'ind_prg1')
drop index ss_programa.ind_prg1
go

create index ind_prg1
on ss_programa(prg_tipo)
go

/*Tabla de ss_programa(Area)*/
if exists(select name from sysindexes where name = 'ind_prg2')
drop index ss_programa.ind_prg2
go
```

```

create index ind_prg2
  on ss_programa(prg_cve_area)
go

```

*/*Carga del catálogo de divisiones (ss_division)*/*

```

if not exists (select div_cve from ss_division where div_cve=0)
  insert into ss_division (div_cve,div_desc) values(0,'DICTYG')
go

if not exists (select div_cve from ss_division where div_cve=1)
  insert into ss_division (div_cve,div_desc) values(1,'DICT')
go

if not exists (select div_cve from ss_division where div_cve=2)
  insert into ss_division (div_cve,div_desc) values(2,'DIMET')
go

if not exists (select div_cve from ss_division where div_cve=3)
  insert into ss_division (div_cve,div_desc) values(3,'DIE')
go

if not exists (select div_cve from ss_division where div_cve=10)
  insert into ss_division (div_cve,div_desc) values(10,'Ciencias Basicas')
go

```

Además de las tablas de la base de datos y de sus respectivos índices, se crearon otros objetos que desempeñan funciones específicas para algunos módulos del sistema. Estos objetos son *stored procedures*, *triggers* y *vistas* y sus características se muestran a continuación.

Stored Procedures		
Nombre	Caso de uso que lo utiliza	Funcionalidad
sp_ndias	Autorizar Alta de un Alumno en el Sistema de Servicio Social	Obtiene el numero de días por mes
sp_dias	Autorizar Alta de un Alumno en el Sistema de Servicio Social	Obtiene un día específico de la cadena de días festivos
sp_fechas	Autorizar Alta de un Alumno en el Sistema de Servicio Social	Obtiene las fechas de terminación y entrega de reportes
sp_consmovalum	Consultar Avance de un Alumno en el Sistema de Administración del Servicio Social.	Obtiene los movimientos realizados por el alumno
sp_meses_ss	Autorizar Alta de un Alumno en el Sistema de Servicio Social	Obtiene el número de meses en el que el alumno realizará su servicio social
sp_consrptbim_ss	Actualizar el estado de los	Obtiene los reportes realizados

	Informes Bimestrales de un Alumno en el Sistema de Administración del Servicio Social	por el alumno
sp_actrptbim_ss	Actualizar el estado de los Informes Bimestrales de un Alumno en el Sistema de Administración del Servicio Social	Actualiza los reportes realizados por el alumno
sp_actlib_ss	Autorizar Terminación de un Alumno en el Sistema de Administración del Servicio Social. Autorizar Liberación de un Alumno en el Sistema de Servicio Social.	Actualiza los estatus de liberación, terminación y carta de liberación
sp_folio_ss	Obtener información para la carta de liberación de un alumno.	Obtiene el ultimo folio del año en curso
sp_bajas_ss	Autorizar Baja de un Alumno en el Sistema de Servicio Social.	Contiene la lógica para dar de baja a alumnos en el proceso de servicio social
sp_actregprg_ss	Asociar nuevo responsable de programa de SS a un alumno.	Asocia el responsable de Programa de SS al registro del alumno.
sp_altaresp_ss	Asociar nuevo responsable de programa de SS a un alumno.	Inserta un nuevo responsable de programa de SS

V i s t a s		
Nombre	Caso de uso que lo utiliza	Funcionalidad
vw_infcarta_ss	Descargar datos para la carta de liberación.	Obtiene la información de la carta de liberación de los alumnos en este proceso.

T r i g g e r s		
Nombre	Caso de uso que lo utiliza	Funcionalidad
upddaterep	Autorizar Alta de un Alumno en el Sistema de Servicio Social.	Actualiza las fechas de entrega de informes bimestrales y liberación de un alumno.
tr_mensaje	Consultar Mensajes en el Sistema de Administración del Servicio Social.	Actualiza el identificador del mensaje cuando se inserta un mensaje.

5.2.4. Paquetes Descriptivos de la Arquitectura

La arquitectura del sistema puede representarse en términos de paquetes de UML los cuales nos permiten representar gráficamente cómo se encuentra construido el sistema y los elementos generales que lo conforman.

Los paquetes organizan elementos de un mismo tipo y gráficamente pueden representar una división física o lógica de los componentes de acuerdo a su función.

El siguiente diagrama representa la arquitectura adoptada para la construcción del sistema.

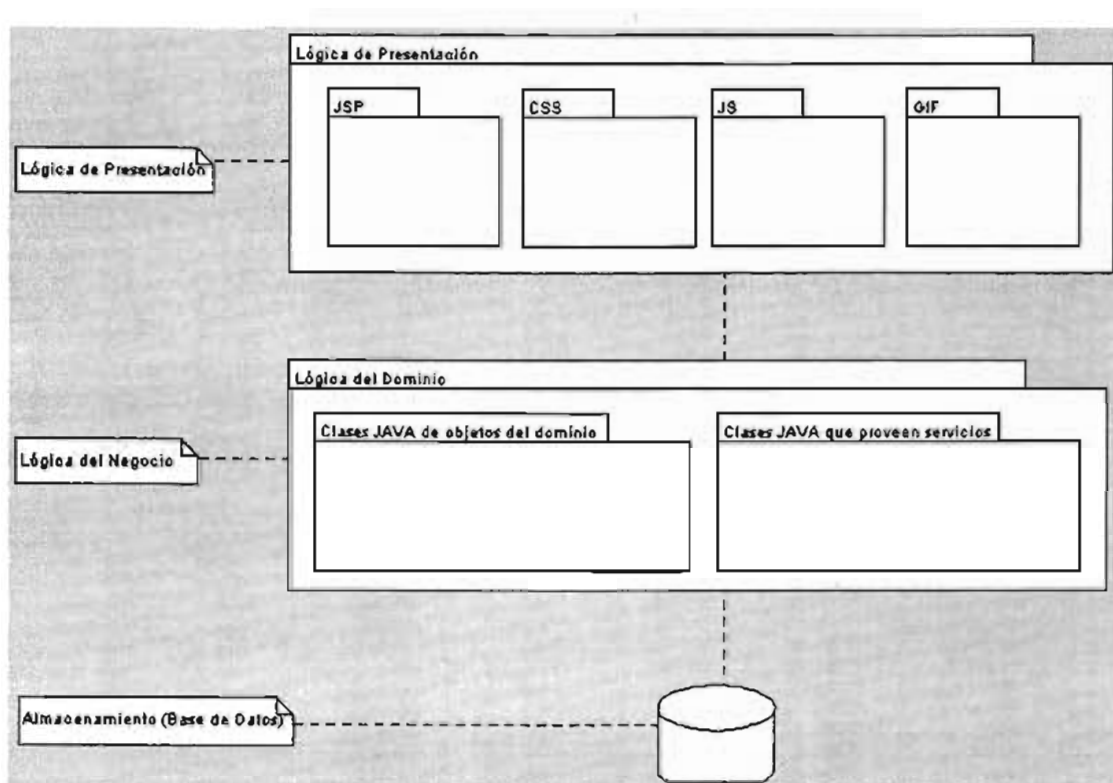


Figura 5.1 Unidades arquitectónicas del sistema expresadas en términos de paquetes de UML.

5.3 Herramientas de Diseño y Programación Utilizadas en la Construcción del Sistema de Administración de Servicio Social.

A continuación se ilustran las herramientas utilizadas como apoyo para el diseño y programación del sistema.

5.3.1. Herramientas de Diseño

- ◆ Adobe Photoshop Elements 2.0

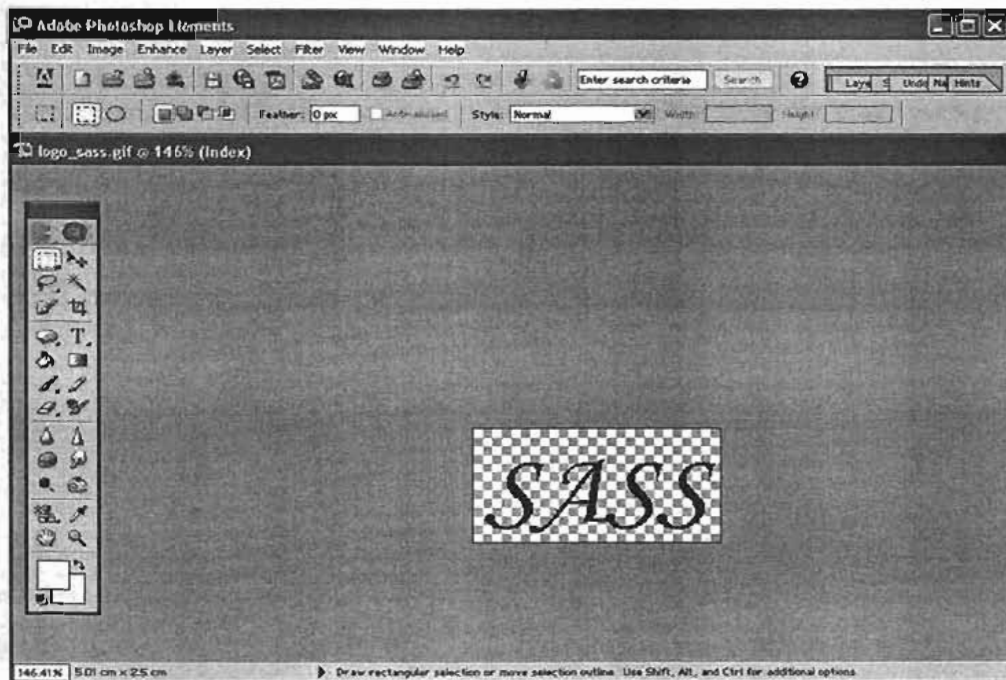


Figura 5.2 Herramienta de diseño utilizada en la creación de imágenes y fondos del SASS.

- ♦ Macromedia Dreamweaver UltraDev4.



Figura 5.3 Herramienta de diseño utilizada en la construcción de los JSP's y HTML's del SASS.

5.3.2 Herramientas de Programación

- ♦ TextPad 4.5.0

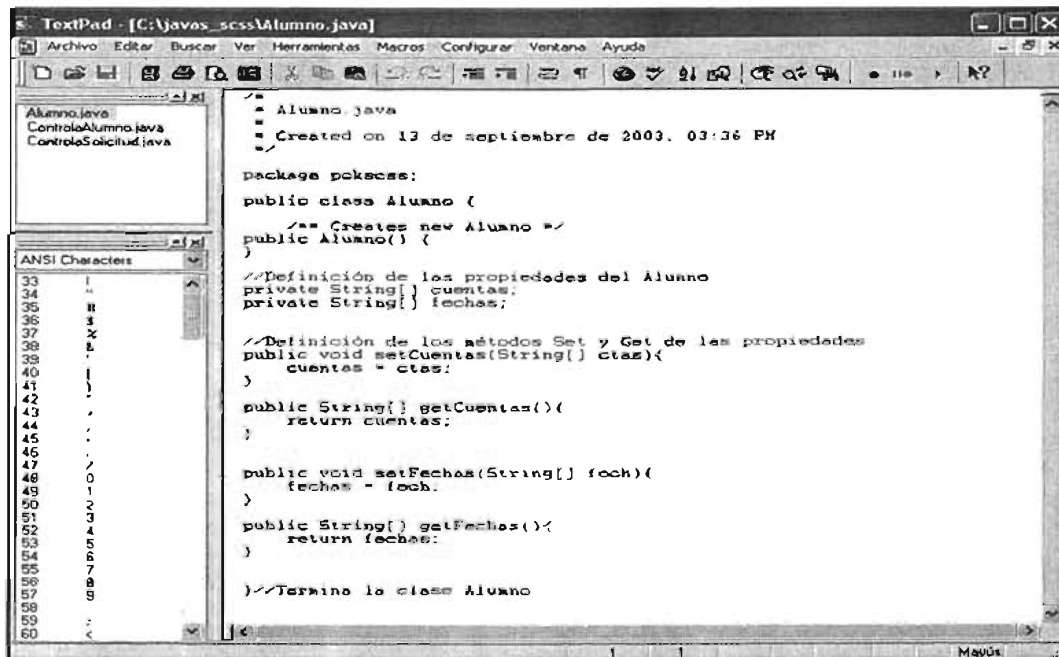


Figura 5.4 Herramienta de programación utilizada en la construcción de las clases JAVA del SASS.

5.4 Estándares de Programación

Los estándares de programación son lineamientos que deben seguirse en el desarrollo de aplicaciones de cualquier tipo y cuyo objetivo es facilitar el entendimiento del código fuente desarrollado por diferentes programadores.

Para el desarrollo del Sistema de Administración del Servicio Social se siguieron los siguientes estándares para la creación de los objetos que componen la aplicación así como de los elementos que componen un programa.

- ◆ **Nombre de Clase:** Todo nombre de una clase debe iniciar con una letra y contener la primera letra de cada palabra que componga el nombre de la clase con letra mayúscula.

Ejemplos: Alumno.class
 Carta.class
 ControlaSolicitud.class

- ◆ **Nombre de Método:** El nombre de un método debe iniciar con letra minúscula y si se conforma de dos o más palabras, a partir de la segunda palabra todas deben iniciar con mayúscula y continuar con minúscula.

Ejemplos: autorizaMovimiento()
 enviaMensaje()
 llenaSeccDepen()

- ◆ **Nombre de Variable:** Una variable debe tener un nombre representativo que ayude a identificar el papel que desempeña dentro de una clase. Es recomendable que inicie con una letra y que se escriba sólo en minúsculas cuando la variable tiene asociado un tipo de dato de Java y que inicie con mayúscula cuando la variable es una instancia de otra clase.

El nombre de una variable debe contener lo menos posible de caracteres especiales, además de que no debe contener una palabra reservada.

Ejemplo: public [static] double contador = 0;

- ◆ **Nombre de Constante:** El nombre de una constante debe ser representativo y escrito todo en mayúsculas, iniciar con una letra y no debe contener palabras reservadas. Las constantes tienen asociado un tipo de dato primitivo, un nombre y un valor.

Ejemplo: public static final int ESCALA = 2;

- ◆ **Nombre de JSP:** El nombre de un componente JSP debe iniciar con el prefijo sass_ y seguir con el nombre que identifica al JSP con la primera letra de cada palabra en mayúscula.

- ◆ Ejemplos: sass_Login.jsp
 sass_MensajeInf.jsp

Pruebas del Sistema

CAPÍTULO 6

PRUEBAS DEL SISTEMA

6.1 Pruebas Realizadas por el Equipo de Trabajo

6.1.1. Pruebas de Software

Con el fin de revisar la funcionalidad del sistema y de verificar que se cubran los requerimientos descritos implícitamente en los casos de uso del Capítulo 4, se realizaron las pruebas semifinales correspondientes en cada uno de los módulos del sistema obteniéndose los resultados que se muestran a continuación.

- ◆ Caso de uso: Realizar el Registro de un Alumno en el Sistema de Administración del Servicio Social.

Tipo de prueba	Resultado esperado	Observaciones
Realizar el registro de un alumno capturando datos consistentes en los campos del formato de solicitud.	En ambos tipos de prueba el sistema debe verificar la consistencia de los tipos de datos capturados de acuerdo a la naturaleza de la información que se solicita en el formato de registro. Además debe realizar el registro del alumno en el programa seleccionado, así como también debe almacenar los datos personales proporcionados por el alumno.	El sistema no permitió capturar letras en el campo de "Número", lo cual es incorrecto ya que existen direcciones que cuyo "Número" incluye letras. Se procedió a realizar las correcciones correspondientes (consultar la sección Corrección de Programas de este capítulo).
Realizar el registro de un alumno capturando datos inconsistentes en los campos del formato de solicitud (Por ejemplo letras en campos de teléfono, monto, código postal, etc.).	El alumno debe quedar en estatus de registro y por lo tanto en espera de autorización de prealta.	El sistema realizó las validaciones correctamente.

- ◆ Caso de uso: Autorizar Prealta de un Alumno en el Sistema de Administración del Servicio Social.

Tipo de prueba	Resultado esperado	Observaciones
Consulta de alumnos en espera de autorización de prealta.	El sistema debe desplegar en pantalla una lista de los alumnos que se encuentran en estatus de registro (en espera de autorización de prealta)	La consulta fue realizada correctamente.

Abrir la solicitud elaborada por un alumno.	Cuando el usuario seleccione un número de cuenta, el sistema debe abrir la solicitud del alumno correspondiente.	La búsqueda y el despliegado de la información se realizaron correctamente.
Autorizar/rechazar la prealta de un alumno.	<p>Si el usuario elige la opción Autorizar, el sistema debe ejecutar dicha acción.</p> <p>Si el usuario elige la opción Rechazar, el sistema debe validar que los Motivos de Rechazo hayan sido capturados.</p>	<p>En el caso de la autorización, el sistema realizó el movimiento correctamente.</p> <p>En el caso del rechazo el sistema realizó el cambio de estatus correctamente pero no se validó la captura de los Motivos de Rechazo (consultar la sección Corrección de Programas de este capítulo).</p>

- ♦ Caso de uso: Autorizar Alta de un Alumno en el Sistema de Administración del Servicio Social.

Tipo de prueba	Resultado esperado	Observaciones
Consulta de las Divisiones de Ingeniería que tienen alumnos en espera de autorización de alta.	El sistema debe desplegar en pantalla una lista de las divisiones de Ingeniería que tienen alumnos en estatus de prealta.	La consulta y el despliegado de información se realizaron correctamente.
Consulta de alumnos en espera de autorización de alta.	Cuando el usuario elige una División de Ingeniería, el sistema debe desplegar en pantalla una lista de los alumnos que se encuentran en estatus de prealta.	La consulta y el despliegado de información fueron correctos.
Abrir la solicitud elaborada por un alumno.	Cuando el usuario seleccione un número de cuenta, el sistema debe abrir la solicitud del alumno correspondiente.	El despliegado de la información se realizó correctamente.
Autorizar/rechazar la alta de un alumno.	<p>Si el usuario elige la opción Autorizar, el sistema debe ejecutar dicha acción.</p> <p>Si el usuario elige la opción Rechazar, el sistema debe validar que los Motivos de Rechazo hayan sido capturados.</p>	<p>En el caso de la autorización, el sistema realizó el movimiento correctamente.</p> <p>En el caso del rechazo el sistema realizó el cambio de estatus correctamente pero no se validó la captura de los Motivos de Rechazo (consultar la sección Corrección de Programas de este capítulo).</p>

- ◆ Caso de uso: Realizar Prebaja de un Alumno en el Sistema de Administración del Servicio Social.

Tipo de prueba	Resultado esperado	Observaciones
Buscar el número de cuenta de un alumno no registrado en el sistema.	El sistema debe enviar al usuario un mensaje que indique la inexistencia del registro del alumno.	El mensaje envía el mensaje satisfactoriamente.
Buscar el número de cuenta de un alumno registrado en el sistema y que se encuentra en ejercicio de su SS.	El sistema debe desplegar en pantalla el formato de la solicitud del alumno.	El sistema sí despliega la solicitud del alumno.
Autorizar la prebaja de un alumno.	El alumno debe registrarse en estatus de prebaja validando previamente que los Motivos de prebaja hayan sido capturados por el usuario.	La prebaja se realizó correctamente.

- ◆ Caso de uso: Autorizar Baja de un Alumno en el Sistema de Administración del Servicio Social.

Tipo de prueba	Resultado esperado	Observaciones
Consulta de las Divisiones de Ingeniería que tienen alumnos en espera de autorización de baja.	El sistema debe desplegar en pantalla una lista de las divisiones de Ingeniería que tienen alumnos en estatus de prebaja.	La consulta y el desplegado de información se realizaron correctamente.
Consulta de alumnos en espera de autorización de baja.	Cuando el usuario elige una División de Ingeniería, el sistema debe desplegar en pantalla una lista de los alumnos que se encuentran en estatus de prebaja.	La consulta y el desplegado de información fueron correctos.
Abrir la solicitud elaborada por un alumno.	Cuando el usuario seleccione un número de cuenta, el sistema debe abrir la solicitud del alumno correspondiente.	La solicitud fue mostrada correctamente.
Autorizar la baja de un alumno.	El sistema debe realizar la baja del alumno cuando el usuario elige la opción Autorizar.	La baja fue exitosa.

- ◆ Caso de uso: Ingresar Comentarios de un Alumno en el Sistema de Administración del Servicio Social.

Tipo de prueba	Resultado esperado	Observaciones
Ingresar un comentario para un alumno que no ha registrado su SS o cuyo proceso no es administrado por el coordinador que intenta ingresar el comentario.	El sistema no debe registrar el comentario.	Correcto, el sistema no registra el comentario y envía el mensaje correspondiente.
Ingresar un comentario para un alumno que sí ha registrado su SS y cuyo proceso sí es administrado por el coordinador que intenta ingresar el comentario.	El sistema sí debe registrar el comentario.	Correcto, el sistema sí registra el comentario y envía el mensaje correspondiente.

- ◆ Caso de uso: Generar Estadísticas de Alumnos en el Sistema de Administración del Servicio Social.

Tipo de prueba	Resultado esperado	Observaciones
Validar opciones en la pantalla de filtros.	El sistema debe mostrar sólo las opciones correspondientes al perfil del usuario.	Validación correcta.
Generar estadísticas sobre los diferentes estatus de los alumnos durante el SS.	Que la estadística generada por el sistema cumpla con las características requeridas en la pantalla de filtros.	El resultado generado sí cumple con las características requeridas para la estadística.

- ◆ Caso de uso: Generar Reportes de Alumnos en el Sistema de Administración del Servicio Social.

Tipo de prueba	Resultado esperado	Observaciones
Validar opciones en la pantalla de filtros.	El sistema debe mostrar sólo las opciones correspondientes al perfil del usuario.	Validación correcta.
Generar reportes sobre los diferentes estatus de los alumnos durante el SS.	Que el reporte generado por el sistema cumpla con las características requeridas en la pantalla de filtros.	El resultado generado sí cumple con las características requeridas para el reporte.

- ◆ Caso de uso: Actualizar Datos Personales de Alumnos en el Sistema de Administración del Servicio Social.

Tipo de prueba	Resultado esperado	Observaciones
Realizar cambios en uno, varios o todos los datos personales del Alumno.	El sistema debe actualizar los datos personales del Alumno.	Actualización exitosa.

- ◆ Caso de uso: Reiniciar Contraseña de un Alumno en el Sistema de Administración del Servicio Social.

Tipo de prueba	Resultado esperado	Observaciones
Intentar reiniciar la contraseña de un Alumno cuando el usuario no tiene los privilegios para ello.	El sistema debe validar que el coordinador que intenta reiniciar la contraseña no es el administrador del proceso de SS del Alumno.	Validación correcta. No se reinicia la contraseña.
Reiniciar la contraseña de un Alumno cuando el usuario si tiene los privilegios para ello.	El sistema debe validar que el coordinador que intenta reiniciar la contraseña si es el administrador del proceso de SS del Alumno o en su defecto que se trata del Jefe de SS de la Facultad.	Validación correcta. La contraseña se reinicia exitosamente.

- ◆ Caso de uso: Cambiar Contraseña del Usuario del Sistema de Administración del Servicio Social.

Tipo de prueba	Resultado esperado	Observaciones
Realizar el cambio de contraseña del usuario firmado en el sistema proporcionando una contraseña incorrecta.	El sistema debe verificar la contraseña proporcionada por el usuario que intenta cambiarla, si ésta coincide con la registrada en el sistema, entonces procede a realizar el cambio, en caso contrario no debe efectuarse la actualización.	Validación correcta. El sistema no realiza el cambio.
Realizar el cambio de contraseña del usuario firmado en el sistema proporcionando una contraseña correcta.		El resultado es correcto ya que el sistema sí ejecuta el cambio.

- ◆ Caso de uso: Autorizar Terminación de un Alumno en el Sistema de Administración del Servicio Social.

Tipo de prueba	Resultado esperado	Observaciones
Terminación, Lista de alumnos	Se presenta la lista de los alumnos susceptibles a terminación.	El desplegado se realizó correctamente.
Autorizar el proceso de terminación de un alumno.	El estado del proceso actual del alumno sea Terminación. La siguiente vez que se consulten los alumnos susceptibles a terminación, el alumno no aparezca en la lista.	La actualización y desplegado de la información se realizó correctamente.

- ◆ Caso de uso: Autorizar Liberación de un Alumno en el Sistema de Servicio Social.

Tipo de prueba	Resultado Esperado	Observaciones
Liberación, Lista de alumnos	Se presenta la lista de los alumnos susceptibles a terminación.	El desplegado se realizó correctamente.
Autorizar el proceso de liberación de un alumno.	El estado del proceso actual del alumno sea Liberación. La siguiente vez que se consulten los alumnos susceptibles a liberación, el alumno no aparezca en la lista.	La actualización y desplegado de la información se realizó correctamente.
Rechazar el proceso de liberación de un alumno.	El estado del proceso actual del alumno sea Terminación. La siguiente vez que se consulten los alumnos susceptibles a liberación, el alumno deberá aparecer en la lista. Al consultar el alumno sus movimientos deberá aparecer el rechazo del proceso de liberación.	La actualización y desplegado de la información se realizó correctamente.

- ◆ Caso de uso: Descargar Datos para la Carta de Liberación.

Tipo de prueba	Resultado Esperado	Observaciones
Información de Carta, Lista de alumnos	Se presenta una lista con los alumnos susceptibles a imprimir su carta de liberación.	El despliegue de la información se realizó correctamente.
Bajar a la máquina del cliente el archivo	El estado del proceso actual del alumno sea Carta de Liberación.	El archivo no se descargó correctamente

con la información de la carta de liberación.	El archivo que se descarga en la máquina del cliente se llama CartasAlum.csv	en la máquina, el archivo viene vacío (consultar la sección Corrección de Programas de este capítulo).
Bajar a la máquina del cliente el archivo con la información de la carta de liberación de los alumnos que requieren reimpresión.	El estado del proceso actual del alumno sea Reimpresión. El archivo que se descarga en la máquina del cliente se llama CartasAlum.csv El número de folio de la carta reimpressa sea el mismo que se imprimió por primera vez.	El archivo no se descargó correctamente en la máquina, el archivo viene vacío (consultar la sección Corrección de Programas de este capítulo).

- ◆ Caso de uso: Consultar Avance de un Alumno en el Sistema de Administración del Servicio Social.

Tipo de prueba	Resultado Esperado	Observaciones
Realizar búsqueda del un alumno inscrito al SS	Mostrar el historial de los movimientos del alumno durante el proceso de SS	El desplegado de la información se realizó correctamente.
Realizar la búsqueda de un alumno que no existe o no está inscrito en algún programa de SS	Mostrar la página informativa de Error que indica que el alumno no existe o no esta dado de alta.	La validación del alumno se realizó correctamente.

- ◆ Caso de uso: Consultar Mensajes en el Sistema de Administración del Servicio Social.

Tipo de prueba	Resultado Esperado	Observaciones
Consultar los mensajes del usuario.	Mostrar los mensajes del usuario enviados por el sistema o por otros usuarios.	El desplegado de la información se realizó correctamente.
Ver un mensaje en específico	La siguiente vez que se vean los mensajes en lista, el mensaje aparece como leído.	El desplegado de la información se realizó correctamente. La actualización del estado del mensaje se realizó correctamente.
Eliminar mensaje	La siguiente vez que se vean los mensajes en lista, el mensaje no aparece.	El desplegado de la información se realizó correctamente.

- ◆ Caso de uso: Actualizar el estado de los Informes Bimestrales de un Alumno en el Sistema de Administración del Servicio Social.

Tipo de prueba	Resultado Esperado	Observaciones
Actualizar Informe, Listar Informes.	Muestra los Informes del alumno.	El despliegue de la información se realizó correctamente.
Actualizar Informe antes de la fecha de entrega.	Muestra mensaje de que la fecha de entrega del informe no se ha cumplido.	La validación se realizó correctamente.
Actualizar Informe en la fecha de entrega o 7 días después.	Se actualiza el estado del informe.	La actualización se realizó correctamente.
Actualizar Informe ocho días después o treinta días después de la fecha de entrega.	Se actualiza el estado del informe desfasando la fecha de entrega de los informes subsecuentes.	No se desfaso la fecha de los informes subsecuentes (consultar la sección Corrección de Programas de este capítulo).
Actualizar Informe treinta días después de la fecha de entrega.	No se permite actualizar el estado del informe, porque el alumno estará dado de baja.	No nos indica que el usuario está dado de baja (consultar la sección Corrección de Programas de este capítulo).

- ◆ Caso de uso: Asociar nuevo responsable de programa de SS a un alumno.

Tipo de prueba	Resultado Esperado	Observaciones
Alta de nuevo responsable de programa de SS.	El programa de SS deberá tener asociado un nuevo responsable	La actualización se realizó correctamente.
Asociar el nuevo responsable de programa de SS o uno existente al alumno.	El alumno deberá tener asociado el responsable del programa de SS elegido.	El despliegue y actualización de la información se realizó correctamente.

- ◆ Caso de uso: Dar mantenimiento de cuentas de Coordinadores de SS

Tipo de prueba	Resultado Esperado	Observaciones
Alta de nuevo coordinador	En la lista de coordinadores deberá aparecer el nuevo coordinador.	El despliegue y actualización de la información se realizó correctamente.
Baja de un coordinador	En la lista de coordinadores no deberá aparecer el coordinador borrado.	El despliegue y actualización de la información se realizó correctamente.
Modificación de un coordinador	Al consultar los datos de un coordinador, deberá aparecer la información actualizada.	El despliegue y actualización de la información se realizó correctamente.

6.1.2. Pruebas de Hardware

El SASS fue implantado para sus pruebas sobre una arquitectura física como la que se muestra en el siguiente diagrama de despliegue.

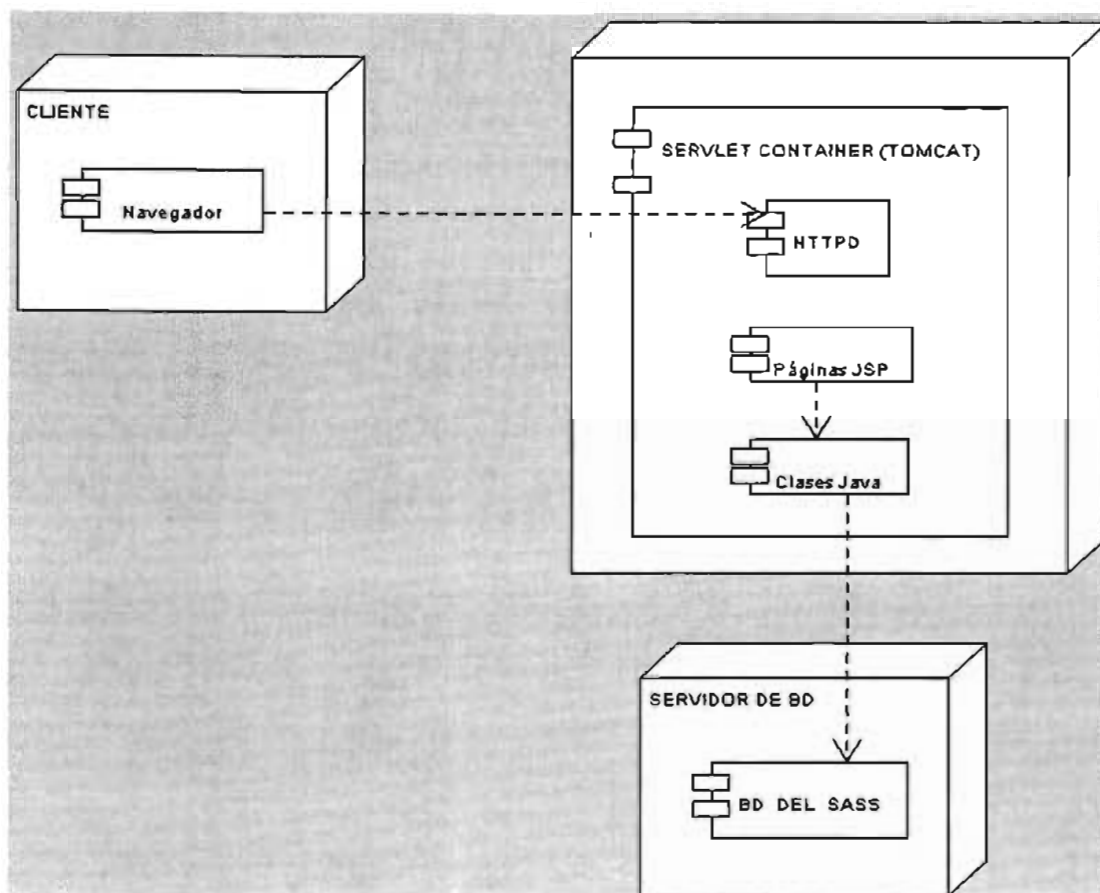


Figura 6.1. Diagrama de Despliegue del SASS.

Las pruebas de hardware del SASS se realizaron con base en las siguientes actividades:

- ◆ Se hicieron pruebas de concurrencia de usuarios, máximo 10 usuarios, se monitoreó el procesador y la memoria del servidor que convive con el servicio de TOMCAT para detectar si el uso de la nueva aplicación instalada generaba algún ciclo infinito en el procesador o bien se elevaba la memoria, para este número de usuarios no presentó ningún comportamiento degenerativo. Para realizar pruebas mas exhaustivas se recomienda realizar pruebas de stress para identificar hasta cuantos usuarios concurrentes nos podrá soportar la arquitectura dispuesta.
- ◆ Las pruebas de concurrencia también fueron útiles para detectar el comportamiento en cuanto a memoria y procesador del servidor de base de datos, no se detectaron problemas que se vieran reflejados tanto en el uso de procesador como de la memoria, el tiempo de respuesta de las consultas a la base de datos no fueron relevantes, pero se recomienda que cuando se hagan

las pruebas de stress, se prenda la herramienta de monitoreo de Sybase para detectar tiempos de respuesta de las consultas SQL y detectar que consultas son las que están tomando mas tiempo, y con esto analizar las posibles causas del retardo de estas.

6.2 Pruebas Realizadas por el Usuario

Los resultados de las pruebas de los coordinadores se resumen a continuación:

- ◆ Caso de uso: Realizar el Registro de un Alumno en el Sistema de Administración del Servicio Social.

Tipo de prueba	Observaciones	
Realizar el registro de un alumno capturando datos consistentes en los campos del formato de solicitud.	22/10/2004	No permite cambiar el jefe inmediato.
Realizar el registro de un alumno capturando datos inconsistentes en los campos del formato de solicitud (Por ejemplo letras en campos de teléfono, monto, código postal, etc.).	22/10/2004	El campo percepción mensual esta aceptando letras. Los campos localizador y pin no son campos obligatorios.

- ◆ Caso de uso: Autorizar Prealta de un Alumno en el Sistema de Administración del Servicio Social.

Tipo de prueba	Observaciones	
Consulta de alumnos en espera de autorización de prealta.	22/10/2004	OK
Abrir la solicitud elaborada por un alumno.	22/10/2004	OK
Autorizar/rechazar la prealta de un alumno.	22/10/2004	Se rechazó la prealta de un alumno. al volver a consultar la opción de autorización de prealta, aparecen aún los datos del alumno al que se le rechazó la prealta. Al activar enviar a cambios, despliega la página de resultado de la operación, pero no dice si fue exitosa o no.

- ◆ Caso de uso: Realizar Prebaja de un Alumno en el Sistema de Administración del Servicio Social.

Tipo de prueba	Observaciones	
Buscar el número de cuenta de un alumno no registrado en el sistema.	22/10/2004	Envía el mensaje: Ha ocurrido un error al obtener su información, Intente de nuevo mas tarde.
Buscar el número de cuenta de un alumno registrado en el sistema y que se encuentra en ejercicio de su SS.	22/10/2004	
Autorizar la prebaja de un alumno.	22/10/2004	Al introducir el número de cuenta y teclear enter, borra el número de cuenta y no ejecuta nada. Al teclear enter manda a Solicitud de Autorización de prestación de Servicio Social.

- ◆ Caso de uso: Ingresar Comentarios de un Alumno en el Sistema de Administración del Servicio Social.

Tipo de prueba	Observaciones	
Ingresar un comentario para un alumno que no ha registrado su SS o cuyo proceso no es administrado por el coordinador que intenta ingresar el comentario.	22/10/2004	Aparece: Resultado de la operación, pero no aparece ningún comentario.
Ingresar un comentario para un alumno que sí ha registrado su SS y cuyo proceso sí es administrado por el coordinador que intenta ingresar el comentario.	22/10/2004	Debido a que no fue posible ver los datos del alumno al que se le aprobó la tramitación del SS, no fue posible realizar la operación.

- ◆ Caso de uso: Generar Estadísticas de Alumnos en el Sistema de Administración del Servicio Social.

Tipo de prueba	Observaciones	
Validar opciones en la pantalla de filtros.	22/10/2004	No había el número de registros significativos para abarcar todas las opciones.
Generar estadísticas sobre los diferentes estatus de los alumnos durante el SS.	22/10/2004	En situación número de cuenta solo presenta número de cuenta bajo situación.

- ◆ Caso de uso: Generar Reportes de Alumnos en el Sistema de Administración del Servicio Social.

Tipo de prueba	Observaciones	
Validar opciones en la pantalla de filtros.	22/10/2004	No había el número de registros significativos para abarcar todas las opciones. Nota: Abreviar División de Ciencias Básicas por DCB.
Generar reportes sobre los diferentes estatus de los alumnos durante el SS.	22/10/2004	Pendiente

- ◆ Caso de uso: Actualizar Datos Personales de Alumnos en el Sistema de Administración del Servicio Social.

Tipo de prueba	Observaciones	
Realizar cambios en uno, varios o todos los datos personales del Alumno.	22/10/2004	OK. Nota: Donde se ve reflejado el cambio de datos del alumno.

- ◆ Caso de uso: Reiniciar Contraseña de un Alumno en el Sistema de Administración del Servicio Social.

Tipo de prueba	Observaciones	
Intentar reiniciar la contraseña de un Alumno cuando el usuario no tiene los privilegios para ello.	22/10/2004	OK.
Reiniciar la contraseña de un Alumno cuando el usuario si tiene los privilegios para ello.	22/10/2004	OK. Nota: Cambiar el mensaje del resultado de la operación cuando se desca cambiar la contraseña a un alumno de una división diferente a la del coordinador por Operación no válida.

- ◆ Caso de uso: Cambiar Contraseña del Usuario del Sistema de Administración del Servicio Social.

Tipo de prueba	Observaciones	
Realizar el cambio de contraseña del usuario firmado en el sistema proporcionando una contraseña incorrecta.	22/10/2004	OK.

Realizar el cambio de contraseña del usuario firmado en el sistema proporcionando una contraseña correcta.	22/10/2004	OK. Nota: Cambiar el mensaje del resultado de la operación cuando se desea resetear la contraseña a un alumno de una división diferente a la del coordinador por Operación no válida.
--	------------	--

- ◆ Caso de uso: Consultar Avance de un Alumno en el Sistema de Administración del Servicio Social.

Tipo de prueba	Observaciones	
Realizar búsqueda del un alumno inscrito al SS	22/10/2004	Pendiente.
Realizar la búsqueda de un alumno que no existe o no está inscrito en algún programa de SS	22/10/2004	OK.

- ◆ Caso de uso: Consultar Mensajes en el Sistema de Administración del Servicio Social.

Tipo de prueba	Observaciones	
Consultar los mensajes del usuario.	22/10/2004	OK.
Ver un mensaje en específico	22/10/2004	OK.
Eliminar mensaje	22/10/2004	OK.

6.3 Corrección de Programas

De acuerdo a los resultados obtenidos en las etapas de pruebas realizadas tanto por el equipo de trabajo como por los usuarios responsables del sistema, se realizaron las correcciones finales en los programas que proveen la funcionalidad a cada uno de los casos de uso del sistema con el fin de que la aplicación quedara funcionando de acuerdo a los requerimientos de los usuarios.

A continuación se describen las correcciones realizadas para cada uno de los casos de uso que presentaron alguna falla durante la etapa de pruebas del equipo de trabajo.

- ◆ Caso de uso: Realizar el Registro de un Alumno en el Sistema de Administración del Servicio Social.

Tipo de prueba	Falla presentada	Corrección
Realizar el registro de un alumno capturando datos consistentes en los campos del formato de solicitud.	El sistema no permitió capturar letras en el campo de "Número", lo cual es incorrecto ya que existen direcciones que cuyo "Número" incluye letras.	Se corrigió el componente Valida_Campos.js el cual es el que se encarga de verificar la consistencia de los datos de acuerdo al tipo de dato que se captura en cada campo del formato de la solicitud.

- ◆ Caso de uso: Autorizar la Prealta de un Alumno en el Sistema de Administración del Servicio Social.

Tipo de prueba	Falla presentada	Corrección
Rechazar la prealta de un alumno.	El sistema realizó el cambio de estatus correctamente pero no se validó la captura de los Motivos de Rechazo de la prealta del Alumno.	Se modificó el componente sass_AltaBaja.jsp para que no ejecute el cambio de estatus hasta que el campo de Motivos de Rechazo contenga información.

- ◆ Caso de uso: Autorizar el Alta de un Alumno en el Sistema de Administración del Servicio Social.

Tipo de prueba	Falla presentada	Corrección
Rechazar el alta de un alumno.	El sistema realizó el cambio de estatus correctamente pero no se validó la captura de los Motivos de Rechazo del alta del Alumno.	Se modificó el componente sass_AltaBaja.jsp para que no ejecute el cambio de estatus hasta que el campo de Motivos de Rechazo contenga información.

- ◆ Caso de uso: Actualizar el estado de los Informes Bimestrales de un Alumno en el Sistema de Administración del Servicio Social.

Tipo de prueba	Resultado Esperado	Observaciones
Actualizar Informe ocho días después o treinta días después de la fecha de entrega.	Se actualiza el estado del informe desfasando la fecha de entrega de los informes subsecuentes.	Se modificó el stored procedure que calcula las fechas de entrega de reportes.
Actualizar Informe treinta días después de la fecha de entrega.	No se permite actualizar el estado del informe, porque el alumno estará dado de baja.	Se agregó a la lógica de la búsqueda de los informes del alumno para que verifique si el estado actual del alumno es baja.

- ◆ Caso de uso: Descargar datos para la carta de liberación.

Tipo de prueba	Resultado Esperado	Observaciones
Bajar a la máquina del cliente el archivo con la información de la carta de liberación.	El estado del proceso actual del alumno sea Carta de Liberación. Se descarga el archivo CartasAjum.csv	Se modificó la consulta a la Base de Datos para obtener la información de la carta de los alumnos.

*Liberación y Mantenimiento del
Sistema*

CAPÍTULO 7

LIBERACIÓN Y MANTENIMIENTO DEL SISTEMA

7.1 Instalación del Sistema de Administración de Servicio Social.

7.1.1 Servidor de Base de Datos SYBASE

Para la creación de la Base de Datos y carga de los programas de Servicio Social en el Sistema de Administración de Servicio Social se implementaron *shells* de Unix que se ejecutarán en el Servidor de Base de Datos. Los shells se encuentran en la raíz del directorio del perfil del usuario de liberación y los archivos de carga se encuentran en el directorio “ss_arch_carga”. El procedimiento para instalar la Base de Datos se presenta a continuación.

Ejecutar el shell “ss_bd.sh”

Permite ejecutar los scripts de bd_usr.sql, tablas.sql y stored_proc.sql.

- ◆ **bd_usr.sql**, permite crear la Base de Datos y el usuario por default de la Base de Datos.
- ◆ **tablas.sql**, permite crear las tablas y los catálogos de la Base de Datos.
- ◆ **stored_proc.sql**, permite crear los procedimientos almacenados y triggers.

Ejecutar el shell “ss_carga.sh”

Permite ejecutar el script inicializa_tablas.sql y la carga de los catálogos mediante el comando bcp.

- ◆ **inicializa_tablas.sql**, permite crear las tablas temporales para cargar la información de los catálogos de institución, dependencia, área, carrera, coordinador, responsable, programa y carrera-programa.
- ◆ **Carga temporal de catálogos mediante el comando bcp**, carga los archivos de institución.txt, dependencia.txt, area.txt, coordinador.txt, responsable.txt, programa.txt, carrera_prg.txt en las tablas temporales.

Ejecutar el shell “ss_actualiza.sh”

Permite ejecutar el script actualiza_tablas.sql.

- ◆ **actualiza_tablas.sql**, tiene la lógica para insertar o actualizar los registros de las tablas de los catálogos a partir de las tablas temporales de catálogos que se cargaron.

Ejecutar el shell “ss actualiza_cb.sh”

Permite actualizar el catálogo de programas, a aquellos programas que corresponden a la División de Ciencias Básicas esto mediante una carga de archivo con el comando bcp y el script actualiza_cb.sql.

Prerrequisitos

Editar el archivo “ss_arch_carga/prgcb.txt” con las claves de programa que pertenecerán a la división de Ciencias Básicas de la siguiente manera:

Año de registro del programa|Clave de la Institución|Clave de la dependencia|Secuencial del programa|Identificador del programa

Es importante destacar que el separador de campo es el carácter “|”

- ◆ **actualiza_cb.sql**, tiene la lógica para actualizar los registros de la tabla del catálogo de programas a partir de la tabla temporal del catálogo de programas de la división de Ciencias Básicas.

Ejecutar el shell “ss usuarios.sh”

Permite actualizar el catálogo de alumnos susceptibles a realizar su Servicio Social, esto mediante una carga de archivo con el comando bcp y el script usuarios.sql.

Prerrequisitos

Editar el archivo “ss_arch_carga/usuario.txt” con la información referente a los alumnos:

Número de Cuenta|Clave de la Carrera|Nombre(s)|Apellido Paterno|Apellido Materno|Fecha de Nacimiento|Año de Ingreso|Sexo|Número de Créditos|Porcentaje de Avance|Promedio

Es importante destacar que el separador de campo es el carácter “|”

- ◆ **usuarios.sql**, tiene la lógica para actualizar los registros de la tabla del catálogo de alumnos a partir de la tabla temporal del catálogo de alumnos de la Facultad de Ingeniería.

7.1.2 Servicio Tomcat

Para la creación del proyecto del Sistema de Administración de Servicio Social en el Servidor Tomcat, es necesario copiar la estructura de Directorios generada para este proyecto (**proySS**) en la ruta de publicación del Servidor Tomcat (**./webapps**), así como realizar los cambios correspondientes en los archivos de configuración del Servicio de Tomcat como se muestra a continuación:

Paquete de java “jConnect”

Copiar el archivo **jconn2.jar** en el directorio **./tomcat/common/lib**, el cual contiene las clases necesarias para permitir realizar la conexión a la Base de Datos Sybase.

Archivo de Configuración “server.xml”

◆ Definición del nuevo recurso de Base de Datos y sus parámetros

Dentro de la sección `<GlobalNamingResources></GlobalNamingResources>` se deberá insertar el siguiente código.

```
<Resource name="sybase" scope="Shareable" type="javax.sql.DataSource">
</Resource>
<ResourceParams name="sybase">
  <parameter>
    <name>validationQuery</name>
    <value>select count(*) from ss_division</value>
  </parameter>
  <parameter>
    <name>username</name>
    <value>ss_usr</value>
  </parameter>
  <parameter>
    <name>url</name>
    <value>jdbc:sybase:Tds:IP_BD/ss_serv_soc?jconnect_version=5</value>
  </parameter>
  <parameter>
    <name>password</name>
    <value>ss_usr</value>
  </parameter>
  <parameter>
    <name>maxActive</name>
    <value>20</value>
  </parameter>
  <parameter>
    <name>maxWait</name>
    <value>5000</value>
  </parameter>
  <parameter>
    <name>driverClassName</name>
    <value>com.sybase.jdbc2.jdbc.SybDriver</value>
  </parameter>
  <parameter>
    <name>maxIdle</name>
    <value>10</value>
  </parameter>
</ResourceParams>
```


Es importante señalar que en la cadena de conexión mostrada en la línea:

```
<value>jdbc:sybase:Tds:IP_BD/ss_serv_soc?jconnect_version=5</value>
```

Se deberá reemplazar la palabra **IP_BD** por el nombre o la dirección IP del Servidor de Base de Datos.

◆ **Crear el contexto del proyecto y asociar el recurso global al proyecto.**

Dentro de la sección `<Hosts></Hosts>` se deberá insertar el siguiente código:

```
<Context className="org.apache.catalina.core.StandardContext" cachingAllowed="true"
charsetMapperClass="org.apache.catalina.util.CharsetMapper" cookies="true" crossContext="false" debug="0"
displayName="Servicio Social" docBase="/opt/tomcat/webapps/proySS"
mapperClass="org.apache.catalina.core.StandardContextMapper" path="/proySS" privileged="true" reloadable="true"
useNaming="true" wrapperClass="org.apache.catalina.core.StandardWrapper">
  <ResourceLink global="sybase" name="jdbc/ssDB"/>
</Context>
```

◆ **Reiniciar el servicio de Tomcat.**

7.1.3 Diagrama de Distribución de la Aplicación

Una vez probado el sistema y validado por los usuarios, se procedió a instalar el SASS en los servidores de la USECAD de la FI, esto de acuerdo al siguiente diagrama de distribución.

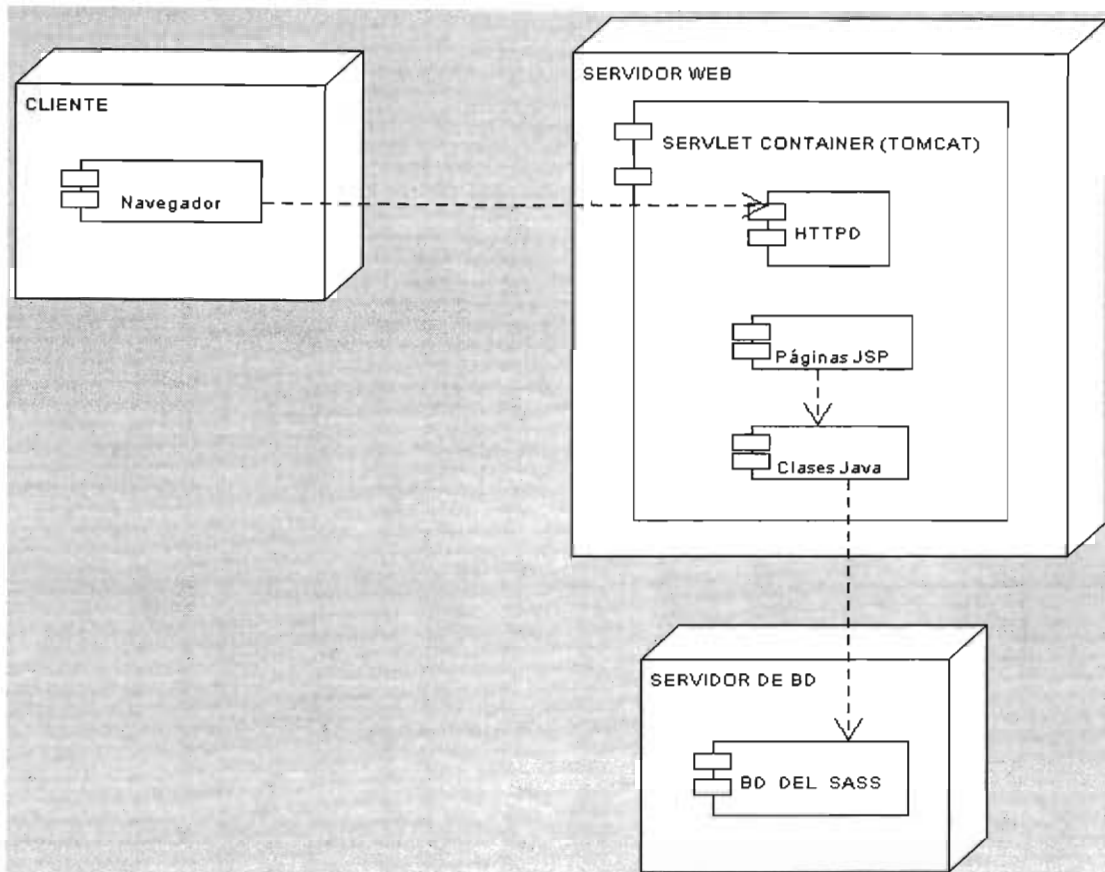


Figura 7.1. Diagrama de Despliegue del SASS

7.2 Mantenimiento del Sistema

Cuando un sistema de información concluye las etapas de pruebas e instalación, es importante determinar un plan de mantenimiento, esto con el fin de garantizar en la medida de lo posible el funcionamiento óptimo del sistema.

Los puntos que conforman el plan de mantenimiento deben considerar también la prevención y corrección de posibles fallas así como necesidades de mejora que pueden presentarse en el futuro. Dependiendo del origen o tipo de estas fallas y necesidades se determinará la forma de resolverlas.

Con el fin de que las diferentes necesidades de mantenimiento sean atendidas oportunamente, el equipo de trabajo de este proyecto de tesis, hizo entrega del Manual Técnico correspondiente

en el cual se encuentra toda la información del sistema como: Arquitectura física y lógica del sistema, estructura de base de datos, estructuración del código del sistema y descripción del funcionamiento.

7.2.1 Tipos de Mantenimiento

Para el Sistema de Administración del Servicio Social, se determinan los siguientes tipos de mantenimiento:

- ◆ **Correctivo:** Este tipo de mantenimiento se origina por fallas precisas en el sistema cuyo efecto se refleja directamente en el funcionamiento de la aplicación y en la operación de los usuarios. El mantenimiento correctivo puede cubrir también acciones que tengan como objetivo mejorar el rendimiento y la eficiencia del producto de software.

Para efectos de garantía en la funcionalidad del producto de software entregado a la Facultad de Ingeniería, el equipo de trabajo de este proyecto se hace responsable de las fallas presentadas en el producto y que correspondan a la funcionalidad original, esto mientras que ningún componente de software haya sido modificado por alguna persona ajena a este proyecto de tesis.

- ◆ **Preventivo:** En todo producto de software resulta conveniente definir acciones que prevengan posibles problemas que afecten el desempeño de la aplicación, por lo que, el mantenimiento preventivo se enfoca en la definición de acciones que deben ejecutarse antes de que se presenten problemas que pueden evitarse en el sistema.

En el SASS, parte importante del mantenimiento preventivo que debe realizarse es el mantenimiento de los catálogos que la DGOSE envía a la FI con el fin de actualizar la información de los programas, dependencias, instituciones, etc. que intervienen en el ejercicio del SS de los alumnos de la FI.

- ◆ **Evolutivo:** El mantenimiento evolutivo surge de las nuevas necesidades del usuario originadas por cambios en sus procedimientos de trabajo. Este tipo de mantenimiento atiende las incorporaciones, modificaciones y eliminaciones necesarias para cubrir dicho cambios en la funcionalidad del sistema.

Las personas responsables de realizar cualquiera de estos tipos de mantenimiento serán designadas por las entidades correspondientes dentro de la Facultad de Ingeniería, estas personas pueden ser servidores sociales, becarios o personal de planta de la Facultad.

Conclusiones



CONCLUSIONES

La ingeniería de software tiene una importancia inmensurable para iniciar, mantener y concluir con éxito el ciclo de vida de los sistemas, uno de los procesos más importantes es el entendimiento correcto de la necesidad, ya que esto, nos da la pauta para decidir cuál es la mejor solución de calidad que se puede implementar.

Para llevar a cabo la ingeniería de software existen dos enfoques de metodología que podemos destacar: la estructurada y la orientada a objetos. De ambos enfoques se ramifican diferentes metodologías, las diferencias entre unas y otras es básicamente el énfasis que recalcan sobre cada una de las fases de la metodología, el modelado y la documentación entregable de las mismas, pero todas convergen hacia el mismo objetivo, dar una solución correcta y de calidad al usuario final.

En cuanto a las metodologías orientadas a objetos, UML estandarizó la forma de representar los sistemas definiendo diferentes diagramas que de acuerdo a la metodología unos se utilizan y otros no.

El paradigma y las metodologías orientadas a objetos desde hace algunos años han tomado mucho poder debido a las ventajas que éstas proveen, la reutilización, el proceso iterativo sobre las fases de la metodología para el mantenimiento de mejora continua, la definición previa de la arquitectura adecuada y la relación estrecha con diferentes herramientas de modelado UML que permite generar código para la fase de implementación del sistema.

La metodología adoptada para el desarrollo del sistema SASS es el Proceso Unificado, debido a que éste ofrece una forma coordinada de trabajar e integrar las diferentes facetas del ciclo de vida del sistema. El proceso Unificado hace énfasis en cuatro aspectos: arquitectura del sistema (arquitectura del negocio, arquitectura de aplicación, arquitectura de información y arquitectura tecnológica), proceso iterativo e incremental, centrado en la arquitectura, guiado por casos de uso y medición de riesgos.

Primeramente, el equipo de trabajo se enfocó por conocer la arquitectura del sistema, investigando cuáles eran los procedimientos que llevaban a cabo en forma manual los involucrados, que sistemas ya existían y con que tipo de información interactúan, abarcando con esto el conocimiento de la arquitectura del negocio y la arquitectura de la información, seguido de este conocimiento y dada la necesidad de una solución vía Internet dado el tipo de interacción de los involucrados en el sistema, se decidió la arquitectura tecnológica, adoptando como solución para el desarrollo del sistema a la plataforma de J2EE como solución integral para SASS, adoptando páginas JSP como lógica de presentación, Java Beans como lógica del negocio y objetos del dominio, y dejando abierto el tipo de manejador de base de datos relacional como repositorio y parte de la lógica de negocio; en cuanto a infraestructura, se adecuó a la ya existente en la institución. La recopilación de requerimientos tanto del análisis como del diseño y las pruebas del usuario fueron guiados y mantenidos mediante casos de uso, el proceso se hizo iterativo mientras se hacían reuniones con el usuario para definir las mejoras que debía tener el sistema, todos estos controles de cambio se recopilaron en forma de minutas y se actualizaron en los casos de uso y aplicaciones relacionadas, los cuales, ayudaron a mantener vigentes los acuerdos con los usuarios.

La tecnología JSP elegida para la construcción del sistema, fue seleccionada por el equipo de trabajo ya que representa una de las mejores opciones proporcionadas por Java para la

construcción de sistemas en ambiente Web, además de representar una excelente oportunidad de aprendizaje para el desarrollo de este tipo de aplicaciones. Los JSP's brindan la oportunidad de separar la lógica de presentación de la lógica del negocio lo cual es una gran ventaja ya que es posible realizar cambios en los procesos del sistema sin afectar la parte gráfica del mismo.

Con las herramientas utilizadas durante el proyecto se obtuvieron grandes beneficios pero además se corrieron varios riesgos, uno los cuales fue la curva de aprendizaje tanto en el conocimiento como en la aplicación de la metodología, el lenguaje de modelado y la tecnología de desarrollo utilizada, la falta de comunicación con el usuario, el tiempo de respuesta del usuario para dar comentarios sobre el sistema y algunas vulnerabilidades de seguridad del sistema que no se consideraron. Los riesgos mencionados afectaron directamente al tiempo de construcción y liberación del sistema.

El sistema SASS desarrollado como un proyecto de tesis, pretende colaborar con la Facultad para automatizar los procesos manuales que consumen tiempo y recursos significativos en las labores de la administración escolar. Este sistema representa el inicio de una aplicación Web que puede ir creciendo de acuerdo a las nuevas necesidades e ideas que surjan durante la utilización y conocimiento de la misma, así como de los nuevos requerimientos de la administración escolar.

Los desarrolladores del sistema y del presente trabajo de tesis, entregaron a USECAD todo el código fuente de la aplicación, así como el manual técnico correspondiente, esto con el fin de que cualquier otro tesista o prestador de Servicio Social pueda modificar la aplicación o incorporar nueva funcionalidad, impulsando así el crecimiento del sistema, el cual a su vez, ofrecerá mayores beneficios a los usuarios, tanto coordinadores como alumnos, quienes recibirán directamente las ventajas de un sistema que les ayude a automatizar los procesos que realizan cotidianamente y que se encuentren relacionados con el Servicio Social.

Actualmente el sistema solo almacena trámites de alumnos que recién inician su Servicio Social, por lo que muchos de los comentarios expuestos por los Coordinadores, fueron la posibilidad de incluir la información de los alumnos que se encuentran en proceso, o bien, hayan terminado su trámite, por lo que este desarrollo de migración de datos podría considerarse como un nuevo proyecto a desarrollar.

Apéndices



APÉNDICE A

MINUTAS DE LAS REUNIONES CON EL USUARIO

A.1 Minuta Fase I de Levantamiento de Información.

Asistentes:

Cliente	Proveedor
Ing. Gonzalo Condado Bernal. Coordinador General de Servicio Social de la Facultad de Ingeniería.	Galicia González Gpe. Miriam y Bruno David Vázquez Chávez. Integrantes del Equipo de Trabajo de la Tesis.
Ing. Elí Israel Hernández García. Coordinador de Servicio Social de la División de Ciencias Básicas.	
Ing. Heriberto Esquivel Castellanos. Coordinador de Servicio Social de la División de Ingeniería Civil, Topográfica y Geodésica.	
Ing. José Arturo Origel Coutiño. Coordinador de Servicio Social de la División de Ingeniería Eléctrica.	

Puntos Tratados:

- ◆ Procedimiento del alumno para realizar su Servicio Social.
- ◆ Requisitos del alumno para el trámite de Servicio Social.
- ◆ Requisitos del alumno para la terminación del Servicio Social.
- ◆ Medios de difusión de Programas de Servicio Social.
- ◆ Autoridades que regulan el trámite de Servicio Social de la Facultad de Ingeniería.
- ◆ Tiempo para la realización de Servicio Social.
- ◆ Apoyo económico para alumnos que ejercen su Servicio Social.
- ◆ Horarios de Prestación de Servicio Social.
- ◆ Sistemas de Software que lleven el control del alumno en el ejercicio de su Servicio Social.

Acuerdos:

- ◆ Procedimiento del alumno para realizar su Servicio Social.

Alta del Alumno

1. Consulta los programas de Servicio Social y elige el más apropiado, la forma en cómo se publican estos programas es a través de cartelones que se pegan en las instalaciones de la Facultad de Ingeniería o bien en cada una de las Divisiones de la misma.

2. Dirigirse al lugar en donde se está promoviendo dicho programa y pedir una carta de aceptación para la prestación del Servicio Social.
3. Dirigirse a la División de Ingeniería a la que pertenece el alumno y llenar la Solicitud de Autorización de Prestación del Servicio Social e Información Estadística, y entregar con ésta, la carta de aceptación e Historial Académico.

Aprobación de Alta del alumno por el Coordinador de Servicio Social de la División

4. Verifica la carta de Aceptación de prestación de Servicio Social del Alumno, la Solicitud de Autorización de Prestación del Servicio Social e Información Estadística y que en el Historial Académico cumpla con el porcentaje de créditos requerido.
5. Se establecen las fechas de entrega de reportes de avance.

Aprobación de Alta del alumno por el Coordinador General de Servicio Social de la Facultad de Ingeniería

6. Verifica la carta de Aceptación de Prestación de Servicio Social del Alumno, la Solicitud de Autorización de Prestación del Servicio Social e Información Estadística y que en el Historial Académico cumpla con el porcentaje de créditos requerido.
7. Da de alta al alumno en el Sistema de Control de Servicio Social que tienen actualmente.

Entrega de Reportes de Avance

8. Cada bimestre el alumno elaborará un reporte de actividades, el cual deberá llevar el visto bueno de la dependencia donde presta el Servicio Social.
9. Entregar el reporte en la unidad de Servicio Social de la División correspondiente.
10. En la entrega del penúltimo reporte la unidad de Servicio Social se le entrega al alumno el formato de terminación de Servicio Social.

Terminación de Servicio Social del Alumno.

11. Entrega con el último reporte la carta de terminación de Servicio Social.
12. Se le indica al alumno cuando puede pasar por su carta de Liberación.

Aprobación de Terminación del alumno por el Coordinador de Servicio Social de la División.

13. Verifica la carta de terminación de Servicio Social.
14. Prepara todos los documentos del alumno para enviárselos a la Coordinación General de Servicio Social de la Facultad de Ingeniería.

Aprobación de Terminación del alumno por el Coordinador General de Servicio Social de la Facultad de Ingeniería.

15. Verifica la carta de terminación de Servicio Social y los reportes bimestrales entregados.
16. Imprime y entrega la carta de Liberación del Alumno.

◆ Requisitos del alumno para el trámite de Servicio Social

1. Obtener la clave del Programa de Servicio Social.
2. Carta de Aceptación de Servicio Social.
3. Solicitud de Autorización de Prestación de) Servicio Social e Información Estadística.

4. Historial Académico.

- ◆ Requisitos del alumno para la terminación del Servicio Social

1. Cumplir con un mínimo de seis meses y máximo de dos años en el programa inscrito.
2. Entregar los reportes bimestrales correspondientes al tiempo de ejercicio de Servicio Social
3. Obtención de la carta de terminación de Servicio Social de la Dependencia o Empresa en donde se haya ejercido el Servicio Social.

- ◆ Medios de difusión de Programas de Servicio Social

Rotafolios y carteles expuestos en las diferentes Instalaciones y Divisiones de Ingeniería de la Facultad de Ingeniería.

- ◆ Autoridades que regulan el trámite de Servicio Social de la Facultad de Ingeniería

1. Coordinador general de Servicio Social de la Facultad de Ingeniería.
2. Coordinadores de las diferentes Divisiones de la Facultad de Ingeniería.
3. Responsable del Programa de Servicio Social.

- ◆ Tiempo para la realización de Servicio Social

El tiempo mínimo es de seis meses y el máximo dos años.

- ◆ Apoyo económico para alumnos que ejercen su Servicio Social

Algunos programas tienen asociados subsidiarias, instituciones o bien la misma unidad que solicita el apoyo de un prestador, la posibilidad de dar apoyo económico al alumno la cual puede tener cualquiera de los siguientes conceptos: sueldo, honorarios, ayuda económica, beca o cualquier otra forma de ayuda, ya sea en capacitación, material de estudio, etc.

- ◆ Horarios de Prestación de Servicio Social

Los horarios de prestación de Servicio Social pueden ser estipulados por la Institución que lo solicita o bien por un acuerdo entre la Institución y el alumno.

- ◆ Sistemas de Software que lleven el control del alumno en el ejercicio de su Servicio Social

La Administración Escolar de la Facultad de Ingeniería ha estado utilizando dos sistemas para llevar el control del Servicio Social de los Alumnos, uno que fue desarrollado en Dbase por alumnos de la Facultad de Ingeniería, el cual le permite llevar el control de las altas, bajas, cambios, terminaciones de los alumnos de la Facultad de Ingeniería así como la generación de reportes y estadísticas por diferentes criterios de interpretación y un módulo de Mantenimiento del Sistema; y otro, desarrollado en Visual Basic por la DGOSE, que lo utilizan para consultar los programas de Servicio Social vigentes en la DGOSE.

Cabe mencionar que este último sistema provee la misma funcionalidad que el de Dbase pero solo se utiliza la parte de carga de Programas de Servicio Social.

A.2 Minuta Fase II de Levantamiento de Información.

Asistentes:

Cliente	Proveedor
Ing. Gonzalo Condado Bernal. Coordinador General de Servicio Social de la Facultad de Ingeniería.	Galicia González Gpe. Miriam y Bruno David Vázquez Chávez. Equipo de Trabajo.
Ing. Elí Israel Hernández García. Coordinador de Servicio Social de la División de Ciencias Básicas.	
Ing. Heriberto Esquivel Castellanos. Coordinador de Servicio Social de la División de Ingeniería Civil, Topográfica y Geodésica.	
Ing. José Arturo Origel Coutiño. Coordinador de Servicio Social de la División de Ingeniería Eléctrica.	

Puntos Tratados:

- ◆ Módulos del Sistema de Control de Servicio Social Actual
- ◆ Fuentes de Información de los Programas de Servicio Social
- ◆ Autorización de Alta de Servicio Social
- ◆ Entrega de Reportes
- ◆ Validación de Reportes Bimestrales
- ◆ Rangos de fechas para entrega de Reportes
- ◆ Autorización de Terminación de Servicio Social
- ◆ Carta de Liberación

Acuerdos:

- ◆ Módulos del Sistema de Control de Servicio Social Actual

Módulo de Altas.

Permite dar de alta a un alumno en un programa de Servicio Social con la información establecida en el **Formato de Solicitud de Autorización de Prestación de Servicio Social e Información Estadística**.

Cabe mencionar que para que este proceso se lleve a cabo, es necesario que la documentación correspondiente para el alta de Servicio Social del alumno se le envíe a la Coordinación General de Servicio Social de la Facultad de Ingeniería.

Módulo de Bajas.

Permite dar de baja a un alumno dentro de un programa de Servicio Social, lo que implica que solo se marque al alumno como que se dio de baja y no se borra de la base de datos, esto con la

finalidad de permitirle al sistema obtener estadísticas sobre los alumnos que se dan de baja durante cierto periodo.

El procedimiento de baja se hace de la siguiente manera:

Se hace la búsqueda del alumno por número de cuenta.

Al solicitar la Baja del alumno se deberá mostrar la información general del Alumno para poder ser verificada.

Cabe mencionar que para que este proceso se lleve a cabo es necesario que el formato de Notificación de Bajas de Servicio Social emitida por cada División de Ingeniería se le envíe a la Coordinación General de Servicio Social de la Facultad de Ingeniería.

Módulo de Modificaciones.

Permite realizar modificaciones sobre los datos generales de los alumnos inscritos en algún programa de Servicio Social.

El procedimiento de la modificación se hace de la siguiente manera:

Se hace la búsqueda del alumno por número de cuenta.

Se pueden hacer modificaciones sólo a los datos generales del alumno, como son:

Nombres y Apellidos.

Dirección.

Colonia.

Código Postal.

Estado, Delegación o Municipio.

Teléfono.

Edad.

El único campo que no se puede modificar es el número de cuenta del alumno.

Cabe mencionar que para que este proceso se lleve a cabo es necesario que se notifique la modificación por cada División de Ingeniería a la Coordinación General de Servicio Social de la Facultad de Ingeniería.

Módulo de Terminación.

Permite validar la terminación del Servicio Social del alumno, así como imprimir la carta de Liberación.

El procedimiento para la terminación se hace de la siguiente manera:

Se hace la búsqueda del alumno por número de cuenta.

Se edita la fecha de término del Alumno.

Se imprime la carta de terminación.

Cabe mencionar que para que este proceso se lleve a cabo es necesario que tanto los Reportes bimestrales del alumno como la carta de Terminación de Servicio Social se le envíen a la Coordinación General de Servicio Social de la Facultad de Ingeniería.

Módulo de Estadísticas.

Permite aprovechar la información almacenada en la Base de Datos para emitir estadísticas sobre los diferentes criterios mostrados a continuación:

Alumnos activos (Generación-Carrera)
Alumnos activos (Carrera-Nombre)
Alumnos activos (Situación-Número de cuenta)
Bajas (Generación-Carrera)
Alumnos que terminaron (Carrera-Nombre)
Inicios en programas int-ext (Carrera-Nombre)
Terminaciones en programas int-ext (Carrera-Nombre)
Inicios en programas int-ext (Carrera-Fecha)
Terminaciones en programas int-ext (Carrera-Fecha)
Emisión de cartas de Terminación.
Terminaciones en programas int-ext (División)
Programas vigentes.

Módulo de Mantenimiento.

Permite actualizar los catálogos de Dependencias y Programas de Servicio Social, así como permitir hacer respaldos y recuperaciones de información de la Base de Datos.

◆ Fuentes de Información de los Programas de Servicio Social

Con la Base de Datos del programa elaborado por la DGOSE es posible obtener la información necesaria sobre los programas de Servicio Social vigentes en formato electrónico.

◆ Autorización de Alta de Servicio Social

Para que el alta de un alumno se lleve a cabo se debe pasar por dos filtros:

Coordinador de Servicio Social de la División de Ingeniería correspondiente a la del alumno.
Coordinador General de Servicio Social de la Facultad de Ingeniería.

◆ Entrega de Reportes

El alumno deberá dirigirse al departamento de Servicio Social de la división correspondiente a su carrera.

◆ Validación de Reportes Bimestrales

El encargado de validar el contenido y formato de los reportes bimestrales entregados por los alumnos es una persona delegada por el Coordinador de Servicio Social de la división correspondiente.

◆ Rangos de fechas para entrega de Reportes

Los reportes se deben entregar cada dos meses, a partir de la fecha de aceptación del Servicio Social. En caso de que el reporte se rechace, se le da al alumno cierto tiempo mas para que entregue su reporte, este tiempo lo establece el departamento de Servicio Social de cada División.

◆ Autorización de Terminación de Servicio Social

Para que la terminación de un alumno se lleve a cabo se debe pasar por dos filtros:

Coordinador de Servicio Social de la División de Ingeniería correspondiente a la del alumno.
Coordinador General de Servicio Social de la Facultad de Ingeniería.

◆ Carta de Liberación

La carta de Liberación la entrega el Coordinador General de Servicio Social de la Facultad de Ingeniería, aproximadamente después de 15 días de que el alumno entregó la carta de Terminación de Servicio Social a la Coordinación de Servicio Social de la división correspondiente.

A.3 Minuta de la Reunión Celebrada el 9 de Marzo de 2004.

Asistentes:

Cliente	Proveedor
Ing. Gonzalo Condado Bernal. Coordinador General de Servicio Social de la Facultad de Ingeniería.	Galicia González Gpe. Miriam y Bruno David Vázquez Chávez. Equipo de Trabajo.

Puntos Tratados:

- ◆ Revisión de los avances en el desarrollo del sistema en los módulos de registro, prealta, alta, prebaja, baja y estadísticas.
- ◆ Análisis del módulo de estadísticas solicitadas por el usuario y comparación con el módulo de reportes propuesto por el equipo de trabajo.

Acuerdos:

- ◆ Solicitó que se desarrollara una nueva opción para que el Alumno pueda buscar un Programa por medio de la clave de éste y que posteriormente a esta búsqueda pueda realizar su registro.
- ◆ Solicitó que los procesos de prealta, alta y baja se pudieran realizar por bloque seleccionando por medio de un check box todos los registros de los alumnos a los que se desee aplicar el movimiento.
- ◆ Después de revisar los reportes propuestos, el usuario solicitó que se desarrollara este nuevo módulo, con el cual podrá visualizar en un mismo reporte, la información de dos o más consultas.
- ◆ Los reportes que conforman el módulo son los siguientes:

Pendientes:

- ◆ Ninguno.

A.4 Minuta de la Reunión Celebrada el 2 de Abril de 2004.

Asistentes:

Cliente	Proveedor
Ing. Gonzalo Condado Bernal. Coordinador General de Servicio Social de la Facultad de Ingeniería.	Galicia González Gpe. Miriam y Bruno David Vázquez Chávez. Equipo de Trabajo.
Ing. Elí Israel Hernández García. Coordinador de Servicio Social de la División de Ciencias Básicas.	
Ing. Heriberto Esquivel Castellanos. Coordinador de Servicio Social de la División de Ingeniería Civil, Topográfica y Geodésica.	
Ing. José Arturo Origel Coutiño. Coordinador de Servicio Social de la División de Ingeniería Eléctrica.	
Ing. Víctor Manuel Vázquez Huarota. Coordinador de Servicio Social de la División de Ingeniería Mecánica e Industrial.	
Ing. Alfredo Victoria Morales. Coordinador de Servicio Social de la División de Ingeniería en Ciencias de la Tierra.	

Puntos Tratados:

- ◆ Revisión de los avances en el desarrollo y modificaciones del sistema en los módulos de Registro, Preacta, Alta, Prebaja, Baja, Estadísticas, Reportes y Actualización de Datos.
- ◆ Solicitud de nuevos requerimientos.

Acuerdos:

- ◆ El usuario solicitó las siguientes modificaciones a los módulos ya desarrollados:
 1. Solicitó que en el módulo de Registro del Alumno, en el formato de Solicitud de Autorización de Prestación de Servicio Social e Información Estadística se encuentren previamente capturados los siguientes datos: Año de ingreso, sexo, créditos, promedio y avance.
 2. También, en el módulo de registro, se requirió que el campo de Jefe Inmediato en la sección Programa pueda ser modificado ya que el nombre de la persona que se encuentra capturado no siempre corresponde con el nombre real, o que en su defecto exista alguna forma de cambiar o seleccionar el nombre correcto de la persona que es el Jefe Inmediato del Programa.
 3. Se solicitó que en el módulo de Actualización de Datos sólo puedan actualizarse los datos personales del Alumno como son: Dirección, teléfono y correo electrónico.

4. Como parte del modo de operación en los procesos de Servicio Social, se requirió que se distinguieran los programas de la División de Ciencias Básicas de la FI ya que estos pueden recibir a Alumnos de cualquier carrera. Para los Alumnos que realicen su Servicio Social en uno de los programas que ofrece esta división, el Coordinador de SS de Ciencias Básica es el que administrará el proceso de Servicio Social del Alumno.
 5. Se solicitó eliminar la opción de Prealta por búsqueda del número de cuenta del Alumno.
 6. Se solicitó eliminar las opciones para aplicar en bloque los procesos de Prealta, Alta y Baja que fueron solicitados en la reunión del 9 de Marzo de 2004.
 7. La estadística de Emisión de Cartas debe estar disponible sólo para el Coordinador General de Servicio Social.
 8. Se indicó no desarrollar las estadísticas de Programas Vigentes y Acumulado Anual de Inicios de 1980 a la fecha.
 9. En el módulo de Estadísticas, agregar "Informes Aceptados" (columnas: carrera y cantidad), "Alumnos sin Beca" (columnas nombre, correo y teléfono) y "Observaciones" (columnas: cuenta y texto).
- ♦ El usuario solicitó el desarrollo de los siguientes nuevos requerimientos no contemplados en el inicio del proyecto:
1. Solicitó que cuando un usuario ingrese al sistema, en la pantalla principal se despliegue un conjunto de reglas, derecho y obligaciones de los usuarios.
 2. Requiere el desarrollo de un módulo para Cambio de Contraseña por parte del Alumno como medida de seguridad y otro módulo para Reinicio de Contraseña, con el cual los Coordinadores podrán reiniciar la contraseña del Alumno a petición de éste en caso de olvido.
 3. Se solicitó desarrollar un módulo de Agregar Notas en el cual los coordinadores ingresarán comentarios adicionales sobre los alumnos que realizan el Servicio Social.

Pendientes:

- ♦ Para cumplir el punto 1 de la lista de modificaciones solicitadas por el usuario mencionada en la sección Acuerdos, el Ing. Gonzalo Condado Bernal solicitará a la USECAD que los datos de: año de ingreso, sexo, créditos, promedio y avance se agreguen al conjunto de datos que esta entidad entregará al Equipo de Trabajo del sistema.

A.5 Minuta de la Reunión Celebrada el 19 de Mayo de 2004.

Asistentes:

Cliente	Proveedor
Ing. Gonzalo Condado Bernal. Coordinador General de Servicio Social de la Facultad de Ingeniería.	Ing. Alejandro Velásquez Mena. Director de Tesis.
Ing. Elí Israel Hernández García. Coordinador de Servicio Social de la División de Ciencias Básicas.	Galicía González Gpc. Miriam y Bruno David Vázquez Chávez. Equipo de Trabajo.
Ing. Heriberto Esquivel Castellanos. Coordinador de Servicio Social de la División de Ingeniería Civil, Topográfica y Geodésica.	
Ing. José Arturo Origel Coutiño. Coordinador de Servicio Social de la División de Ingeniería Eléctrica.	

Puntos Tratados:

10. Revisión de la Funcionalidad General del Sistema.

Acuerdos:

11. Para los movimientos de rechazo de Prealta y Alta, deben existir dos modalidades. La primera modalidad consistirá en que el Coordinador realizará un rechazo parcial y en los motivos de rechazo especificará los cambios que debe hacer el Alumno en su solicitud para que su registro sea aceptado, estas indicaciones del Coordinador son enviadas al Alumno por medio de un mensaje en el sistema. La segunda modalidad consistirá en que el Coordinador, después de revisar la solicitud decide hacer un rechazo definitivo de la solicitud, indicando también los motivos de rechazo definitivo para que éstos sean enviados al Alumno por medio de un mensaje; cabe mencionar que en este caso el Alumno deberá comenzar de nuevo su trámite, es decir no podrá modificar la solicitud llenada previamente.
12. En el campo de "Duración" de la Solicitud de Prestación de Servicio Social, sólo pueden ingresarse valores de 6 a 24 (meses) y con base en este valor, el campo de "Horas a la Semana" se calculará automáticamente.
13. Cuando en el campo "Remuneración" el Alumno elija la opción "No Remunerado", se inhabilitará el campo de "Sueldo" y su valor será igual a cero.
14. En el Campo de "Número" que forma parte de la dirección del Alumno se deben aceptar también letras.
15. En los módulos de Alta, Baja y Liberación, donde el Coordinador General de SS consulta la pantalla de las Divisiones de Ingeniería se mostrarán a la izquierda de cada División, la cantidad de Alumnos que se encuentran pendientes para el movimiento en cuestión.

16. El módulo de "Consultas" se agregará como una opción más al menú de opciones para los Coordinadores y éstos buscarán los movimientos del Alumno por medio del número de cuenta.
17. En el módulo de "Revisión de Informes", se realizarán las adecuaciones necesarias para controlar las siguientes situaciones: El Alumno tiene siete días a partir de su fecha de entrega para entregar su reporte de actividades, por lo que si el Alumno lo entrega en el intervalo del 8vo. al 30vo día, entonces el sistema desfazará automáticamente las siguientes fechas de entrega de documentos para cualquiera de los trámites o etapas posteriores en el ejercicio de su Servicio Social. Si el alumno entrega su reporte después del día núm. 30 a partir de la fecha de entrega estipulada, entonces el sistema realiza una baja automática irreversible.
18. Para el módulo de "Terminación", el mismo criterio descrito anteriormente aplica para la entrega de la carta de terminación. Además en el proceso de terminación no aplica el movimiento de rechazo por lo que será eliminado de la funcionalidad del sistema.
19. Se desarrollará nueva funcionalidad que permita a los coordinadores realizar modificaciones a la información correspondiente a la administración del Servicio Social de los Alumnos.

Pendientes:

- ♦ El Coordinador General de Servicio Social de la FI, revisará el estatus de la solicitud de información de los alumnos que se realizó a la USECAD, esto de acuerdo a los requerimientos de la reunión del 2 de Abril de 2004.

APÉNDICE B

VISIÓN GENERAL DE UML

B.1 UML (Unified Modeling Language)

El UML es un lenguaje notacional destinado a los sistemas de modelado que utilizan conceptos orientados a objetos.

Nació en 1994 por iniciativa de Grady Booch y Jim Rumbaugh quienes laboraban en Rational Software Corporation y combinaron sus dos famosos métodos diseñados individualmente: el de Booch y el OMT (Object Modeling Technique) de Rumbaugh. Un año después, Ivar Jacobson, creador del método OOSE (Object Oriented Software Engineering), ingresó a Rational, uniendo sus esfuerzos a los antes mencionados.

Los anteproyectos del UML circularon rápidamente en la industria del software generándose así considerables modificaciones. Cuando los diversos corporativos vieron que el UML era útil a sus propósitos, se conformó un consorcio del UML de cuyos miembros puede destacarse a DEC, Hewlett-Packard, Intellicorp, Microsoft, Oracle, Texas Instruments y Rational.

El OMG (Grupo de Administración de Objetos), realizó una petición para definir un lenguaje y una notación estándar del lenguaje de construcción de modelos, por lo que en 1997 el consorcio produjo la versión 1.0 del UML y lo propuso como candidato.

El consorcio aumentó y generó la versión 1.1 del UML, misma que puso nuevamente a consideración del OMG. Dado que sus creadores representan métodos muy difundidos de la primera generación del análisis y diseño orientado a objetos, este lenguaje recibió la aprobación del grupo a finales de 1997.

El OMG se encargó de la conservación del UML y produjo otras dos revisiones en 1998. El UML ha llegado a ser el estándar de facto en la industria del software, y su evolución continúa.

El UML estandariza los artefactos y la notación, pero no define un proceso oficial de desarrollo.

B.2 Casos de Uso

Un caso de uso es un documento que describe la secuencia de eventos que se desencadenan cuando una entidad externa realiza una acción en el sistema para completar un proceso [5]. A las entidades que inician tales secuencias se les conoce como actores y pueden ser: una persona, otro sistema, una parte del hardware o el paso del tiempo. El resultado de la secuencia de eventos deber ser algo útil, ya sea para el actor que la inició o para otro actor.

Los casos de uso son historias o casos de utilización del sistema, no son los requerimientos ni las especificaciones funcionales sin embargo sí requieren al menos un conocimiento parcial de

los mismos para ejemplificarlos en las historias que narran. A esta descripción de la utilización del sistema se le conoce también como *escenario de caso de uso*.

En un escenario de caso de uso debe encontrarse a modo de texto a los siguientes elementos:

- ◆ El actor que inicia el caso de uso
- ◆ Condiciones previas para el caso de uso
- ◆ Pasos en el escenario
- ◆ Condiciones posteriores cuando se finaliza el escenario
- ◆ El actor que se beneficia del caso de uso
- ◆ Descripción breve del escenario

Además de los escenarios, existe otra forma de representar los casos de uso, esta forma es el *diagrama de caso de uso* que provee UML. Los diagramas de casos de uso (o *modelos de casos de uso*) representan gráficamente, la forma en que un actor interactúa con el sistema.

El actor, el caso de uso y la línea de interconexión son los elementos básicos que componen un diagrama de caso de uso. La figura B.1 muestra estos elementos.

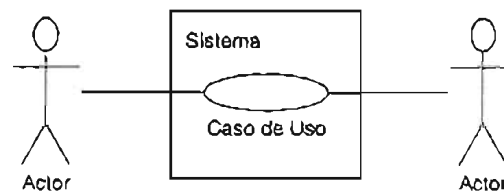


Figura B.1 Elementos Básicos de un Caso de Uso.

Entre los casos de uso se destacan los siguientes tipos de relaciones:

La *inclusión*, la cual permite volver a utilizar los pasos de un caso de uso dentro de otro. Esta relación se utiliza cuando el caso de uso incluido se considera como parte de la funcionalidad del caso de uso que lo incluye.

La relación de *extensión*, permite crear un nuevo caso de uso mediante la adición de pasos a uno ya existente llamado *caso base*. La extensión se realiza en puntos específicos dentro de la secuencia de pasos del caso de uso base; estos puntos se conocen como *puntos de extensión*.

Tanto la inclusión como la extensión se representan con una línea discontinua con punta una punta de flecha. En la figura B.2 se ejemplifican ambas relaciones.

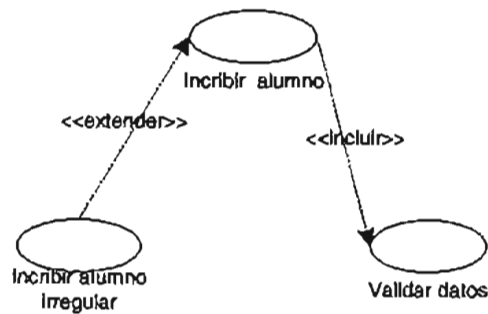


Figura B.2 Inclusión y Extensión de Casos de Uso.

Como en las clases, en los casos de usos también existe la herencia. Cuando un caso de uso se hereda de otro se dice que existe la relación de *generalización*, en la cual el caso de uso secundario hereda las acciones y significado del primario agregando, además, sus propias acciones. La generalización puede establecerse también entre actores (por ejemplo en un organigrama).

Por último, tenemos la relación de *agrupamiento*, mediante la cual es posible agrupar u organizar varios casos de uso. Este tipo de relación es muy útil cuando un sistema consta de varios subsistemas, o cuando los casos de uso de un mismo sistema necesitan ser organizados de acuerdo a categorías que pueden identificarse durante la recopilación de los requerimientos.

B.3 Diagramas de Actividades

Las actividades que ocurren dentro de un caso o dentro del comportamiento de un objeto se dan, normalmente en secuencia y en el UML se representan con un *diagrama de actividades*. Estos diagramas muestran los pasos, puntos de decisión y bifurcaciones que se presentan en un proceso de negocios u operación.

En un diagrama de actividades, a cada actividad se le representa por un rectángulo con las esquinas redondeadas. El procesamiento dentro de una actividad se lleva a cabo y al realizarse, se continúa con la siguiente actividad; la transición de una actividad a otra se representa por una flecha. Cuando una actividad llega a un punto en que se divide en dos actividades que se ejecutan de forma concurrente, se dice que esta actividad se bifurca; una bifurcación se representa por una línea gruesa perpendicular a la transición. El punto inicial del diagrama de actividades se representa con un círculo relleno y el punto final con una diana. Los puntos de decisión se presentan por un rombo. La figura B.3 simboliza la representación básica de un diagrama de actividades.

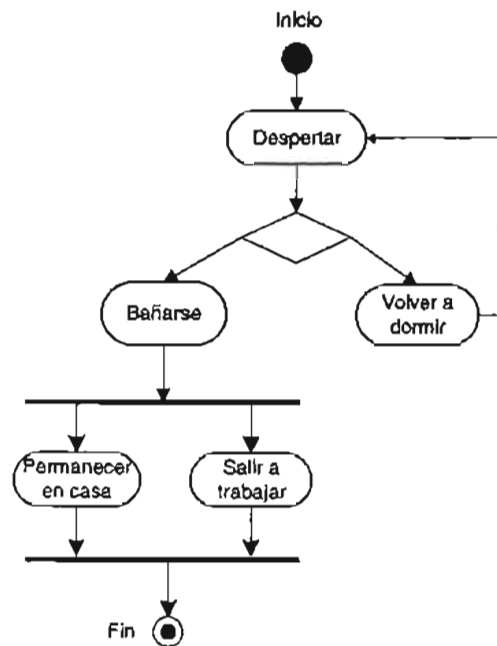


Figura B.3 Diagrama de Actividades.

B.4 Diagramas de Clases

La orientación a objetos sugiere que es más sencillo desarrollar aplicaciones que simulen algún aspecto del mundo cuando el software representa clases de cosas reales. Los diagramas de clases facilitan las representaciones a partir de las cuales los desarrolladores podrán trabajar. A su vez, los diagramas de clases colaboran en lo referente al análisis. Permiten al analista hablarles a los clientes en su propia terminología, lo cual hace posible que los clientes indiquen importantes detalles de los problemas que requieren ser resueltos.

Una clase se representa por un rectángulo que se divide en tres áreas. El área superior contiene el nombre, el área central contiene los atributos, y el área inferior las acciones (métodos). Un diagrama de clases está conformado por un conjunto de rectángulos de este tipo conectados por líneas que muestran la forma en que las se relacionan entre sí.

La siguiente figura muestra un diagrama de clases:

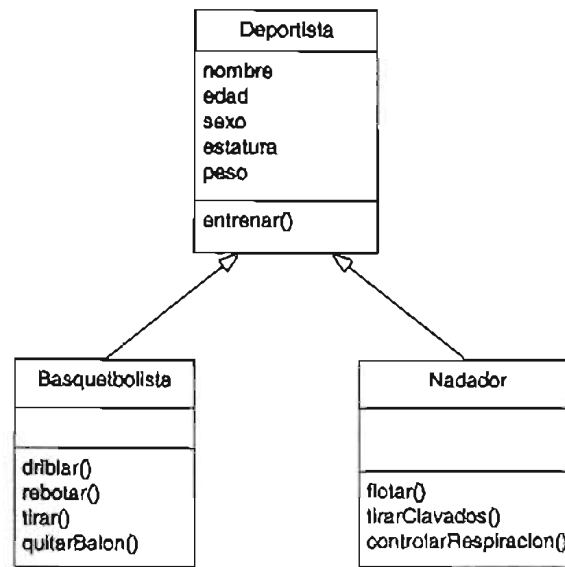


Figura B.4 Diagrama de Clases.

B.5 Diagramas de Secuencias

El diagrama de la secuencia de un sistema es una representación que muestra en determinado escenario de un caso de uso, los eventos generados por actores externos, su orden y los eventos internos del sistema.

El diagrama de secuencia debería prepararse para el curso normal de los eventos de un caso de uso teniendo en cuenta los cursos opcionales más interesantes.

Un diagrama de secuencias muestra las interacciones de los objetos ordenados en una secuencia de tiempo. Muestra los objetos y las clases involucradas en el escenario y la secuencia de mensajes intercambiados entre los objetos necesaria para llevar a cabo la funcionalidad del escenario.

Los diagramas de secuencia por lo regular se asocian con las realizaciones de casos de uso en la vista lógica del sistema en desarrollo.

En UML, un objeto en un diagrama de secuencia es dibujado como un rectángulo conteniendo el nombre del objeto, subrayado. Un objeto puede aparecer de tres formas: el nombre del objeto, el nombre del objeto y su clase, o solamente el nombre de la clase (objeto anónimo).

Los nombres de los objetos pueden ser específicos (por ejemplo objeto2: clase) o pueden ser genéricos (por ejemplo objeto1). Comúnmente un objeto anónimo puede ser usado para representar cualquier objeto en la clase.

Cada objeto tiene también su línea de tiempo representada por una línea punteada debajo del objeto. Los mensajes entre los objetos se representan por flechas que apuntan del cliente (el emisor del mensaje) al proveedor (receptor del mensaje).

Para representar condicionales se utilizan expresiones entre corchetes antecediendo a los mensajes y si se antepone un asterisco la condicional la expresión se evalúa como iteración, esto es, el mensaje se envía del objeto emisor al receptor mientras se cumpla la condición.

La siguiente figura ejemplifica los elementos y notación de los mismos en un diagrama de secuencias.

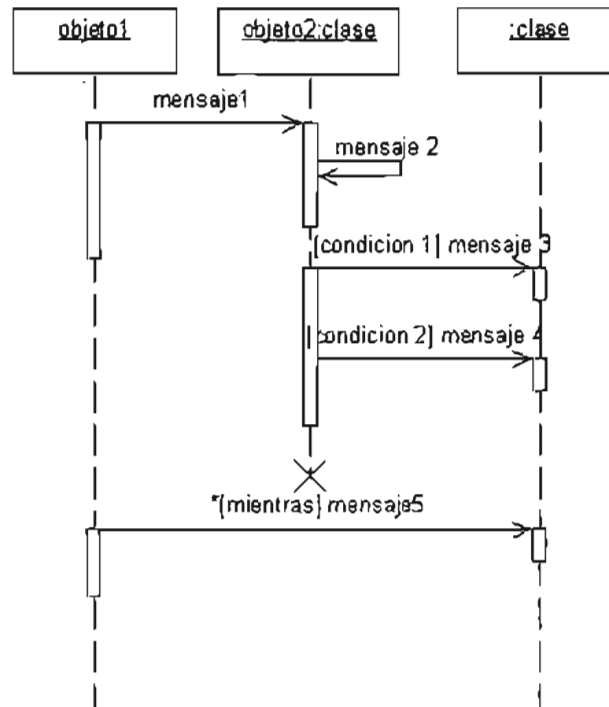


Figura B.5 Diagrama de Secuencias.

Diagramas de Secuencia y Clases Límite. Las clases límite son agregadas a los diagramas de secuencia para mostrar la interacción con el usuario u otro sistema. En las etapas tempranas del análisis, el propósito de mostrar las clases límite en un diagrama de secuencia es capturar y documentar los requerimientos de interfaz, no para mostrar como la interfaz será implementada.

B.6 Diagramas de Componentes

La parte funcional de un sistema tiene como elementos básicos a los componentes de software, los cuales viven en la computadora y no en la mente del analista ni del desarrollador, esto es un elemento físico y no conceptual.

Un *componente* es una parte física de un sistema y puede ejemplificarse por una tabla, un archivo de datos, un archivo ejecutable, una biblioteca de vínculos dinámicos, documentos, etc.

El modelado de componentes y los diagramas de componentes tienen como objetivos principales los siguientes:

- ◆ Que los desarrolladores obtengan habilidades para crear componentes de software reutilizables con el fin de ahorrar tiempo en desarrollos con requerimientos similares.
- ◆ Que las personas encargadas de las notas técnicas y la documentación tengan una base para entender de qué escribirán.
- ◆ Que los desarrolladores cuenten con una estructura del sistema con la cual puedan trabajar en adelante.
- ◆ Que los clientes puedan ver y comprender la estructura del sistema finalizado.

Un punto importante en los diagramas de componentes, es determinar la relación que existe entre un componente y una clase. Un componente, es la "personificación" (implementación) en software de una clase, mientras que ésta es una abstracción de un conjunto de atributos y operaciones. Además, se debe destacar que un componente puede ser la implementación de más de una clase.

Cuando se desee ejecutar las operaciones de un componente, esto será posible sólo a través de una interfaz. Una interfaz es un conjunto de operaciones que especifica algo respecto al comportamiento de una clase. Varias clases pueden incluir las mismas operaciones con las mismas firmas sin necesidad de estar relacionadas con una clase principal por medio de la herencia. Esta reutilización de un conjunto de operaciones de clase en clase, es posible gracias a una interfaz.

Una interfaz puede ser física o conceptual, de tal manera que la interfaz que utiliza una clase es la misma que la que utiliza su implementación en software (componente). En UML, no existe distinción para la representación de una interfaz conceptual y una física.

Un componente puede hacer disponible su interfaz para que otros componentes puedan utilizar las operaciones que contiene. El componente que proporciona tales servicios provee una *interfaz de exportación* mientras que el que accede a dichos servicios utiliza una *interfaz de importación*. La relación que existe entre un componente y una interfaz de importación se conoce como dependencia.

El modelado de interfaces simplifica la vida del desarrollador si al intentar sustituir o reutilizar un componente, puede encontrar la información de su interfaz disponible en un modelo.

Otra característica importante de los componentes es su capacidad de sustitución y reutilización. Un componente puede sustituirse por otro si el nuevo tiene las mismas interfaces que el anterior. Además, puede reutilizarse un componente en otro sistema si éste puede acceder al componente reutilizado por medio de sus interfaces.

Al modelar componentes se pueden distinguir tres tipos:

- ◆ *Componentes de distribución*, los cuales conforman los sistemas ejecutables (DLL, controles ActiveX, ejecutables y Java Beans).
- ◆ *Componentes para trabajar en el producto*, a partir de los cuales se han creado los componentes de distribución (por ejemplo, archivos de base de datos y de código).
- ◆ *Componentes de ejecución*, los cuales son creados como resultado de un sistema en ejecución.

Un *diagrama de componentes* contiene, obviamente, componentes, además de interfaces y relaciones. Un componente se simboliza con un rectángulo que tiene 2 rectángulos pequeños

sobrepuestos en su lado izquierdo. El nombre del componente es una cadena y debe colocarse dentro del símbolo. Además, un componente puede ser miembro de un paquete, entonces para su representación, debe agregarse el nombre del componente como prefijo para el nombre del componente.

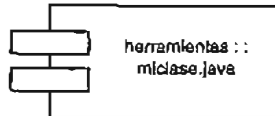


Figura B.6 Representación Básica de un Componente.

Cuando un componente es la implementación de más de una clase, es posible agregar información que muestre más detalle, como por ejemplo las clases que implementa el componente.

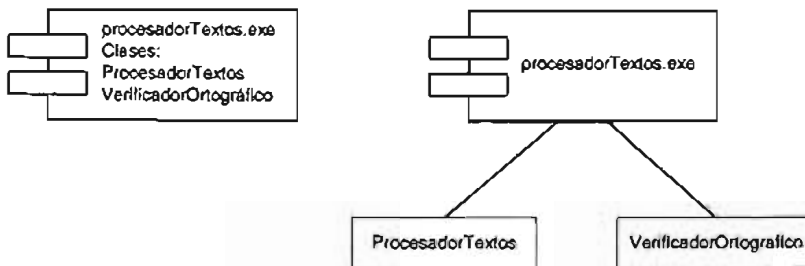


Figura B.7 Componente Implementado por más de una Clase.

Las interfaces son otro de los elementos de los diagramas que ahora estudiamos. En UML existen dos formas de representar una interfaz y la relación con su componente (realización). Ambas formas se muestran en la siguiente figura:

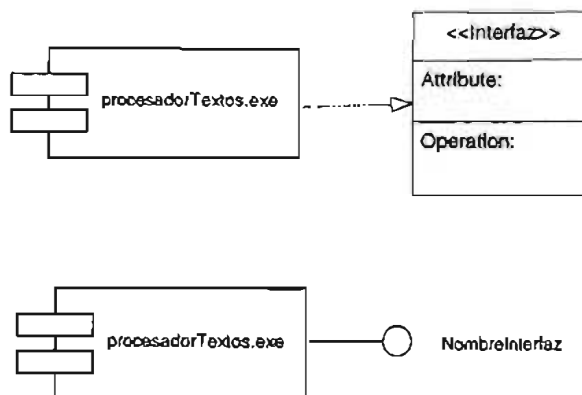


Figura B.8 Representación de Interfaces.

Como se mencionó anteriormente, un componente y una interfaz de importación se relacionan por medio de la dependencia, la cual se representa en la figura siguiente:

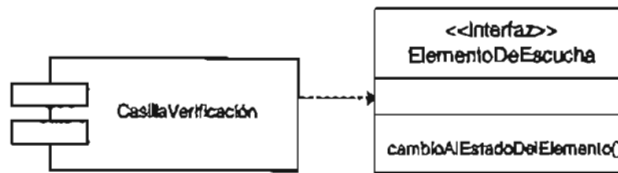


Figura B.9 Representación de la Dependencia.

B.7 Diagramas de Distribución o Despliegue

Los diagramas de despliegue se utilizan para modelar la vista de despliegue estática de un sistema. Esta vista cubre principalmente la distribución, entrega e instalación de las partes que conforman el sistema físico.

El hardware es un tema importante en un sistema de varios componentes. En la actualidad, un sistema puede abarcar diversos tipos de plataformas en ubicaciones dispersas, por lo que un diseño sólido de distribución de hardware es básico para el diseño del sistema [6].

El elemento primordial de hardware es un *nodo*, que es un nombre genérico para todo tipo de recurso de cómputo.

Un diagrama de despliegue es un diagrama que muestra la configuración de los nodos que participan en la ejecución y de los componentes que residen en ellos. Gráficamente, un diagrama de despliegue es una colección de nodos y arcos.

Es posible utilizar dos tipos de nodos:

- ◆ Un procesador: el cual puede ejecutar un componente.
- ◆ Un dispositivo: el cual no ejecuta un componente. Normalmente los dispositivos tienen contacto con el mundo exterior (por ejemplo una impresora o un monitor).

Un nodo se representa básicamente con un cubo, al cual se le deberá asignar un nombre (cadena de texto). Esta representación básica se muestra en la siguiente figura:

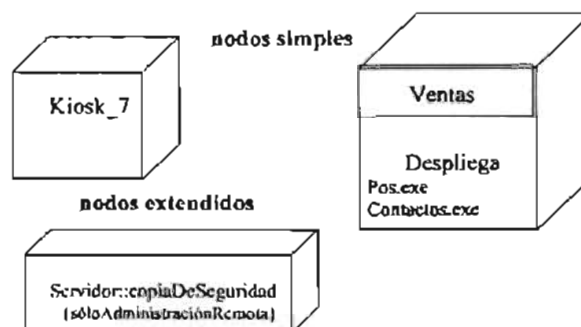


Figura B.10 Representación Básica de los Nodos.

Los diagramas de despliegue también pueden contener componentes, cada uno de los cuales debe residir en algún nodo. Además también es posible representar paquetes o subsistemas, los cuales se utilizan para agrupar elementos del modelo en bloques más grandes.

Para representar la conexión entre dos nodos, los cubos que los representan deberán unirse por una línea que los asocie. Cabe señalar que una conexión no es necesariamente un cable o alambre, sino que también pueden visualizarse conexiones inalámbricas como las infrarrojas o satelitales.

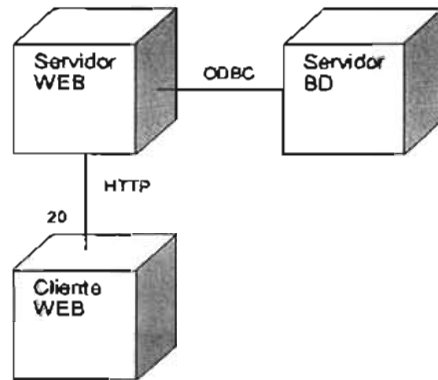


Figura B.11 Ejemplo de Relación entre Nodos.

Si el nodo es parte de un paquete, el nombre puede contener también el nombre del paquete. Adicionalmente, el cubo puede dividirse en compartimentos que agreguen información como puede ser, los componentes colocados en el nodo. Otra forma de indicar los componentes distribuidos es la de mostrarlos en relaciones de dependencia con un nodo.

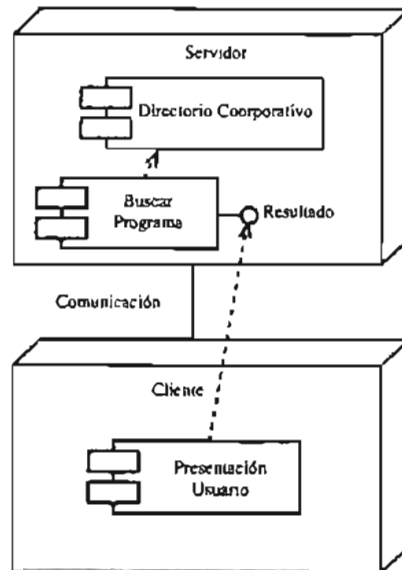


Figura B.12 Representación de Componentes Contenidos en Nodos.

Glosario



GLOSARIO

A

- ActiveX* Conjunto de tecnologías desarrolladas por Microsoft que proporciona objetos basados en el Modelo de Objetos Componente y que busca establecer un estándar especialmente para integrarse en Internet.
- Actor* Entidad externa a un sistema de software que interactúa con él para obtener un resultado importante. Un actor puede ser una persona, otro sistema, una parte del hardware o el paso del tiempo. Para mayor información, consulte el Apéndice B.
- ADO* Abreviatura de ActiveX Data Objects de Microsoft. Es la más reciente interfaz de alto nivel para objetos de datos. Junto con OLE y ODBC, ADO es uno de los principales componentes de la especificación UDA (Universal Data Access) de Microsoft, la cual está designada para proveer un medio consistente de acceso a datos sin tener en cuenta cómo están estructurados los datos.
- API* Abreviatura de Application Program Interface. Conjunto de herramientas, rutinas y protocolos utilizados para la construcción de aplicaciones. Una API facilita el desarrollo de aplicaciones ya que provee todos los bloques de construcción.

B

- Bridge* Dispositivo que conecta dos redes de área local, o bien, dos segmentos de la misma red de área local que usan el mismo protocolo, tales como Ethernet o Token-Ring.

C

<i>Caso de uso</i>	Es el conjunto de acciones que un actor realiza en el sistema para obtener un resultado importante. Para mayor información, consulte el Apéndice B de este trabajo.
<i>Cliente</i>	Cualquier elemento de un sistema de información que requiere un servicio mediante el envío de solicitudes al servidor. Cuando dos programas se comunican por una red, el cliente es el que inicia la comunicación, mientras que el programa que espera ser contactado es el servidor. Cualquier programa puede actuar como servidor para un servicio y como cliente para otro.
<i>COM</i>	Abreviatura de Component Object Model. Arquitectura de software desarrollada por Microsoft para construir aplicaciones basadas en componentes. Estos componentes son independientes del lenguaje, tienen capacidad de comunicación entre procesos y son fácilmente colocados en un diseño de un programa orientado a objetos.
<i>Constraint</i>	Es una restricción en los valores contenidos en un objeto particular de base de datos como puede ser una tabla o una columna. Un constraint define las reglas de integridad del objeto de base de datos.
<i>Cookie</i>	Mensaje que un servidor de Web envía a un navegador. El navegador almacena el mensaje en un archivo de texto. Este mensaje es enviado de regreso al servidor cada vez que el navegador solicita una página al servidor Web.
<i>CORBA</i>	Siglas de Common Object Request Broker Architecture. Especificación de mensajes basada en objetos desarrollada por el OMG. Arquitectura que habilita piezas de programas, llamados objetos, para comunicarse con algún otro sin tener en cuenta el lenguaje de programación en el que están escritos o en qué sistema operativo están corriendo.
<i>Cron</i>	Comando del sistema operativo UNIX que sirve para calendarizar trabajos a ser ejecutados alguna vez en el futuro, una vez o periódicamente.
<i>CSS</i>	Abreviatura para Cascading Style Sheets. Lenguaje que permite definir formatos de estilos de presentación para documentos escritos en HTML o XML.

D

- DataWarehouse** Una tecnología y una disciplina orientada a la construcción de una colección de datos que permita y facilite el acceso a la información según lo requieran los procesos de toma de decisiones. Data WareHouse es el soporte a una diferencia competitiva en el corto/medio plazo que, a largo plazo, puede asegurar la supervivencia.
- DOM** Abreviatura de Domain Object Model: Es un parser de XML que permite representar el documento estructurado en un modelo orientado a objetos, el cual permite hacer operaciones de actualización, inserción y borrado de estructuras o elementos del documento.
- DSN** Abreviatura de Data Source Name. Un DSN provee conectividad a una base de datos a través de un driver ODBC. El DSN contiene el nombre de la base de datos, directorio, driver, usuario, contraseña, etc. Una vez que se ha creado del DSN para una base de datos particular, éste se puede utilizar en una aplicación para llamar información desde la base de datos.

F

- Frame** En gráficos: Área rectangular en la cual el texto y los gráficos son desplegados.
- En HTML: Se refiere a la división del área de despliegue de un browser en secciones separadas, cada una de las cuales es realmente una página Web diferente.
- Freeware** Software registrado como propiedad intelectual dado libremente por el autor. Aunque es dado libremente, el autor retiene el registro intelectual, lo cual significa que no es posible hacer cualquier cosa con él que no esté expresamente permitido por el autor. Usualmente, el autor permite el uso del software pero no venderlo.
- Front-End** O capa de presentación. Es el conjunto de objetos que constituyen la interfaz gráfica de usuario en una aplicación, como por ejemplo las pantallas, menús, etc.

G

- GET** Método utilizado para enviar los datos contenidos en un formulario al servidor Web. Con este método, los datos del formulario son agregados al URL y se envían como parte de éste.

H

- Hilo* En programación, una parte de un programa que puede ejecutarse independientemente de otras.
- HTML* Es el lenguaje utilizado para la creación de páginas Web. Sus siglas vienen del inglés HyperText Markup Language.
- HTTP* Protocolo de comunicación utilizado para la transferencia de hipertexto. Sus siglas vienen del inglés HyperText Transfer Protocol.

I

- IIS* Abreviatura de Internet Information Server. Es el servidor Web de Microsoft que corre en plataformas Windows NT. De hecho, IIS viene incluido con Windows NT y actualmente está disponible sólo para esta plataforma, mientras que los servidores de Web de Netscape corren en la mayoría de las plataformas, incluyendo Windows NT y UNIX.
- Internet* Red mundial de computadoras que transporta datos y hace posible el intercambio de información. Internet es en último término un conjunto de servidores que proporcionan servicios de transferencia de ficheros, correo electrónico o páginas WEB, entre otros.

J

- Java Bean* Clase Java que tiene las características: Clase pública, constructor sin argumentos, propiedades y contiene métodos "set" y "get". Por cada propiedad deben existir dos métodos, uno "set" para dar valores a la propiedad y uno "get" para obtenerlos.
- JDBC* Del inglés Java Database Connectivity. API de Java que brinda a los programas la posibilidad de interactuar con una base de datos y ejecutar instrucciones sql.
- JDK* Siglas de Java Development Kit. Conjunto de herramientas de desarrollo que provee la tecnología Java.

K

KDE (K Desktop Environment o Common Desktop Environment): es un ambiente de escritorio y de desarrollo gratuito, que corre bajo plataformas UNIX, BSD, Linux, Solaris, en MAC para ambientes gráficos de X-Windows y en Windows utilizando Cygwin, que es una colección de Software gratuito que permite que el ambiente de Windows se maneje como UNIX.

M

Metodología Justificación, discusión y análisis de los diversos procedimientos concretos que se emplean en las investigaciones y la discusión acerca de sus características, cualidades y debilidades.

Middleware Término utilizado para describir productos separados que sirven como medio de unión entre dos aplicaciones o bien entre dos capas de una misma aplicación y pasan datos entre ellas.

Modelo Es la abstracción de algo real con el fin de analizarlo y comprenderlo fácilmente. Un modelo puede representarse gráfica o matemáticamente.

N

Navegador O Web browser. Software utilizado para localizar y desplegar páginas Web.

O

Open source Término en inglés utilizado para referirse a un programa en el cual el código fuente está disponible para el público general, para el uso y/o modificación libre de cargo. Este tipo de software es creado por la colaboración de comunidades de programadores que mejoran el código y comparten los cambios dentro de la comunidad.

ODBC Abreviatura del inglés Open DataBase Connectivity. Método estándar para el acceso a bases de datos desarrollado por Microsoft.

OLTP Abreviatura del inglés On-Line Transaction Processing. Tipo de procesamiento, en el cual la computadora responde inmediatamente a las peticiones de los usuarios. Cada petición es considerada como una transacción.

P

POST Método utilizado para enviar los datos contenidos en un formulario al servidor Web. Con este método, los datos del formulario son incluidos como el cuerpo del mensaje.

Proceso En computación: Un programa en ejecución. Término utilizado como sinónimo de tarea.

En lenguaje: Un proceso puede ser definido como un conjunto de actividades enlazadas entre sí que, partiendo de uno o más entradas los transforma, generando un resultado. Conjunto de pasos u operaciones ordenadas, cuyo fin es la obtención de un resultado determinado.

PWS Del inglés Personal Web Server. Término genérico para un servidor Web que es utilizado en una computadora personal. PWS es una versión menor del IIS de Microsoft y por lo tanto está limitada en sus capacidades. Está diseñado para soportar sitios Web con tráfico limitado o para ser usado como un servidor de inicio para la construcción de páginas que serán transferidas a un servidor que puede manejar grandes cantidades de tráfico.

S

Shareware Software que es entregado libre de cargo pero el autor pide un pequeño pago si el programa es del gusto del usuario y lo utiliza regularmente. Por el envío del pago, el usuario es registrado y puede recibir servicio de asistencia y actualizaciones.

Scripting Lenguaje de programación de alto nivel que es interpretado por otro programa en tiempo de corrida en vez de ser compilado por el procesador de la computadora como otros lenguajes de programación. Este tipo de lenguaje es utilizado para agregar funcionalidad a una página Web.

Servidor Genéricamente, dispositivo de un sistema que resuelve las peticiones de otros elementos del sistema, denominados clientes. Suele utilizarse para mantener datos centralizados o para gestionar recursos compartidos.

Software Conjunto de instrucciones de computadora que al ser ejecutadas proporcionan el resultado y el comportamiento deseado.

Stored Procedure Programa conformado de una secuencia de sentencias SQL que se encuentra almacenado en la base de datos y es usado para desempeñar una tarea particular.

T

Transacción En informática, se llama transacción a la operación que modifica el estado de una base de datos, sin que los datos en sí mismos pierdan consistencia alguna.

Trigger Es una forma especial de stored procedure ejecutado automáticamente cuando un usuario ejecuta una instrucción que modifica los datos.

V

Vista Es una consulta que es almacenada en la base de datos como un objeto. Permite a los usuarios ver un subconjunto de registros o columnas desde una o más tablas. Cada vez que un usuario usa una vista de una tabla particular o combinación de tablas, esta vista es ejecutada en el momento con la información contenida en esas tablas.

W

Web Sistema conformado por un conjunto de servidores que soportan colecciones de documentos especialmente formateados con el lenguaje de marcación HTML que soportan ligas a otros documentos tan bien como archivos con gráficos, audio y/o video.

X

XHTML XHTML (EXtensible Hypertext Markup Language), Es un lenguaje basado en etiquetas con la misma finalidad que HTML pero que posee mayores restricciones en cuanto a la sintaxis durante la construcción de páginas Web.

XML Abreviatura de Extensible Markup Language. Lenguaje diseñado especialmente para documentos Web, el cual permite a los diseñadores crear sus propias etiquetas habilitando la definición, transmisión, validación e interpretación de datos entre aplicaciones y entre organizaciones.

REFERENCIAS

- [1] Basque, Richard. *El Modelo de Madurez de Capacidades del Software del SEI (CMM)*, Fédération de l'Informatique du Québec (FIQ), Québec Canadá, 1999.
- [2] Jacobson Ivar, et al. *Object Oriented Software Engineering: A Use Case Driven Approach*, Addison-Wesley, 1992.
- [3] Booch Grady, Jacobson Ivar, Rumbaugh James. *El Proceso Unificado de Desarrollo de Software*, Pearson Educación SA, Madrid España, 2000.
- [4] Sun Microsystems. *Web Component Development with Servlet and JSP TM Technologies SL-314 Student Guide*, USA, 2002.
- [5] Larman Craig. *UML y Patrones, Introducción al Análisis Orientado a Objetos*, Prentice Hall, México, 1999.
- [6] Booch Grady, Jacobson Ivar, Rumbaugh James, *El lenguaje Unificado de Modelado*, Addison Wesley Iberoamericana, Madrid 1999.
- ◆ García, Miguel Ángel. Introducción a JSP [en línea]. Comunidad javaHispano, versión 1, 1, Febrero 2003, [consultada Julio 2003]. Disponible en World Wide Web: <<http://www.javahispano.org/tutorials.type.action?type=j2ee>>
 - ◆ The Leading IT Encyclopedia and Learning Center [en línea], [consultada en Agosto 2003]. Disponible en World Wide Web: <<http://whatis.techtarget.com/>>
 - ◆ Java Technology: Products & Services [en línea]. Sun Microsystems Home Page, 1994, revisada en 2004 [consultada en 2003 y 2004]. Disponible en World Wide Web: <<http://www.sun.com>>
 - ◆ Diccionario Informático [en línea]. La Web del Programador: Comunidad de Programadores, 2000, [consultada Septiembre 2004]. Disponible en World Wide Web: <<http://www.lawebdelprogramador.com>>
 - ◆ Real Academia Española [en línea], [consultada en Septiembre 2004]. Disponible en World Wide Web: <<http://www.rae.es>>
 - ◆ Soca, Ricardo. La Página del Idioma Español [en línea]. Asociación Cultural Antonio de Nebrija, 1996, actualizado 2004, [consultada Septiembre 2004]. Disponible en World Wide Web: <<http://www.elcastellano.org/>>

