



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE ESTUDIOS SUPERIORES
“ARAGÓN”

“ANÁLISIS E IMPLEMENTACIÓN DE CIDE INTRANET
A TRAVÉS DE LA METODOLOGÍA RUP PARA USO
DEL PERSONAL ACADÉMICO Y ADMINISTRATIVO
DEL CENTRO DE INVESTIGACIÓN Y DOCENCIAS
ECONÓMICAS”.

T R A B A J O E S C R I T O
EN LA MODALIDAD DE DESARROLLO
DE UN CASO PRÁCTICO
QUE PARA OBTENER EL TÍTULO DE :
INGENIERO EN COMPUTACIÓN
P R E S E N T A :
OMAR MUÑOZ MORALES



ASESOR
M. en TI. OMAR MENDOZA GONZÁLEZ

MÉXICO 2009



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:

Mamá: Detrás de esté y todos mis logros estás tú, con tu infinito cariño y comprensión, sin ti nada de esto sería posible. Gracias por siempre confiar en mí y apoyarme para continuar mis estudios.

Papá: Gracias por tú apoyo, confianza y cariño, siempre estaré en deuda.

Aldo: Gracias por tus comentarios, y sugerencias, por supuesto que siempre contarás con todo mi apoyo al igual que yo he contado con el tuyo.

Michel: No pretendo ser un ejemplo, pero espero y deseo que al igual que yo culmines tus estudios y siempre sigas siendo la persona noble y cariñosa que eres.

Abuelita (Asunción): Por ser el pilar de nuestra familia, gracias por compartir esté y todos mis logros.

Abuelito (Emiliano): Gracias por dejar un legado importante en mi vida.

Abuelita (Isaura): Gracias por tu cariño y enseñanzas.

Abuelito (Benjamín): Gracias por compartir mis alegrías y logros.

Primos: Gracias por compartir tantos recuerdos memorables, a todos y cada uno les deseo éxito en todo proyecto.

A todos los tíos/tías de la familia Muñoz Morales, por siempre mostrar interés y apoyo, de cada uno he aprendido grandes lecciones.

A mis compañeros y amigos de la Preparatoria 7, FES Aragón y CIDE que estuvieron conmigo compartiendo una estancia placentera dentro y fuera de las aulas a todos ellos gracias.

Gracias a todos mis profesores que participaron en mi desarrollo académico, sin su dedicación y esfuerzo la Universidad Nacional Autónoma de México no sería lo mismo.

1.	INTRODUCCIÓN.....	1
1.1	DESARROLLO DE PROCESOS A TRAVÉS DE RATIONAL UNIFIED PROCESS.....	3
1.2	FLUJO DE TRABAJO.....	9
1.3	ITERACIONES.....	10
1.4	FASES.....	12
1.4.1	<i>Inicio</i>	13
1.4.2	<i>Elaboración</i>	13
1.4.3	<i>Construcción</i>	13
1.4.4	<i>Transición</i>	13
2.	PLAN DE DESARROLLO DE SOFTWARE.....	16
2.1	INTRODUCCIÓN PLAN DE DESARROLLO DE SOFTWARE.....	16
2.2	PROPÓSITO.....	16
2.3	ALCANCE.....	16
3.	VISTA GENERAL DEL PROYECTO.....	17
3.1	PROPÓSITOS, ALCANCE Y OBJETIVOS.....	17
3.2	SUPOSICIONES Y RES FRICCIONES.....	19
3.3	ENTREGARLES DEL PROYECTO.....	20
3.4	EVOLUCIÓN DEL PLAN DE DESARROLLO DE SOFTWARE.....	20
4.	ORGANIZACIÓN DEL PROYECTO.....	21
4.1	ROLES Y RESPONSABILIDADES.....	21
5.	GESTIÓN DEL PROCESO.....	21
5.1	ESTIMACIONES DEL PROYECTO.....	21
5.2	PLAN DE PROYECTO.....	22
5.2.1	<i>Plan de las fases</i>	22
5.2.2	<i>Objetivos de la iteración</i>	22
5.3	SEGUIMIENTO Y CONTROL DEL PROYECTO.....	23
5.3.1	<i>Gestión de Requisitos</i>	23
5.3.2	<i>Control de Calidad</i>	24
6.	INTRODUCCIÓN ANÁLISIS DE REQUERIMIENTOS.....	25
6.1	PROPÓSITO GENERAL DEL SISTEMA.....	25
6.2	ALCANCE DEL SISTEMA.....	26
6.3	OBJETIVOS.....	26
6.4	DEFINICIONES, ACRÓNIMOS Y ABREVIATURAS.....	27
7.	SISTEMA ACTUAL.....	29
8.	SISTEMA PROPUESTO.....	33
8.1	VISIÓN GENERAL.....	33
8.1.1	<i>Especificaciones</i>	34
8.1.2	<i>Descripción general</i>	35
8.1.3	<i>Diseño</i>	36
8.2	REQUISITOS FUNCIONALES.....	37
8.3	REQUISITOS NO FUNCIONALES.....	38
8.3.1	<i>Facilidad de uso ("usability")</i>	39
8.3.2	<i>Confiabilidad</i>	39
8.3.3	<i>Performance</i>	39

8.3.4	<i>Restricciones de diseño</i>	40
8.3.5	<i>Seguridad</i>	41
8.3.6	<i>Documentación de usuario y sistemas de ayuda</i>	42
8.3.7	<i>Interfaces</i>	42
8.4	MODELOS DEL SISTEMA	46
8.5	ESCENARIOS	47
8.6	MODELO DE CASOS DE USO.....	48
8.6.1	<i>Diagramas del Modelo de Casos de Uso. (Diseño del Modelo de Casos de Uso)</i>	48
9.	INTRODUCCIÓN A LA ARQUITECTURA DE SOFTWARE	63
9.1	ALCANCE	63
10.	RESUMEN ARQUITECTÓNICO	63
10.1	HECHOS MÁS IMPORTANTES.....	63
10.2	ESTILO ARQUITECTÓNICO	65
10.3	OBJETIVOS DE LA ARQUITECTURA	66
11.	COMPONENTES SIGNIFICATIVOS DE LA ARQUITECTURA DEL SISTEMA	66
11.1	LENGUAJE Y FRAMEWORK DE DESARROLLO.....	68
11.2	SERVIDOR WEB.....	74
11.3	SERVIDOR DE BASE DE DATOS.....	74
12.	VISTA DE CASOS DE USO	74
13.	VISTA LÓGICA	77
13.1	REPARTICIÓN DEL PROCESAMIENTO.....	78
14.	ASEGURAMIENTO DE LA CALIDAD	80
15.	OBJETIVO DE LA IMPLEMENTACIÓN	82
15.1	CALENDARIO DE TRABAJO ITERATIVO.....	83
16.	GUÍA DE INSTALACIÓN	85
16.1	COMPILACIÓN Y EJECUCIÓN DE LA APLICACIÓN.....	86
16.2	ESPECIFICACIÓN DE LA APLICACIÓN.....	92
16.3	ADMINISTRACIÓN DEL SISTEMA.....	126
16.4	ARCHIVOS DE CONFIGURACIÓN.....	126
17.	ESQUEMA DE LA BASE DE DATOS	128
18.	CONCLUSIONES	139
19.	GLOSARIO	142
20.	BIBLIOGRAFÍA	147
21.	FUENTES DE INFORMACIÓN DIGITAL (INTERNET)	148

1. Introducción

El presente trabajo muestra el desarrollo del sistema CIDE INTRANET para uso del personal académico y administrativo del Centro de Investigación y Docencia Económicas, el cual es un centro de investigación y educación superior perteneciente a la red de Centros Públicos de Investigación del Consejo Nacional de Ciencia y Tecnología (CONACYT).

En una etapa inicial se contemplaba la migración total del sistema SIGCIDE, debido a motivos como: incompatibilidad de plataforma, reinstalaciones frecuentes en los equipos de usuarios e interfaz compleja y poco *amigable*, pero al analizar el código fuente, procedimientos almacenados y funcionalidad del sistema, se llegó a la conclusión de no desarrollar los módulos correspondientes al área contable, becas, caja y almacén, debido a que resultaba incosteable la asignación de tiempo y personal para el desarrollo de los módulos, esta decisión se tomó con base en el análisis de perfiles de usuarios.

Por lo tanto se tiene como objetivo desarrollar un sistema multiplataforma, capaz de atender el mayor número de usuarios del Centro de Investigación y Docencia Económicas, evitando las reinstalaciones y la dependencia de una plataforma específica para realizar sus actividades.

El desarrollo de sistemas no sólo implica el conocimiento de una tecnología o plataforma específica, también se requiere el manejo de todas las relaciones y elementos existentes entre el desarrollador de software y la organización para la cual se desarrolla un sistema, por tal motivo, resulta una obligación del Arquitecto de Software diseñar un sistema que esté en función del contexto y necesidades del proyecto.

Describir los procesos implicados en el desarrollo de software resulta una labor ardua, por lo general debe ser una representación abstracta y sin vínculos aparentes con una determinada tecnología.

Comúnmente los procesos se definen de dos maneras totalmente opuestas; la primera suele definir los procesos de forma muy genérica, plagados de métodos, procedimientos y una logística de acuerdo al tamaño del proyecto; en el segundo tipo de descripción el proceso se define de manera precisa y específica, convirtiéndose en un proceso enorme, el cual es difícil de manejar debido a la cantidad de información especificada.

Es conveniente definir el proceso de manera abstracta, teniendo como premisa la idea de que un proceso sólo tiene valor en la medida en que colabora con el desarrollo de software, es necesario recordar que la posibilidad de que una misma solución funcione de manera íntegra en otro desarrollo suele ser muy baja, por eso es necesario desarrollar soluciones acordes a la definición adecuada del proceso.

Además de la correcta definición de procesos, es indispensable contar con una fusión óptima de tecnologías y procesos que permitan el adecuado desarrollo e implementación del sistema en función de las necesidades del proyecto, por tal motivo se ha hecho la siguiente elección:

Tecnologías: Java, JEE, UML

Procesos: Rational Unified Process (RUP)

El desarrollo de sistemas está conformado por una serie de atributos comunes a la mayoría de proyectos, entre los cuales se encuentran: rendimiento, escalabilidad, confiabilidad, mantenibilidad, seguridad, facilidad de incorporar nuevas funcionalidades, así como la posibilidad de integración en otros sistemas, todos y cada uno de los aspectos mencionados deben ser considerados al momento de realizar el análisis de tecnologías y procesos implicados en la construcción de un sistema.

Los sistemas empresariales se constituyen por el conjunto de reglas mencionadas anteriormente y la implementación de procesos de negocio, los cuales se pueden definir como un cúmulo de reglas propias de cada negocio y, por tanto contienen una amplia gama de conceptos propios del proceso. Por ejemplo, un sistema de procesamiento de pedidos debe realizar un seguimiento de productos, clientes, unidades, etcétera. Cada una de sus fases representa la unión de reglas de negocio propias y comunes de otros sistemas, es por ello que, a pesar de contar con una definición realizada de manera general, la construcción se lleva a cabo de forma particular, siguiendo las reglas propias del negocio.

Además de recabar y procesar información, los sistemas deben de ser capaces de gestionar las reglas de negocio, estructuras y asociaciones que vayan de acuerdo con el tipo de requerimiento establecido por el procedimiento realizado por el usuario.

Otro aspecto fundamental en el desarrollo de software es la proyección a futuro del sistema, y este hecho se refiere a la capacidad que deberá tener el sistema para mantenerse a pesar de uno o más cambios en una o más condiciones para las cuales se desarrolló

originalmente. En caso de que el ritmo de crecimiento rebase las expectativas, el sistema deberá contar con la capacidad de ser extensible, dicha extensión permitirá adecuarse a las necesidades actuales sin dejar de lado el desarrollo realizado previamente. La extensión del sistema permitirá un ajuste y adaptación de procedimientos, a lo cual se le denomina como mantenimiento o actualización del sistema, el proceso deberá implicar el menor riesgo y, la menor complejidad posible.

1.1 Desarrollo de procesos a través de Rational Unified Process

El Proceso Unificado Relacional (RUP, *Rational Unified Process*, por sus siglas en inglés), es una metodología que tiene por objetivo el desarrollo de software a través de la asignación y especificación de tareas dentro de un grupo de desarrollo; su premisa es la producción de software de calidad que satisfaga las necesidades de los usuarios finales en un marco de tiempo y presupuesto previsible ("mejor software, más rápido").

RUP fue diseñado específicamente con la finalidad de apearse a las siguientes prácticas consideradas como buenas e indispensables en el desarrollo de software.

Desarrollo iterativo: Establece la funcionalidad del sistema a través de una serie de tareas, mismas que aumentan de manera gradual a lo largo del desarrollo. Cada una de las fases o etapas de la iteración es precedida por otra etapa de menor complejidad.

Gestión de requerimientos: Enfoque sistemático y ordenado para obtener, documentar, controlar y gestionar la información referente a los requisitos de especificación del sistema, incluida la evaluación de repercusiones causadas por cualquier cambio en el sistema. Una gestión eficaz implica el mantenimiento de una definición clara de requisitos, así como el seguimiento y actualización de cada uno de ellos.

Uso de arquitectura basada en componentes: Desarrollo de software mediante una arquitectura basada en componentes, con lo cual se tiende a reducir la complejidad de la solución, permitiendo la reutilización de componentes.

Modelado visual: Conjunto de diagramas que especifican y detallan las funcionalidades del sistema. El objetivo del modelo es proporcionar una mejor comprensión del sistema que se está desarrollando.

Verificación continua de la calidad: La calidad del sistema deberá ser evaluada continuamente de acuerdo con los requisitos funcionales, la verificación de calidad en cada

etapa emitirá una oportuna corrección en una fase temprana del ciclo de desarrollo.

Control de cambios: Establece políticas de control y gestión de cambios (cambios de requisitos, tecnologías, recursos, plataformas, etcétera), estos cambios de control son establecidos como parte de la especificación del sistema y proporcionan un medio eficiente de sincronización entre los cambios producidos en cada iteración.

Estas prácticas son producto de la experiencia conjunta de un grupo de desarrolladores y, han sido nombradas como "*best practices*" (mejores prácticas); tienen como finalidad el desarrollo de software de alta calidad.

La concepción y desarrollo de software requiere el uso y seguimiento de un proceso eficaz, capaz de realizar una adecuada descripción de funciones, procedimientos y tiempos en los cuales se desarrollarán cada una de las actividades planificadas. RUP es una metodología de desarrollo completamente apegada a las siguientes especificaciones:

Roles: The who (*Quiénes*)

Artefactos: The what (*Cómo*)

Actividades: The how (*Cuáles*)

Fases, iteraciones, metodologías, flujos de trabajo: The when (*Cuando*)

Roles: Se considera como *rol* al conjunto de actividades y responsabilidades definidas para una persona o grupo de individuos que trabajan en equipo dentro de una organización de desarrollo de software. En el desarrollo de software una persona puede desempeñar múltiples roles, o de manera inversa un conjunto de personas pueden realizar un sólo rol.

La figura 1.1 muestra la representación gráfica de roles acorde a la metodología RUP



Figura 1.1

Representación de roles acordes a la metodología RUP

Artefactos: Un artefacto es una pieza de información producida y/o usada durante la etapa de ejecución del proceso. El producto final se encuentra conformado por un subconjunto de artefactos producidos durante el desarrollo del proyecto.

Un artefacto puede definirse de distintas maneras, entre las más comunes se encuentran:

- ✓ Modelo de casos de uso
- ✓ Diseño de clases
- ✓ Documentación, definición Arquitectura de Software
- ✓ Código fuente
- ✓ Ejecutables
- ✓ Plan de Desarrollo de Software

Durante el desarrollo de software los artefactos son producidos o modificados de acuerdo a la asignación de roles especificada previamente.

En la figura 1.2 se muestra la representación gráfica de artefactos en RUP.



Figura 1.2

Representación gráfica de artefactos en RUP

Actividades: Una actividad es un conjunto de pasos capaces de satisfacer un objetivo, se caracteriza por tener un propósito definido que pueda producir o actualizar uno o más artefactos. Cada actividad es asignada a una función específica, algunas funciones pueden repetirse en algunas ocasiones.

En la figura 1.3 se muestra la representación de una actividad en RUP.



Figura 1.3

Representación de actividades en RUP

Disciplinas: Dentro de la metodología RUP se conoce como disciplinas al conjunto de actividades que tienen por objetivo desglosar uno o más procesos de relativa importancia dentro del sistema.

La figura 1.4 muestra el conjunto de disciplinas establecidas por RUP.

Disciplina RUP	Descripción
Modelado de negocio	<p>Finalidades:</p> <ul style="list-style-type: none"> ▪ Comprender la estructura organizacional para su adecuada implementación acorde al entorno de la institución o empresa. ▪ Comprender e identificar los problemas comunes dentro de la estructura organizacional con la finalidad de realizar mejoras en los procesos. ▪ Establecer los requerimientos necesarios que cumplan con el objetivo de la organización. ▪ Establecer un punto de acuerdo común entre clientes, usuarios finales y desarrolladores.
Requerimientos	<p>Propósitos:</p> <ul style="list-style-type: none"> ▪ Establecer y mantener acuerdos entre los usuarios y toda aquella persona implicada en los procesos organizacionales.
	<ul style="list-style-type: none"> ▪ Proporcionar una visión clara y concisa del proceso que será desarrollado. ▪ Delimitar las funciones del sistema.

	<ul style="list-style-type: none"> ▪ Establece el contenido y planificación de las iteraciones implicadas en el desarrollo del sistema.
Análisis y diseño	<p>El análisis y diseño tienen las siguientes finalidades:</p> <ul style="list-style-type: none"> ▪ Convierte los requisitos funcionales en lo que será el diseño del sistema. ▪ Desarrollo de una arquitectura robusta para el sistema. ▪ Adapta el diseño al entorno de la aplicación.
Implementación	<p>Objetivos:</p> <ul style="list-style-type: none"> ▪ Define la organización seguida durante la etapa de implementación. ▪ Implementa el diseño de los elementos previamente especificados. ▪ Realización de pruebas. ▪ Integración del código producido de manera individual, generando una versión funcional.
Pruebas	<p>El propósito de esta disciplina es:</p> <ul style="list-style-type: none"> ▪ Detectar y documentar los defectos encontrados en el sistema.
	<ul style="list-style-type: none"> ▪ Documentación acerca de la percepción general de la calidad de software. ▪ Validación de la especificación de requisitos implementados en el sistema. ▪ Validación del funcionamiento del sistema acorde a las funciones especificadas en el diseño. ▪ Validación de funciones implementadas (especificación de requerimientos implementada apropiadamente)

Puesta en marcha (<i>Deployment</i>)	<p><i>El propósito de esta disciplina es:</i></p> <ul style="list-style-type: none"> ▪ Asegurar que el <i>software</i> se encuentra disponible para todos los usuarios finales.
Configuración y gestión de cambios	<p><i>Su propósito es:</i></p> <ul style="list-style-type: none"> ▪ Identificar la configuración de <i>items</i>¹ ▪ Restricción de los futuros cambios que serán realizados. ▪ Auditoría de los cambios realizados.
Gestión del proyecto	<p><i>Su propósito es:</i></p> <ul style="list-style-type: none"> ▪ Gestión del desarrollo de actividades relacionadas al proyecto. ▪ Planificación de actividades dentro del equipo de desarrollo.
	<ul style="list-style-type: none"> ▪ Monitoreo del estatus del proyecto. ▪ Gestión de riesgos.
Entorno de desarrollo	<ul style="list-style-type: none"> ▪ El entorno de desarrollo tiene como objetivo proporcionar <i>software</i> acorde al entorno, requerimientos, procesos y actividades particulares a la organización para la cual es desarrollado.

Figura 1.4

Tabla de disciplinas establecidas por la metodología RUP

¹Un *item* es un elemento de configuración gestionado por un sistema de control de versiones, el cual es identificado de manera única, por lo tanto establece un punto único en la etapa de desarrollo del sistema.

1.2 Flujo de trabajo

Un flujo de trabajo es una secuencia de actividades producidas de manera conjunta por dos o más procesos independientes que siguen un orden definido. La representación gráfica de los diagramas de flujo de trabajo es realizada mediante el Lenguaje Unificado de Modelado (UML, por sus siglas en inglés, Unified Modeling Language).

En la figura 1.5 se muestra el flujo de trabajo correspondiente al modelo estructural e implementación de requerimientos.

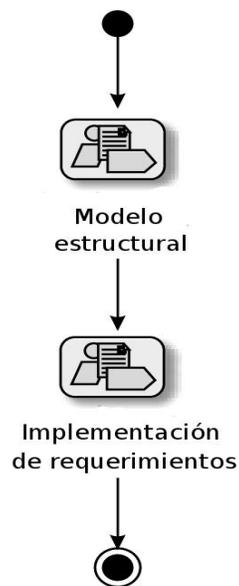


Figura 1.5

Flujo de trabajo modelo estructural e implementación de requerimientos

En la figura 1.6 se muestra un diagrama detallado del flujo de trabajo correspondiente a las actividades de implementación de requerimientos, realización de pruebas y revisión.

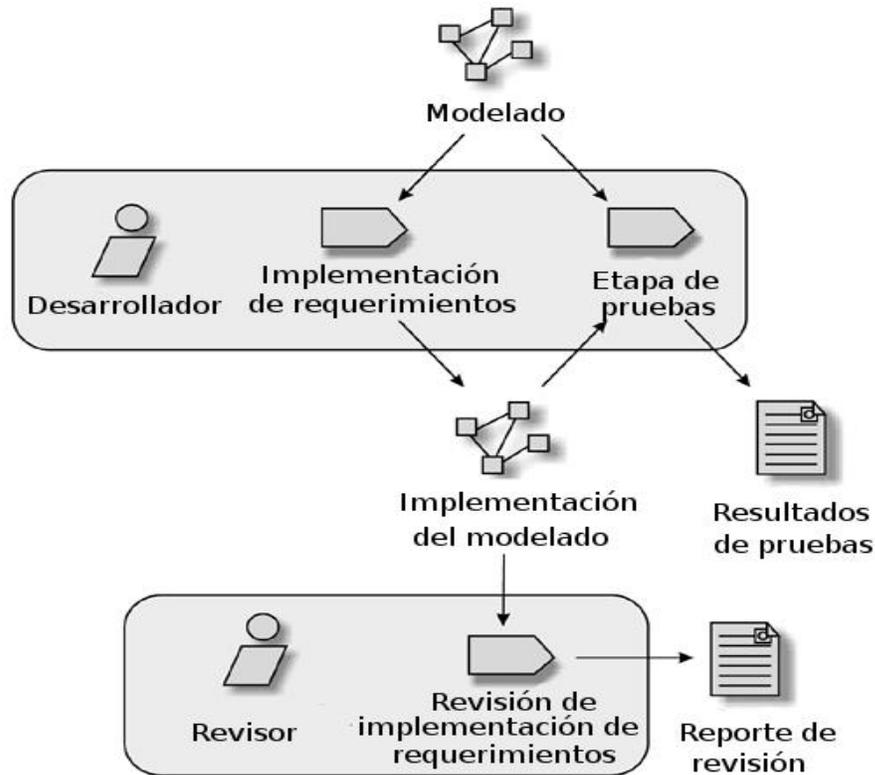


Figura 1.6

Implementación de flujo de trabajo detallado

1.3 Iteraciones

El desarrollo iterativo hoy en día es considerado como la clave de éxito de todo desarrollo de software. El ciclo de vida de desarrollo de software es realizado a través de las nueve disciplinas mencionadas anteriormente, cada una de ellas constituyen una iteración del sistema, permitiendo una mejor comprensión de los requisitos y especificaciones del mismo.

Las iteraciones son representadas por una sucesión de cajas; el tamaño del sombreado interior ilustra la relación entre el tiempo de ejecución y las actividades que conforman esa iteración.

La figura 1.7 muestra la representación gráfica de tres iteraciones, cada una compuesta por

los mismos componentes pero con un sombreado acorde a la iteración de cada una de las actividades que compone la iteración.

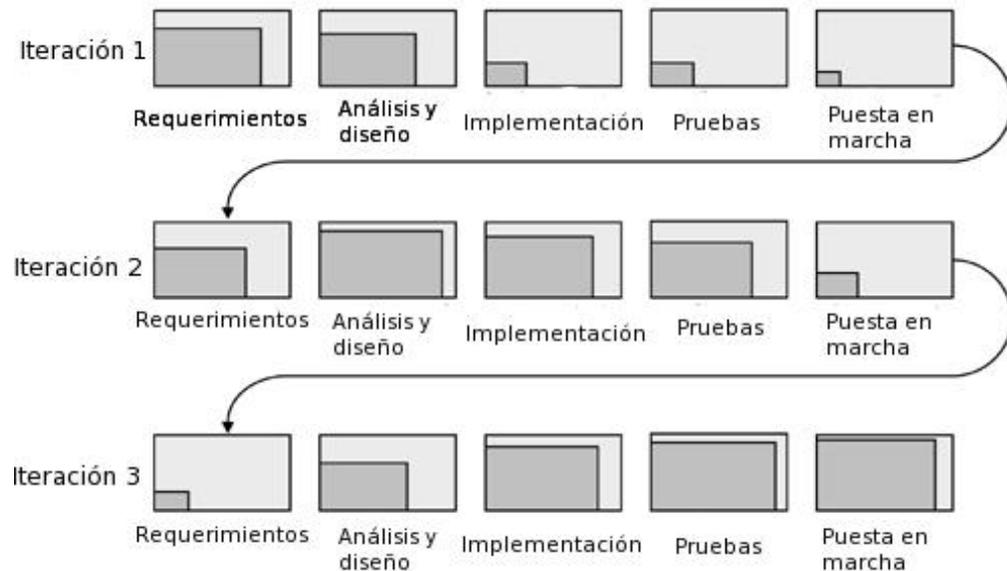


Figura 1.7

Ciclo iterativo compuesto por tres iteraciones

A continuación se enuncian una serie de aspectos de vital importancia en todo proceso iterativo:

- ✓ Definición clara de aspectos a realizar.
- ✓ Demostración del cumplimiento de los objetivos alcanzados.
- ✓ Cada iteración es resultado de otra iteración de menor importancia, la cual es evaluada de manera integral.
- ✓ Durante la iteración los artefactos son actualizados.
- ✓ Durante cada iteración se evalúa el sistema.

1.4 Fases

Todo proceso de desarrollo se encuentra formado por un conjunto de actividades realizadas por medio de un orden establecido mediante un plan de proyecto que tiene como premisa establecer las metas y objetivos de cada iteración. A su vez cada iteración está formada por un conjunto de fases. Las fases proporcionan la definición de *hitos*² necesarios para continuar con la siguiente iteración.

La base del desarrollo iterativo se encuentra conformada por fases e iteraciones, los objetivos de cada fase son logrados mediante la ejecución de una o más iteraciones. El término de una fase es alcanzado una vez que se ha cumplido el hito que marca el fin y, para continuar es necesario realizar la evaluación que determine si los objetivos de la etapa se han cumplido satisfactoriamente.

El desarrollo de software a través de la metodología RUP es descompuesto en cuatro fases secuenciales: Inicio, Elaboración, Construcción y Transición.

La figura 1.8 muestra la relación entre fases e hitos establecidos en la metodología RUP.

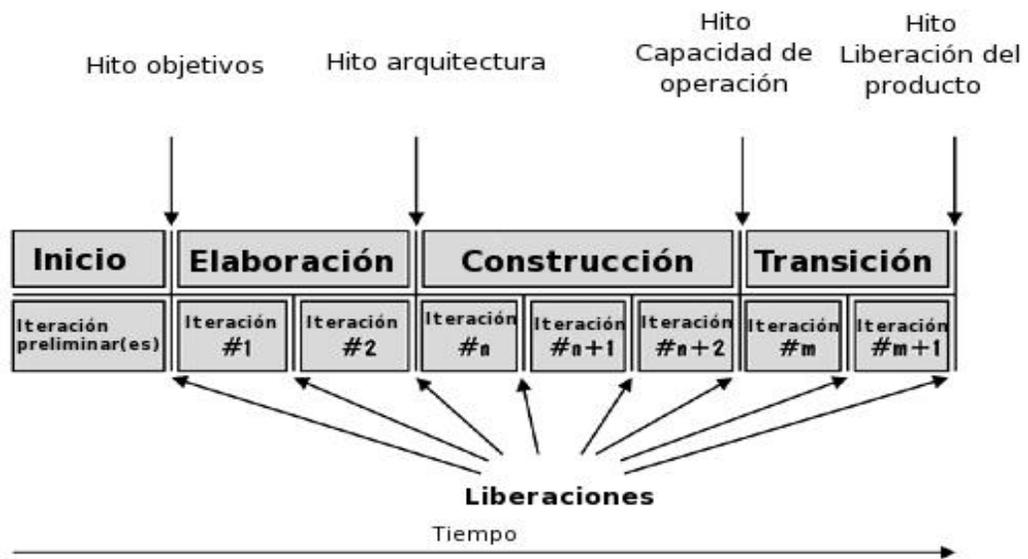


Figura 1.8

Diagrama de relaciones entre fases e hitos

²Hito: Hecho clave y fundamental dentro de un ámbito o contexto.

1.4.1 Inicio

La fase inicial es la etapa en la que se establecen los casos de negocio referentes al proyecto, concentra y establece los objetivos que determinan y justifican la viabilidad del proyecto. Los objetivos y criterios de evaluación correspondientes a la etapa inicial son alcanzados una vez que se realiza una definición clara y precisa de los alcances, criterios, características del sistema, calendario de prioridades, riesgos y entorno de desarrollo.

1.4.2 Elaboración

En la fase de elaboración se establece la arquitectura de software encargada de proporcionar la base para el diseño e implementación durante la etapa de construcción. Los objetivos y criterios de esta fase son cumplidos al momento de definir la arquitectura, requerimientos, visión del producto a elaborar, definición y plan de mitigación de riesgos.

1.4.3 Construcción

La fase de construcción aclara los requerimientos que no han sido definidos en su totalidad en la etapa de elaboración, realiza el desarrollo del sistema basado en la arquitectura especificada, centra la atención en la comprensión y solución de problemas detectados durante la fase de construcción, el cual dará como resultado un producto que será puesto en marcha. Los objetivos y criterios para esta etapa comprenden la producción y archivado de versiones preliminares, desarrollo de acuerdo a las especificaciones y planes establecidos durante la etapa de elaboración, así como un completo análisis, diseño, implementación y pruebas de cualquier funcionalidad desarrollada. Una vez superados estos puntos el sistema se encuentra listo para ponerse en marcha.

1.4.4 Transición

En la fase de transición el sistema es instalado en un entorno que permitirá al usuario realizar una serie de evaluaciones y pruebas que tendrán como finalidad la aprobación y perfeccionamiento del producto. Al finalizar la etapa de transición el proyecto debería de encontrarse en condiciones de ser cerrado. Los objetivos y criterios de revisión para esta fase comprenden la puesta en marcha y aprobación del sistema por parte del usuario final.

En la figura 1.9 se muestra el diagrama de flujo de actividades de la metodología RUP.

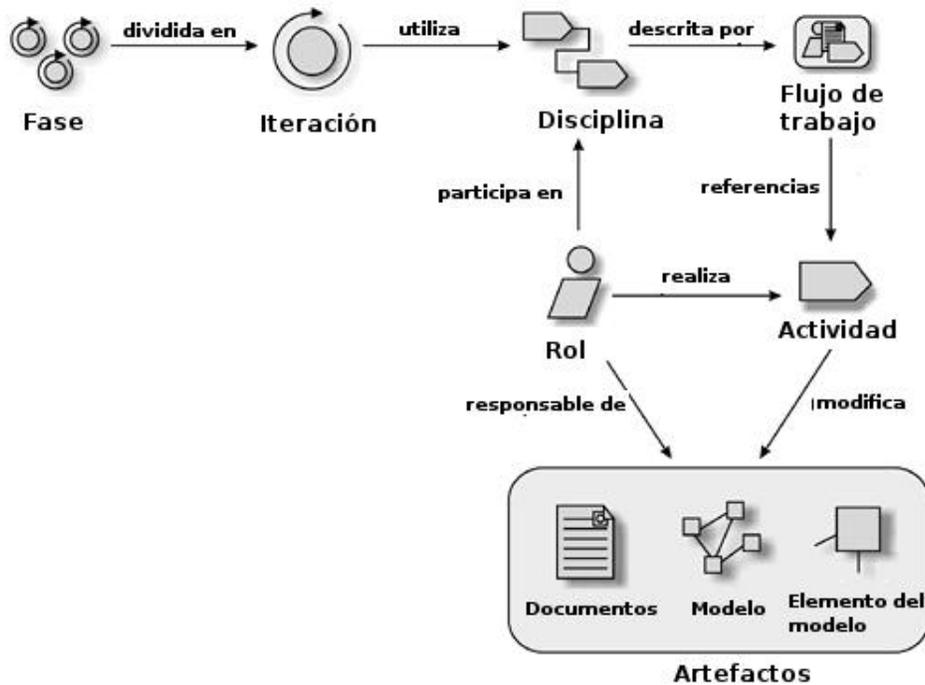


Figura 1.9

Diagrama de flujo metodología RUP

RUP es considerada una metodología integral y completa debido a la definición clara y concisa de actividades, artefactos y plantillas. A pesar de parecer cerrada y poco flexible RUP puede adaptarse fácilmente a cualquier tipo de desarrollo, independientemente de la organización para la cual se realice y del número y tamaño de procesos implicados, esta característica permite la personalización adecuada al tipo de proyecto y organización, por tal motivo el desarrollo de CIDE INTRANET se realizará con la adecuada personalización de RUP pero, sin soslayar el conjunto de *buenas prácticas*.

CIDE INTRANET

Plan de desarrollo de software

2. Plan de desarrollo de Software

2.1 Introducción plan de desarrollo de software

El Plan de Desarrollo de Software describe los procesos, alcances, definiciones, abreviaciones, referencias, así como una visión actual y completa de la organización del proyecto. Es elaborado por el equipo de desarrollo en conjunto con el personal académico y administrativo.

El proyecto se realizará mediante la metodología conocida como Proceso Unificado Racional (*Rational Unified Process* o de manera abreviada *RUP por sus siglas en inglés*), a continuación se enlistan las fases o artefactos que conforman dicha metodología.

- ✓ Plan de desarrollo de software
- ✓ Análisis de requerimientos
- ✓ Arquitectura de software

2.2 Propósito

El propósito del Plan de Desarrollo de Software es proporcionar la información requerida para la gestión del proyecto, describe el enfoque de desarrollo, identifica los recursos disponibles, consideraciones a seguir referentes a la arquitectura, lenguajes de programación, control de calidad y manejo de riesgos, así como otras actividades pertinentes en el desarrollo.

Los usuarios del Plan de Desarrollo del Software son:

- ✓ **Líder de proyecto:** Lo utiliza para describir y organizar recursos, metodologías y agenda; en general mantiene la administración del proyecto a través de la definición de procesos y roles, con lo cual se tiene un seguimiento en cada una de las iteraciones.
- ✓ **Equipo de desarrollo:** Lo utiliza para conocer y entender las metodologías y procedimientos a seguir durante la etapa de análisis y desarrollo, además de servirle como guía de referencia en cada una de las iteraciones del sistema.

2.3 Alcance

El Plan de Desarrollo de Software describe la estructura a seguir durante la creación del sistema. En el apartado “Visión” se definen las características del producto a desarrollar, lo cual constituye la base para la planificación de las iteraciones.

El Plan de Desarrollo de Software es elaborado mediante la captura de requerimientos obtenidos a partir de una serie de entrevistas con los usuarios y desarrolladores del anterior sistema. La obtención de requerimientos tiene como finalidad realizar una estimación del tiempo necesario para la construcción del sistema.

Una vez iniciado el proyecto y durante la fase inicial se generará el apartado “Visión”, el cual se utilizará con la finalidad de ajustar este documento de acuerdo a los requerimientos previamente establecidos.

Organización y estructura

A continuación se presenta la descripción de la estructura del proyecto:

Vista General del Proyecto.- Proporciona una descripción del propósito, alcances y objetivos. Se establecen los artefactos que serán producidos durante el proyecto.

Organización del Proyecto.- Describe la estructura organizacional, roles y actividades del equipo de desarrollo.

Gestión del Proceso.- Planificación estimada en la cual se definen las fases e hitos del proyecto, así como la descripción del seguimiento en cada iteración.

Planes y Guías de aplicación.- Proporciona una vista global del proceso de desarrollo de software, incluyendo métodos, herramientas y técnicas que serán utilizadas.

3. Vista General del proyecto

3.1 Propósitos, Alcance y Objetivos

CIDE INTRANET tiene como propósito ser una herramienta en la cual se concentren las tareas realizadas con mayor frecuencia por los usuarios del Centro de Investigación y Docencia Económicas “CIDE”. El sistema tiene como premisa ser multiplataforma, debido a que se cuenta con una amplia variedad de plataformas dentro de la institución. El sistema será implementado en un entorno Web, por consiguiente se hará uso de los estándares empleados por los principales navegadores Web.

Tecnológicamente se tiene como propósito el acoplamiento con el diseño en el actual Sistema Gestor de Bases de Datos (SGBD), el cual se encuentra operando y concentra la información correspondiente a usuarios, proveedores, alumnos, así como registros acumulados durante la operación del sistema anterior.

CIDE INTRANET establecerá las reglas de negocio mediante actividades como:

- ✓ Análisis de requerimientos
- ✓ Análisis de procesos.
- ✓ Análisis y revisión del código fuente de la aplicación anterior.
- ✓ Análisis e implementación de procedimientos almacenados en el SGBD.

En la primera etapa se tiene como objetivo el desarrollo de los siguientes módulos:

Módulo de requerimientos: Entre sus actividades se encuentran la captura, consulta, aprobación y rechazo de requerimientos. El conjunto de procedimientos y reglas que conforman el módulo serán analizados y aplicados de acuerdo al tipo de requerimiento solicitado.

Módulo de alumnos deudores: Encargado de la gestión y administración de los adeudos generados por los alumnos, entre sus actividades se encuentran las siguientes: captura, consulta, actualización y cancelación de adeudos en diferentes áreas.

Además de los módulos mencionados, la primera etapa contempla la implementación de los mecanismos y políticas de seguridad para el acceso y resguardo de información, las cuales se aplicarán desde la perspectiva del cliente/servidor.

La segunda etapa contempla el desarrollo e implementación del módulo denominado *buzón de alumnos*, su función será la de mantener informados a los usuarios de aquellos alumnos con cambio de estatus.

A continuación se presenta una propuesta de los procedimientos implicados en los subsistemas pertenecientes a la primera y segunda fase.

a) La Gestión de requerimientos incluye:

- ✓ Procedimiento de captura de requerimientos.
- ✓ Procedimiento de autorización y rechazo de requerimientos.
- ✓ Procedimiento de consulta requerimientos.

b) La Gestión de adeudos incluye:

- ✓ Procedimiento de captura de adeudos.
- ✓ Procedimiento de cancelación de adeudos.
- ✓ Procedimiento de consulta de adeudos.

c) Gestión buzón de alumnos

- ✓ Procedimiento de consulta para cambio de estatus.
- ✓ Procedimiento de firma cambio de estatus.

3.2 Suposiciones y Restricciones

A continuación se enuncian las suposiciones y restricciones del sistema, las cuales derivan directamente de las entrevistas realizadas al personal que hará uso del sistema.

a) Deben contemplarse las implicaciones de los siguientes puntos críticos:

- ✓ Compatibilidad con el diseño actual de la base de datos.
- ✓ Sistemas seguros: protección de información, seguridad en las transmisiones de datos, etcétera.
- ✓ Gestión de flujos de trabajo, seguridad de transacciones e intercambio de información.
- ✓ Adaptación a la normativa de Protección de Datos.

b) Requerimientos

- ✓ Generación de folios temporales para requerimientos no oficiales.
- ✓ Impresión de formatos de acuerdo al tipo de requerimiento.
- ✓ Visibilidad de requerimientos acorde al perfil de usuario.

c) Adeudos

- ✓ Búsqueda de alumnos mediante diversos criterios.
- ✓ Captura de adeudos en múltiples áreas.
- ✓ Consulta de alumnos no deudores.

d) Buzón de alumnos

- ✓ Agrupación por niveles (licenciatura, maestría, doctorado)
- ✓ Visualización de la información principal de los alumnos: nombres, apellidos y número de matrícula).

3.3 Entregables del proyecto

A continuación se indican y describen los artefactos pertenecientes a la metodología RUP, es preciso destacar que de acuerdo a la filosofía de RUP <<y de todo proceso iterativo e incremental>>, todos los artefactos son objeto de modificaciones a lo largo del proceso de desarrollo, con lo cual, sólo al término del proceso existiría una versión definitiva y completa de cada uno de ellos. Sin embargo, el resultado de cada iteración y los hitos del proyecto están enfocados a conseguir un cierto grado de plenitud y estabilidad de los artefactos.

Plan de desarrollo de software: En esta sección se definen los procesos y alcances del sistema, se proporcionará la información necesaria para controlar el proyecto, así como la definición de las características del producto a desarrollar.

Análisis de requerimientos: Es un modelo de las funciones de negocio vistas desde la perspectiva de los actores externos (Agentes de registro, usuarios finales, otros sistemas etcétera). Permite situar al sistema en el contexto organizacional haciendo énfasis en los objetivos de este ámbito. Este modelo se representa con un Diagrama de Casos de Uso mediante estereotipos específicos. Se delimitan los alcances del sistema y se definen de manera general los principales módulos del sistema.

Arquitectura de software: Describe los factores estáticos y dinámicos más significativos del sistema, es realizado con base a las necesidades de la institución y, está influenciado por diversos factores como: la plataforma de software en la que se ejecutará, el patrón de desarrollo, la disponibilidad de los componentes reutilizables, etcétera. La arquitectura de software es considerada como la vista del diseño completo con las características más importantes, dejando de lado los detalles.

3.4 Evolución del plan de desarrollo de Software

El Plan de Desarrollo de Software contempla la revisión periódica del documento, la cual permitirá realizar las adecuaciones necesarias de acuerdo a la etapa en que se encuentre el

desarrollo del sistema. Las revisiones permitirán actualizar la información en caso de no haber sido incluida por omisión o debido a una actualización en el flujo del sistema.

4. Organización del proyecto

4.1 Roles y Responsabilidades

A continuación se describen las principales responsabilidades de cada uno de los integrantes en el equipo de desarrollo durante las fases de inicio y elaboración de acuerdo con los roles que desempeñan en RUP.

Líder de Proyecto: Asigna los recursos, gestiona las prioridades, coordina las interacciones con los clientes y usuarios, mantiene al equipo del proyecto enfocado en los objetivos y establece un conjunto de prácticas que aseguran la integridad y calidad de los artefactos del proyecto. Además, el líder se encargará de supervisar el establecimiento de la arquitectura del sistema, así como la planificación y control del proyecto.

Analista de Sistemas: Entre sus tareas se encuentran la captura, especificación y validación de requerimientos, mismos que obtendrá al interactuar con los usuarios en las entrevistas. Apoya en la elaboración de las pruebas funcionales y el modelo de datos.

Programador: Se encarga de la construcción de prototipos. Colabora en el modelado de datos, así como validaciones con el usuario, gestión de requerimientos y cambios.

Ingeniero de Software: Se encarga de realizar las pruebas funcionales y contribuir con la documentación de apoyo al usuario final.

5. Gestión del proceso

5.1 Estimaciones del Proyecto

El tiempo para el desarrollo e implementación de la primera fase contempla un período de cuatro meses posteriores al levantamiento de requerimientos. El desarrollo del sistema será precedido por la elaboración del análisis de requerimientos y la arquitectura de software. A través de dichos documentos se establecen las especificaciones del sistema, así como los requisitos funcionales.

Al término del período establecido para el desarrollo e implementación de la primera fase se realizará la capacitación tanto para el personal académico como administrativo y, a

través de la capacitación se conocerá la opinión de los usuarios. La opinión del usuario mostrará si el sistema cumple con la especificación planteada en la arquitectura de software; en este contexto cabe mencionar que la especificación es elaborada a partir de los requerimientos capturados en las entrevistas realizadas al personal académico y administrativo.

La implementación de la segunda fase es estimada tres meses después de que se ha liberado la primera versión, durante el transcurso de este período se incluye una etapa de detección y corrección de fallas en la primera versión del sistema, así mismo comprende el desarrollo de la segunda fase, la cual está constituida por el módulo denominado *Buzón de alumnos*.

5.2 Plan de Proyecto

En esta sección se presenta la organización en fases e iteraciones del proyecto y se desglosan las tareas implicadas dentro de cada iteración.

5.2.1 Plan de las fases

El desarrollo se llevará a cabo mediante una serie de fases, las cuales implicarán una o más iteraciones en cada una de ellas. La figura 2.1 muestra un estimado de la distribución de tiempos y el número de iteraciones de cada fase.

Fase	Nro. Iteraciones	Duración
Fase de Inicio	1	3 semanas
Fase de Elaboración	1	2 semanas
Fase de Construcción	2	7 semanas
Fase de Transición	1	2 semanas

Figura 2.1

Estimación en la distribución de tiempos e iteraciones

5.2.2 Objetivos de la iteración

Fase de Inicio: En esta fase se desarrollarán los requisitos del producto desde la perspectiva del usuario, los cuales serán establecidos en el artefacto *Visión* perteneciente al

análisis de requerimientos. Los principales casos de uso serán identificados y se hará un refinamiento del Plan de Desarrollo del Proyecto. La aceptación del Plan de Desarrollo por parte del usuario marca el final de esta fase.

Fase de Elaboración: En esta fase se analizan los requisitos y se desarrolla un prototipo de arquitectura que incluye las partes más relevantes y/o críticas del sistema. Al final de esta fase se obtiene la primera versión estable del sistema. La revisión y aceptación del prototipo de la arquitectura del sistema marca el final de esta fase.

La primera iteración tendrá como objetivo la identificación y especificación de los principales casos de uso, así como su realización preliminar en el Modelo de Análisis/Diseño, también permitirá hacer una revisión general del estado de los artefactos hasta este punto y ajustar <<si es necesario>> la planificación para asegurar el cumplimiento de los objetivos.

Fase de Construcción: En esta fase de construcción se terminan de analizar y diseñar todos los casos de uso, además, en esta etapa también se elabora el material que servirá de apoyo para la capacitación de usuarios.

Al término de esta etapa se produce la primera versión del sistema la cual será evaluada a través de una serie de pruebas realizadas por el personal académico y administrativo. El hito que marca el fin de esta fase es la obtención de una primera versión con capacidad operacional y parcial del sistema, incluyéndose aquellos módulos considerados como críticos.

Fase de Transición: Se prepara la versión final para realizar la transición del sistema. Aquí se incluye la capacitación de los usuarios. Esta etapa concluye con la entrega de toda la documentación generada, manuales de instalación, así como el material de apoyo al usuario (video tutoriales) y con la implementación del sistema en los servidores de producción.

5.3 Seguimiento y Control del Proyecto

5.3.1 Gestión de Requisitos

Los requisitos del sistema son especificados en el apartado *Visión*. Los cambios en los requisitos serán gestionados mediante una Solicitud de Cambio, las cuales serán evaluadas y distribuidas para asegurar la integridad del sistema y el correcto proceso de gestión de configuración y cambios.

5.3.2 Control de Calidad

Los defectos encontrados en las revisiones tendrán un seguimiento, esto con el fin de asegurar la correcta solución con base a las reglas de negocio.

Para la revisión de cada artefacto y su correspondiente garantía de calidad se utilizarán las especificaciones establecidas en el Análisis de Requisitos, así como el modelo correspondiente al Caso de Uso implicado.

CIDE INTRANET

Análisis de requerimientos

6. Introducción Análisis de Requerimientos

El propósito de este documento es dar a conocer los objetivos y alcances del sistema que llevará por nombre CIDE INTRANET. Se realizará el análisis de las distintas iteraciones que formarán parte de los componentes del sistema propuesto, así como los casos de uso de primer nivel que conformarán el desarrollo de los módulos de captura de requerimientos, captura de multas y buzón de alumnos.

6.1 Propósito general del sistema

CIDE INTRANET es un sistema planeado para uso del personal académico y administrativo de la institución, el cual permitirá a los usuarios realizar con mayor eficiencia sus actividades, entre las cuales podemos mencionar: alta de requerimientos, captura de adeudos por parte de alumnos, consulta de estatus de alumnos, entre otras funciones.

Uno de los propósitos principales de CIDE INTRANET es ofrecer al usuario un sistema con una interfaz Web ya que se cuenta con el Sistema de Información Gerencial el cual es instalado en la máquina del usuario y, presenta una interfaz tipo Windows la cual no resulta agradable a un sector de usuarios a los cuales se encuentra dirigido. Al ser un sistema Web no requerirá ser instalado en la computadora de los usuarios, bastará con ingresar al portal de la intranet para realizar sus operaciones habituales.

Esta aplicación estará compuesta con un módulo llamado *requerimientos*, el cual se construirá a partir del análisis de los casos de uso recabados mediante entrevistas al personal académico y administrativo, así como las diferentes pruebas realizadas a SIGCIDE. De manera paralela, se realizará el análisis para los módulos de Adeudos de Alumnos y Cambio de Estatus, los cuales serán implementados por primera ocasión, se pretende obtener una interfaz sencilla y amigable para los usuarios que no poseen gran conocimiento en sistemas computacionales.

El sistema hará uso del diseño, funciones y procedimientos almacenados que actualmente se encuentran implementados, esto con el fin de no realizar una migración del gestor de base de datos y mantener una relación con el sistema anterior ya que, una vez procesado un requerimiento el área de recursos materiales le dará continuidad a través de SIGCIDE, este

procedimiento es ejemplificado posteriormente mediante un diagrama a bloques en el cual se muestra la interacción de ambos sistemas.

6.2 Alcance del sistema

CIDE INTRANET realizará los procesos que a continuación se describen:

Captura de requerimientos.- Consiste en realizar el registro de un requerimiento por parte de una o varias áreas que integran al CIDE, este registro contiene la información dependiendo del tipo de solicitud, así como los roles de usuario.

Actualización de requerimientos.- La actualización de requerimientos se ejecuta una vez que existe un registro, siempre y cuando el requerimiento no haya sido autorizado.

Consulta de requerimientos.- Permitirá obtener aquellos requerimientos que se han ingresado al sistema, las consultas se formulan bajo diversos criterios, entre los cuales podemos mencionar: áreas, rango de fechas, ejercicio fiscal, tipo de requerimiento, etcétera.

Captura de adeudos: Este módulo permitirá la captura de adeudos que hayan generado los alumnos en cualquier área dentro del CIDE.

Consulta de adeudos: La consulta dará a conocer si es que un alumno cuenta con un adeudo, en caso de contar con uno o más -en cualquiera de las áreas- se deberán especificar las fechas y motivos de los adeudos.

Cancelación de adeudos: En esta sección se permitirá a los administradores del sistema cancelar el adeudo, para ello será necesario indicar el motivo por el cual se está efectuando la cancelación.

Buzón de alumnos: Este módulo informará a los usuarios del sistema en caso de que algún alumno haya sufrido un cambio de estatus; se encontrará una señalización en la parte superior del menú y, estará agrupado dependiendo el grado: licenciatura, maestría, doctorado.

6.3 Objetivos

CIDE INTRANET tiene como objetivo ser un sistema WEB con una interfaz amigable para

el usuario, facilitando sus actividades mediante la implementación de los siguientes módulos: requerimientos, adeudos y buzón de alumnos.

A continuación se enlistan los objetivos de manera general:

- ✓ Ofrecer el acceso sin depender de alguna plataforma específica.
- ✓ Ofrecer una interfaz conocida por los usuarios.
- ✓ Estabilidad y manejo seguro de transacciones.
- ✓ Optimización y mejora de procedimientos y recursos.
- ✓ Integridad y aseguramiento en el manejo de información
- ✓ Aplicación y manejo de políticas de seguridad

6.4 Definiciones, acrónimos y abreviaturas

Centro de Investigación y Docencia Económicas (CIDE): Centro de investigación y educación superior especializado en ciencias sociales, orientado por estándares internacionales de calidad y financiado con recursos públicos.

El CIDE forma parte de la red de centros públicos de investigación coordinados por el CONACYT (*Consejo Nacional de Ciencia y Tecnología*), entre los principales objetivos del centro de investigación podemos mencionar los siguientes:

- ✓ Divulgar en la sociedad la ciencia y tecnología.
- ✓ Fomentar la tecnología local y adaptarla a la tecnología extranjera.
- ✓ Innovar en la generación, desarrollo, asimilación y aplicación del conocimiento de ciencia y tecnología.
- ✓ Vincular la ciencia y tecnología en la sociedad y el sector productivo para atender problemas.
- ✓ Crear y desarrollar mecanismos e incentivos que propicien la contribución del sector privado en el desarrollo científico y tecnológico.
- ✓ Incorporar estudiantes en actividades científicas, tecnológicas y de vinculación para fortalecer su formación.
- ✓ Fortalecer la capacidad institucional para la investigación científica, humanística y tecnológica.

- ✓ Fomentar y promover la cultura científica, humanística y tecnológica de la sociedad mexicana.

Las actividades del CIDE son tres: investigación científica; formación académica a nivel licenciatura, maestría y doctorado; y difusión de conocimiento socialmente útil. Estas actividades se llevan a cabo en seis divisiones académicas:

- ✓ Administración pública
- ✓ Economía
- ✓ Estudios Internacionales
- ✓ Estudios Jurídicos
- ✓ Estudios Políticos
- ✓ Historia

Requerimiento: El requerimiento es cualquier petición hecha por un área con el fin de surtir material o algún otro recurso para poder efectuar sus actividades.

Sistema de Información Gerencial (SIGCIDE): Sistema con el que actualmente se crean los requerimientos. Se encuentra desarrollado en Visual Basic.

Framework: Estructura de software mediante la cual otro proyecto de software es organizado y desarrollado, incluye soporte de librerías y otros componentes con la finalidad de obtener una integración estándar.

Versión: En el ámbito del desarrollo de software una versión es una representación numérica que indica el grado de desarrollo de un programa. La versión es comúnmente representada por dos dígitos separados por un punto, el cambio significativo de una versión ocurre cuando existe un cambio en el dígito de la izquierda.

Sistema de Gestión de Base de Datos (SGBD): En inglés: (**DataBase Managment System** abreviado DBMS) es una interfaz de software entre la base de datos, el conjunto de aplicaciones y el usuario. Para este desarrollo se empleara SQL Server en su versión 2008.

Procedimiento almacenado: Programa almacenado físicamente en la base de datos, el cual al ser ejecutado actúa directamente sobre los datos almacenados.

Disparador (trigger): Procedimiento almacenado, el cual es ejecutado al cumplirse cierta condición establecida.

Contingencia: Acto o hecho no presente en el funcionamiento habitual del sistema.

Control de versiones: Se denomina control de versiones a la gestión de los cambios que son realizados en el código fuente de un sistema.

Subversion: Software de sistema de control de versiones.

7. Sistema actual

El sistema actual denominado Sistema de Información Gerencial del CIDE (SIGCIDE) se encuentra desarrollado en Visual Basic, SIGCIDE es una aplicación cliente la cual es instalada en los equipos de aquellos usuarios con acceso al sistema.

A continuación se muestran los diagramas de actividades que corresponden a los módulos implementados en el Sistema de Información Gerencial del CIDE.

Módulo de requerimientos

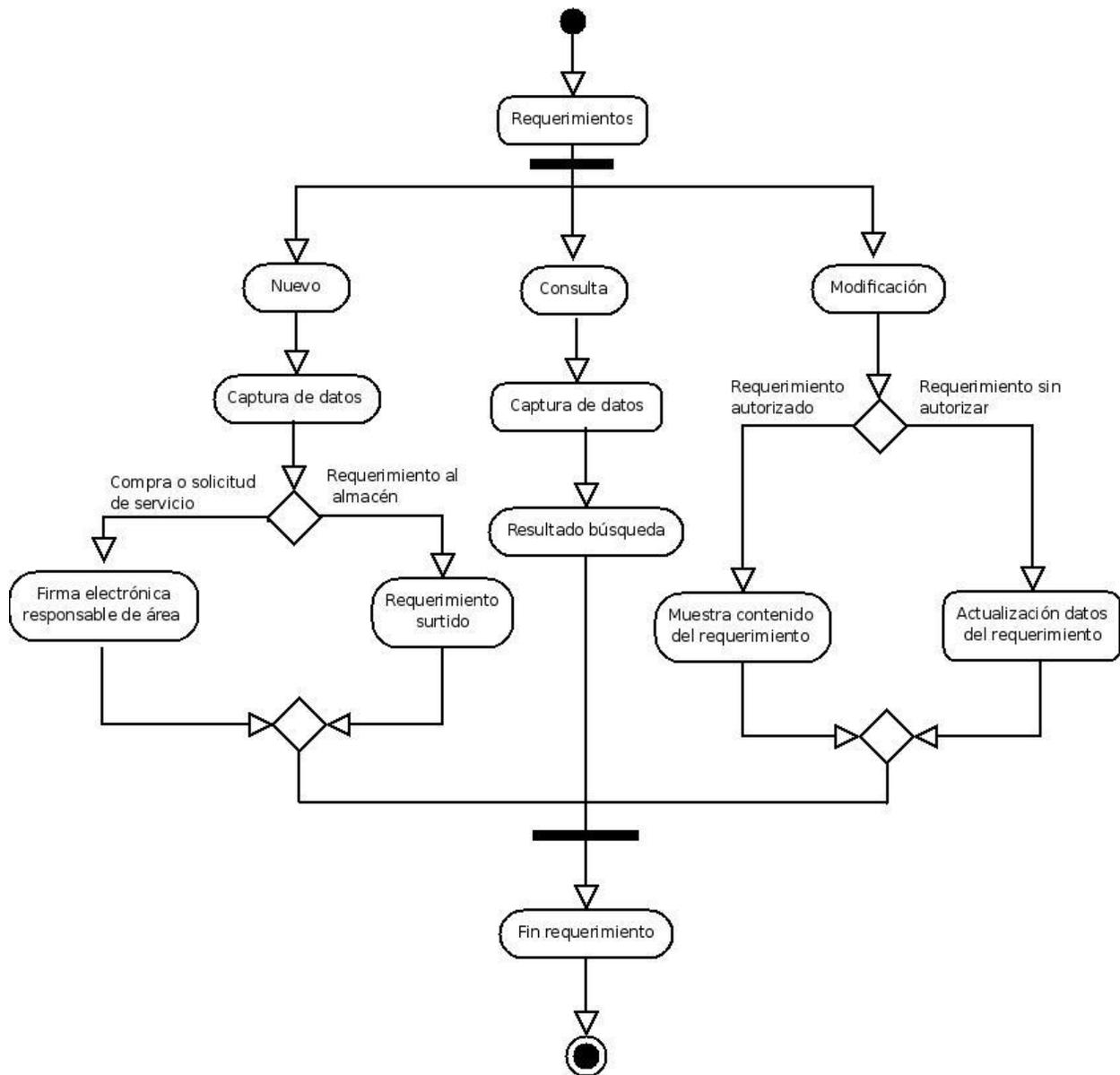


Figura 3.1

Diagrama de procesos modulo de requerimientos

Módulo alumnos deudores

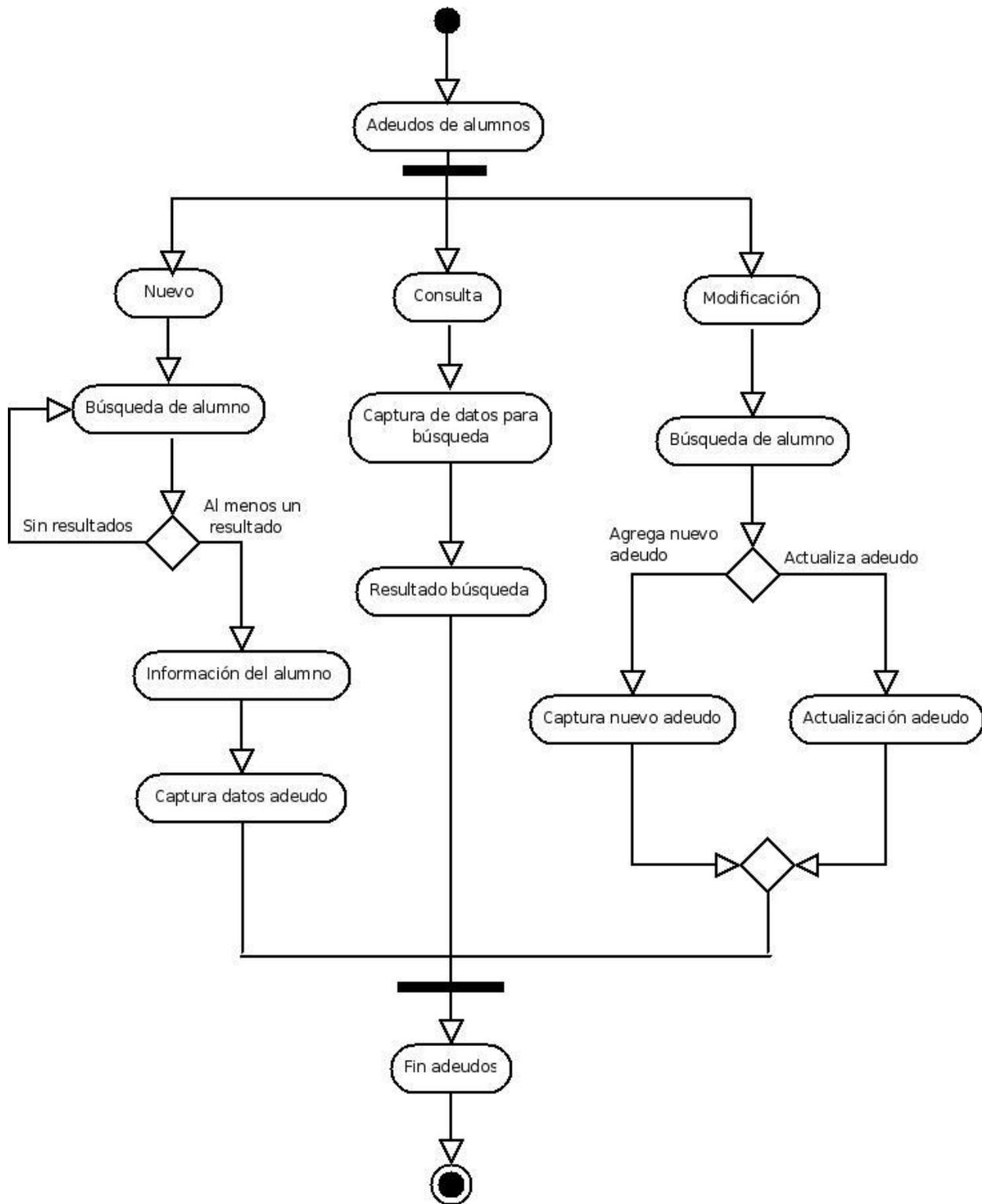


Figura 3.2

Diagrama de procesos módulo alumnos deudores

Módulo buzón de alumnos

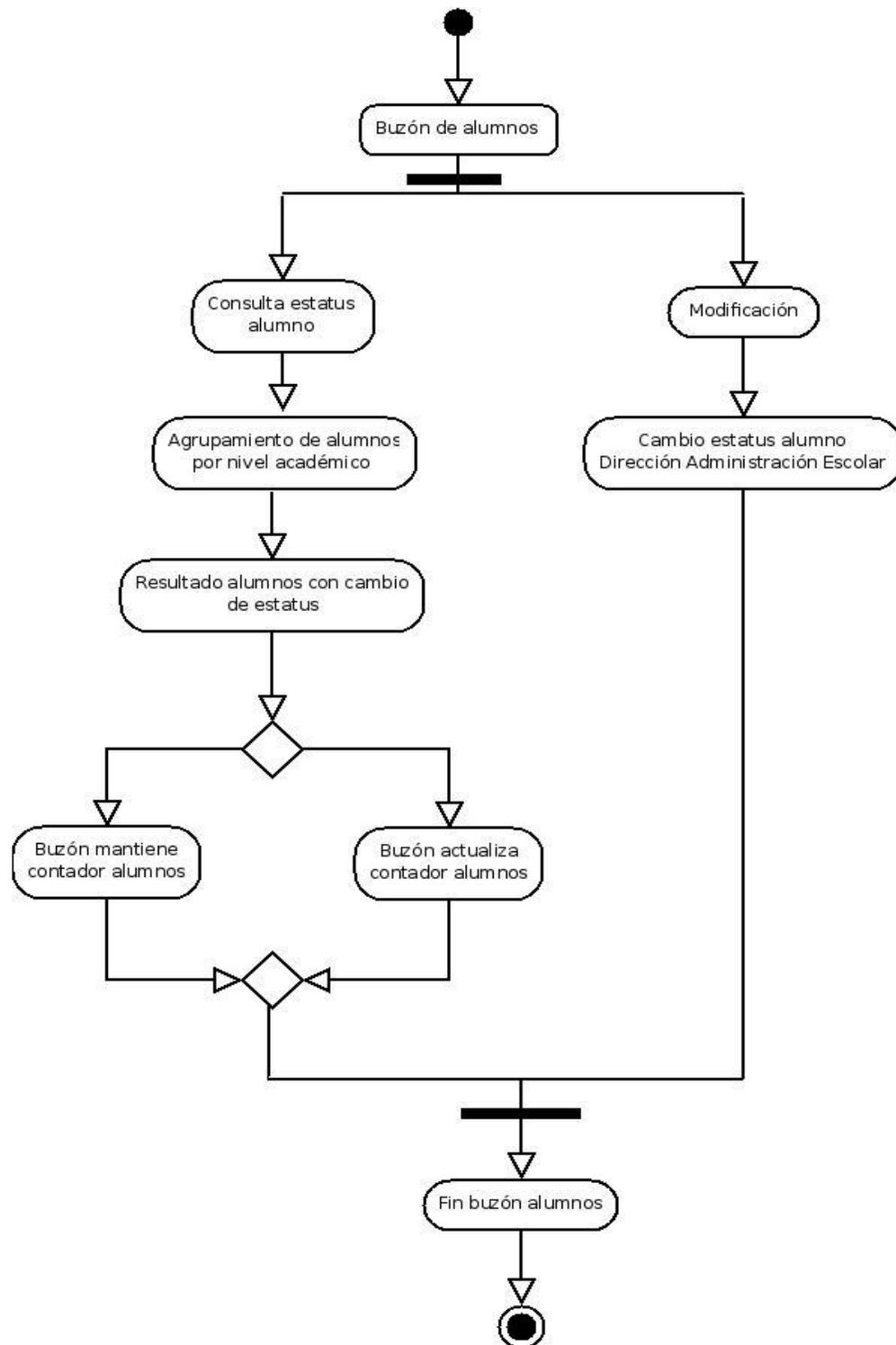


Figura 3.3

Diagrama de procesos buzón de alumnos

8. Sistema propuesto

8.1 Visión general

CIDE INTRANET ofrecerá al usuario final las siguientes ventajas:

- ✓ Acceso al sistema sin depender de alguna plataforma en específico.
- ✓ Gestión de recursos por parte del servidor y no del cliente.
- ✓ Actualizaciones del sistema sin necesidad de realizar reinstalaciones en las máquinas de los usuarios.
- ✓ Optimización de consultas y procedimientos almacenados, esto implica un menor tiempo de respuesta.
- ✓ Mayor seguridad a través del uso de sesiones.
- ✓ Interfaz amigable e intuitiva para los usuarios.

CIDE INTRANET trabajará con SIGCIDE al emplear en conjunto el recurso del servidor de base de datos. La base de datos no será el único punto en común; mediante el siguiente diagrama se ejemplifica la manera en que interactúan ambos sistemas para completar un requerimiento.

En la figura 3.4 se muestra el diagrama de procesos que corresponde a la captura de requerimientos, en el bloque superior se muestran los procesos correspondientes a la funcionalidad implementada por CIDE INTRANET. Una vez que el requerimiento ha sido firmado por el encargado del área solicitante, SIGCIDE retoma la información del requerimiento ingresado y continúa el proceso hasta surtir el requerimiento o el servicio al área que lo ha solicitado, siendo el área de recursos materiales quien finaliza el proceso.

Dentro de este proceso cualquier cambio realizado se reflejará inmediatamente en ambos sistemas, con esto evitaremos mantener sincronizadas dos o más bases de datos cada determinado tiempo.

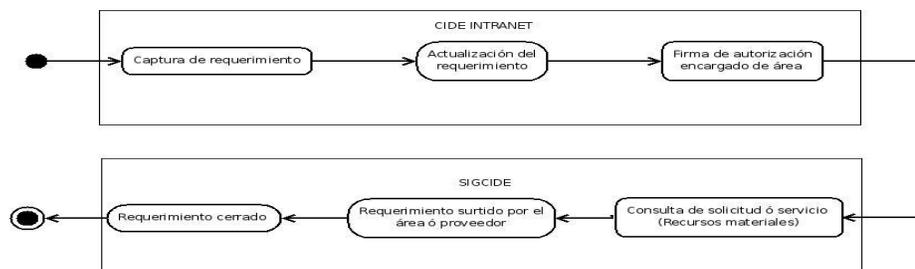


Figura 3.4
Diagrama de procesos CIDE INTRANET / SIGCIDE

El sistema está dirigido a todo usuario que cuente con acceso autorizado a SIGCIDE, en particular podemos destacar aquellos que únicamente acceden a los módulos de requerimientos y adeudos de alumnos, para estos usuarios se realizará una migración por completo. En el caso de usuarios con acceso a módulos implementados sólo en SIGCIDE, tanto el acceso como el funcionamiento de los módulos en CIDE INTRANET estará garantizado. Para este último caso ambos sistemas podrán convivir en un mismo entorno de trabajo, sin que esto cause una complicación para el usuario o la base de datos.

8.1.1 Especificaciones

El sistema arquitectónicamente tiene como principal objetivo ser multiplataforma para su ejecución, aunque la flexibilidad del lenguaje y el entorno de desarrollo elegido para su construcción permiten que el sistema sea programado desde cualquier arquitectura con soporte para la máquina virtual que provee *Sun Microsystems*. La base de datos se encontrará en un servidor con arquitectura diferente a la empleada en el servidor Web, por lo que, para establecer comunicación entre ambos servidores será necesario hacer uso de un conector que pueda ser ejecutado por el lenguaje de programación Java, el cual es asociado al gestor de base de datos.

CIDE INTRANET será desarrollado en una plataforma similar a la empleada en el servidor de producción, lo anterior permitirá tener plena confianza en qué tanto la implementación del Framework, así como la base de datos será completamente funcional al momento de la liberación del sistema, ya sea en el servidor de pruebas o desarrollo.

Todo sistema en condiciones normales se encuentra en continuo desarrollo, por tal motivo es susceptible a cambios que impliquen mejoras o corrección de errores detectados, ya sea en fase de pruebas o en el ámbito de producción con la finalidad de seguir un orden y control. Las versiones de código generadas serán almacenadas en un servidor de control de versiones.

Después de evaluar varias herramientas con el propósito de elegir la más adecuada, se ha optado por *subversión*, debido a sus principales características:

- ✓ Es de libre distribución.
- ✓ Tanto el cliente como el servidor son multiplataforma

- ✓ Permite la gestión de versiones para un mismo proyecto en diferentes ramas.

Subversion mantendrá como mínimo dos copias del código fuente, la versión estable tiene como finalidad ser el código fuente de la aplicación que se encuentra en el servidor de producción, en caso de presentarse una contingencia que afecte el funcionamiento del servidor de producción y por consiguiente de la aplicación, se procederá a realizar la restauración del mismo, para lo cual es necesario obtener una copia del código etiquetado como estable, realizar la compilación y al finalizar se obtendrá el ejecutable de la última versión etiquetada como estable.

La versión de desarrollo tiene como finalidad la realización de pruebas e implementación de nuevos módulos, esta versión será actualizada continuamente por el desarrollador con el propósito de mantener una copia de todo cambio realizado al código fuente; a partir de esta versión se realizarán las pruebas para comprobar su funcionalidad, en caso de existir algún error se llevarán a cabo las correcciones necesarias, al concluir las correcciones se repetirán las pruebas para demostrar que cumple con el funcionamiento establecido, una vez superadas las pruebas el nuevo código pasa a formar parte de la versión estable.

8.1.2 Descripción general

En este apartado se realiza el análisis y la descripción de los perfiles de usuario existentes dentro de CIDE INTRANET.

Encargado de área: Usuario designado dentro de la estructura jerárquica como director de área o de alguna división académica, este usuario podrá realizar las siguientes tareas en el sistema:

- ✓ Captura, modificación, consulta y autorización de requerimientos.
- ✓ Consulta del buzón de alumnos.
- ✓ Captura y actualización de adeudos.

Empleado: Este usuario puede ser administrativo o académico dependiendo su área o división a la que se encuentre adscrito; estará autorizado para efectuar las siguientes actividades:

- ✓ Captura, modificación y consulta de requerimientos.
- ✓ Captura y actualización de adeudos.

8.1.3 Diseño

En esta sección se llevará a cabo el análisis de los casos de uso detectados mediante una serie de entrevistas realizadas al personal académico, administrativo y desarrolladores de SIGCIDE, este análisis nos permitirá elaborar la capa conocida como lógica de negocio, la cual se encargará de gestionar los procesos que garanticen el adecuado funcionamiento del sistema y seguimiento de las reglas de negocio establecidas mediante los casos de uso. El análisis describe las principales características de los módulos que serán implementados así como la interacción con los perfiles de usuario previamente establecidos.

Modulo de requerimientos

El sistema será capaz de realizar de manera similar a SIGCIDE la captura y consulta de requerimientos independientemente de la plataforma. Se seguirán las restricciones de diseño ya establecidas en la base de datos, debido a que al realizar un requerimiento desde SIGCIDE este nuevo requerimiento y todos los anteriores deberán ser accesibles a través de CIDE INTRANET, la interfaz será similar a la de SIGCIDE, en la cual se encuentra un formulario con campos iniciales en donde se ingresa la información y posteriormente el sistema mostrará los campos correspondientes con el tipo de requerimiento seleccionado. En la interfaz se presentarán colores y tonos acordes a la institución con la finalidad de que sea atractiva para los usuarios.

Para el módulo se encuentran dos perfiles:

Usuario Solicitante: Puede realizar las funciones de: alta, modificación y consulta de requerimientos. Para que un requerimiento sea oficial será necesario ingresar el detalle del requerimiento, el cual contiene la información que permitirá generar un folio oficial, mismo que indicará al usuario encargado de área los requerimientos pendientes por autorizar.

Encargado de área: Cuenta con los mismos permisos que un usuario solicitante, además, está encargado de revisar si un requerimiento es válido, para ello es necesario revisar la información ingresada, si la información es válida el encargado procederá a realizar la firma electrónica ingresando su clave de usuario y contraseña en el requerimiento en proceso.

Modulo de alumnos deudores:

El módulo de alumnos deudores será administrado por los usuarios pertenecientes a las

áreas de administración escolar, biblioteca e idiomas; los usuarios con acceso a este módulo serán capaces de añadir multas de tipo económicas o administrativas, las cuales serán generadas por incumplimiento en la entrega de materiales o documentos en alguna de las áreas mencionadas anteriormente.

Una vez que el alumno ha cubierto el adeudo se procederá a cancelar la multa en el sistema, para ello será necesario ingresar un comentario en el cual se indique el motivo por el cual se cancela el adeudo.

Para este módulo se cuenta con el siguiente perfil:

Encargado de área: Cuenta con los permisos para agregar, cancelar y realizar búsquedas a través de diferentes criterios, entre los cuales se encuentran el nivel académico (licenciatura, maestría, doctorado), generación, nombre, apellido paterno, apellido materno, sólo por mencionar algunos, en caso de que el alumno cuente con adeudos en diversas áreas el resultado de la búsqueda mostrará el compendio de adeudos generados por el alumno.

Si el resultado de una búsqueda al ingresar la matrícula de un alumno es nulo, se emitirá un mensaje que servirá como comprobante de no adeudos para dicho alumno.

Buzón de alumnos:

El buzón de alumnos será un módulo que permitirá conocer los cambios de estatus de los alumnos de licenciatura, maestría y doctorado. Al ingresar al sistema se informa de la cantidad de alumnos con cambio en su estatus, la información será mostrada en un recuadro ubicado en el menú principal. Al presionar el recuadro se expondrá la información de los alumnos agrupándolos por nivel académico, en la parte superior e inferior se mostrará un botón, al ser presionado confirmará que el usuario se encuentra enterado del cambio de estatus de los alumnos enlistados.

En este módulo se cuenta con un perfil:

Encargado de área: Este usuario se encargará de revisar periódicamente la aplicación con la finalidad de conocer si existen alumnos con cambio de estatus, cada usuario llevará el conteo de su buzón de manera individual, en caso de que el usuario no revise de manera habitual su buzón éste acumulará los registros.

8.2 Requisitos funcionales

Requisito #1: Compatibilidad con el diseño actual de la base de datos, el seguir el diseño de la base de datos permitirá mantener la compatibilidad entre los requerimientos generados

por SIGCIDE y CIDE INTRANET, con la finalidad de mantener la compatibilidad en ambos sistemas se hará uso de: funciones, disparadores (triggers) y procedimientos almacenados ya contenidos en la base de datos.

Requisito #2: Captura de requerimientos, se realizará de manera similar a SIGCIDE, la interfaz que se presentará al usuario deberá ser lo más agradable e intuitiva posible, con ello se espera que la curva de aprendizaje para el usuario sea mínima.

Requisito #3: Consulta de requerimientos, para realizarse será necesario ingresar uno o más parámetros de búsqueda, entre los cuales podemos mencionar los siguientes: un rango de fechas, una área en específico o un tipo de requerimiento. Los resultados de las consultas serán exportados con la información más relevante de cada requerimiento.

Requisito #4: Autorización de requerimientos, al ingresar al sistema los usuarios encargados de área serán notificados de la existencia de uno o varios requerimientos que se encuentren en espera de autorización, si el requerimiento cumple las especificaciones, el usuario realizará la autorización, en caso de no ser así será rechazado.

Requisito #5: Captura de adeudos, se realizará por los usuarios autorizados pertenecientes a cada área, se deberá incluir un comentario que indique el motivo por el cual se ha generado el adeudo.

Requisito #6: Cancelación de adeudos, una vez que el alumno realiza el pago o entrega del material que haya generado el adeudo, el encargado de área deberá proceder a cancelar la multa, será necesario ingresar una breve descripción y confirmación por parte del usuario.

Requisito #7: Buzón de alumnos, su función será mantener informados a los usuarios al respecto de cambios en el estatus de los alumnos. Esta información se agrupará por niveles académicos para ser mostrada al usuario de forma más simple y organizada.

8.3 Requisitos no funcionales

Se describen en lenguaje natural los requisitos de usuario no relacionados directamente con la funcionalidad del sistema. El formato de descripción será el mismo que el utilizado en la sección de requisitos funcionales. Los requisitos se agrupan por categorías. A continuación se proporcionan algunas de las categorías más habituales de este tipo de requisitos.

8.3.1 Facilidad de uso (“usability”)

El sistema mostrará una interfaz amigable e intuitiva para el usuario. La interfaz estará compuesta por un conjunto de campos de uso común en toda aplicación Web, con el objetivo de que el usuario se sienta familiarizado con la interfaz. Se hará uso de los colores institucionales en las ventanas principales, dichos colores son el verde y el blanco, en las ventanas emergentes y subsecciones se agregarán colores y tonos diferentes al de las pantallas principales, con la finalidad de mantener una clara diferencia entre las distintas secciones del sistema.

8.3.2 Confiabilidad

Disponibilidad: El sistema deberá permanecer disponible en el horario administrativo de la institución, el cual comprende de 8:00a.m. a 18:00p.m. El mantenimiento podrá efectuarse antes y después de este horario, en caso de existir una contingencia que afecte el funcionamiento del sistema será necesario atenderlo al momento. Se estima que el sistema permanecerá en línea el 95% de tiempo a pesar de contar con un horario administrativo, debido a que el 5% restante se empleará para realizar los cambios y actualizaciones pertinentes.

Tiempo medio de reparación: El tiempo estimado para que el sistema se encuentre fuera de línea debido a una falla, será menor a diez minutos. En caso de existir una contingencia que afecte al servidor de producción y, por consecuencia su operación se vea afectada por más de veinte minutos, será necesario habilitar el servidor de pruebas como sistema de respaldo para no afectar por mucho tiempo la sesión de los usuarios.

8.3.3 Performance

El tiempo de respuesta de la aplicación se evaluará con base al tiempo empleado para realizar determinadas operaciones en el sistema, ya que existirán diferentes tiempos de respuesta, esto se debe a que en algunos casos será necesario realizar consultas de catálogos lo cual implicará un mayor tiempo de respuesta debido a que el sistema deberá crear los objetos necesarios para ser enviados en respuesta a la petición. Cuando no sea necesario realizar consultas o que estas no impliquen catálogos con un número de registros elevado, el tiempo de respuesta será de microsegundos.

La capacidad de usuarios que podrán acceder se establecerá con base al número de peticiones definidas en el servidor Tomcat, la cantidad de usuarios que pueden acceder de manera recurrente al sistema será definida tanto por la infraestructura en la cual se encuentre alojado el servidor como por el número de conexiones permitidas en el servidor Web. Con base en dicha referencia se estima que el porcentaje de usuarios a los cuales el sistema permitirá una conexión simultánea será del 100%. La estimación se realiza a partir del hecho de que el sistema será de uso exclusivo para el personal académico y administrativo.

El sistema permanecerá alojado en dos servidores, el primero se encargará de administrar el servidor Web, en tanto que el segundo almacenará la base de datos. Este esquema permite el balanceo de transacciones en el sistema. El balanceo de transacciones tiene como resultado un óptimo desempeño en el sistema aún cuando el número de usuarios conectados de manera simultánea sea elevado.

8.3.4 Restricciones de diseño

El desarrollo se realizará haciendo uso del lenguaje de programación Java, el motivo principal de esta elección es el hecho de ser un lenguaje multiplataforma, por lo cual no es necesario depender de una plataforma específica para el desarrollo o implementación del sistema.

La construcción del sistema se realizará con una herramienta para el desarrollo de aplicaciones Web bajo el patrón MVC (*Modelo Vista Controlador*), siguiendo la estructura del Framework llamado Struts, permitiendo reducir el tiempo de desarrollo a través de la separación de los siguientes componentes:

Modelo: Representado por el Sistema de Gestión de Base de Datos.

Vista: Interfaz visible por el usuario, en este caso representada a manera de páginas HTML.

Controlador: Interfaz encargada de la gestión de eventos recibidos a través de la interfaz de usuario.

El uso de esta arquitectura de software permite la gestión de capas individuales, por tal motivo al efectuar un cambio en cualquier interfaz, no se afectará de manera directamente

otro componente del sistema.

Para la etapa de codificación se ha elegido el entorno de desarrollo integrado NetBeans, al igual que Java pertenece a la familia *Sun Microsystems*, cuenta con soporte para múltiples arquitecturas, así como excelente integración con el Framework Struts, y para el servidor de aplicaciones Web Tomcat, este último se encargará de alojar la aplicación una vez concluida.

La evaluación de consultas se realizará mediante SQL Server Management Studio, este cliente permitirá conocer la estructura de la base de datos, funciones, disparadores de eventos (triggers) y procedimientos almacenados.

Al término de la fase de construcción se realizarán las pruebas funcionales para conocer si se ha cumplido de manera satisfactoria la especificación de cada requerimiento. Dichas pruebas se llevarán a cabo en servidores que cuenten con software y hardware similar al utilizado en el ámbito de producción.

8.3.5 Seguridad

En este apartado se mencionará la seguridad que se implementará para salvaguardar la integridad de la información, así como para mantener el óptimo funcionamiento del sistema, por tal motivo es necesario realizar el análisis de la seguridad desde la perspectiva del servidor y de la aplicación Web.

El servidor Web Tomcat será alojado en un servidor Debian GNU/Linux, mismo que filtrará las conexiones mediante la implementación de un firewall dentro del mismo servidor, al ejecutar este filtrado se limitará el acceso sólo a conexiones provenientes desde la intranet de la institución, el filtrado permitirá evitar algún acceso no autorizado desde cualquier punto no ubicado en las instalaciones del CIDE. La actualización del servidor se realizará de manera periódica según los ciclos de liberación de la rama estable en la distribución Debian GNU/Linux.

El servidor de Base de Datos SQL Server se encuentra implementado en una arquitectura Windows Server 2008, el cual periódicamente ha recibido actualizaciones con el fin de mantener la confidencialidad de datos. La configuración del servidor solo permitirá el acceso a los usuarios a través de SIGCIDE o CIDE INTRANET, dicha política de acceso permitirá garantizar la coercitividad de la información contenida en la base de datos.

El acceso a la red se permitirá al ingresar y verificar el nombre de usuario y contraseña previamente proporcionados, para comprobar la certeza de ambos datos se efectuará una consulta a la base de datos, en caso de ser correcta dicha información, se procederá a obtener el perfil del usuario, con base al perfil se otorgará el acceso a los módulos correspondientes en la base de datos. Por tal motivo resulta de vital importancia contar con la definición precisa de cada perfil de usuario con acceso al sistema.

La aplicación establecerá conexiones a la base de datos por medio de la gestión del controlador, la conexión establecida permanecerá activa mientras sea requerida, una vez finalizada la actividad por parte del controlador la conexión será cerrada, sin embargo, la sesión de usuario permanecerá activa mientras no expire el tiempo de inactividad de la sesión.

El sistema definirá un período de inactividad con cada inicio de sesión por parte de un usuario, dicho período como su nombre lo indica monitoreará el tiempo que el sistema se encuentre inactivo. En caso de que el tiempo de sesión expire, la sesión del usuario se invalidará, de tal forma que todos los valores almacenados previamente como el perfil de usuario y valores aún no guardados en la base de datos se perderán, por lo cual será necesario ingresar de nueva cuenta el usuario y contraseña para generar una sesión nueva.

8.3.6 Documentación de usuario y sistemas de ayuda

El sistema ofrecerá ayuda al usuario a través de una serie de video-tutoriales, los cuales estarán disponibles en el menú principal de la aplicación, la reproducción se realizará en una ventana emergente y permitirá adelantar, atrasar, detener y reproducir el video tantas veces se deseé, cada video será explicado a través de mensajes con forma de globo, en dichos mensajes se describe de manera breve pero concisa el procedimiento realizado.

Los tutoriales siempre se encontrarán disponibles al ingresar al sistema, su objetivo es resolver en cualquier momento las dudas del usuario, así como mantener la confidencialidad del sistema, con lo anterior se asegurará que dicho material será de uso exclusivo para los usuarios del sistema.

8.3.7 Interfaces

En esta sección se definen las interfaces que soportará la aplicación. Contendrá la especificación de protocolos, puertos y direcciones lógicas de forma que el software pueda ser desarrollado y verificado con base a estos requisitos.

8.3.7.1 Interfaces de usuario

El sistema empleará una interfaz con colores referentes a la institución, en el caso de ventanas emergentes o secciones secundarias los tonos serán acordes a la base de colores establecida en la pantalla principal de requerimientos.

Captura / consulta de requerimientos:

La pantalla de requerimientos estará constituida por elementos de uso común en los formularios de cualquier aplicación Web, entre ellos podemos mencionar los siguientes: casillas de texto, campos de selección y campos de selección múltiple, el acceso o las opciones de cada elemento se determinará con base al perfil de usuario y las reglas de negocio establecidas por los casos de uso.

La búsqueda se realizará a través de un formulario en el cual se mostrarán los campos empleados como parámetros de búsqueda, en caso de que los parámetros ingresados coincidan con uno o más requerimientos se mostrará un listado con la información más relevante de aquellos requerimientos que hayan coincidido con los parámetros ingresados por el usuario, mediante esta consulta el usuario accederá a la actualización o consulta detallada del requerimiento.

Captura / consulta de adeudos

La interfaz para la captura de adeudos estará constituida por un formulario en el cual es solicitada la siguiente información: nombre de alumno, tipo de adeudo, estatus del adeudo, importe y descripción del adeudo. En el caso del tipo de adeudo y el estatus, las opciones mostradas en el catálogo serán de acuerdo al perfil y adscripción de área del usuario.

La interfaz mostrada en la consulta de adeudos será similar a la presentada en la búsqueda de requerimientos, la búsqueda se realizará con base a los criterios ingresados por el usuario, dichos criterios incluyen la búsqueda por matrícula, nombre del alumno, apellido paterno, apellido materno, generación y estatus, dentro del estatus podemos mencionar la búsqueda por alumnos con adeudos, alumnos sin adeudos o ambos.

Buzón de alumnos

El buzón de alumnos se encontrará ubicado en la parte superior del menú principal del sistema, el cual sólo será visualizado por usuarios que cuenten con el perfil apropiado, al presionar sobre dicho recuadro se accederá al listado de alumnos con cambio de estatus, el listado deberá ser firmado de manera electrónica por el usuario. Una vez realizado el proceso, el buzón mostrado en la parte superior será actualizado.

8.3.7.2 Interfaces Hardware

La interfaz de hardware soportada por la aplicación será cualquier tipo de arquitectura que cuente con la capacidad de ejecutar un Sistema Operativo de 32 ó 64 bits, de igual manera deberá ser compatible con la máquina virtual de java conocida como J2EE. El software es de libre distribución, por lo tanto no es necesario adquirir algún tipo de licencia, además cuenta con soporte para una amplia gama de arquitecturas. Una vez instalada la máquina virtual será necesario instalar el servidor Web, en este caso se empleará Tomcat, el cual no depende de una arquitectura en específico.

La arquitectura encargada del Servidor de Base de Datos será un servidor con un Sistema Operativo de 32 bits, el cual ejecutará el Sistema Operativo conocido como Windows 2008 Server, en el servidor se alojará la base de datos, misma que será gestionada mediante SQL Server 2008. El servidor se encontrará resguardado mediante la configuración del Firewall de Windows, la meta en este caso es restringir el acceso a puertos o servicios.

El modelo elegido para la construcción del sistema permitirá al usuario total independencia de la arquitectura que empleé, bastará que el Sistema Operativo del usuario cuente con conexión a la red interna del CIDE y un navegador Web, este modelo realizará actualizaciones o adecuaciones al sistema sin necesidad de reinstalar el sistema en los clientes.

8.3.7.3 Interfaces Software

La arquitectura de software empleada para el desarrollo del sistema será el Framework llamado Struts en su versión 1.2.x, el motivo principal de su elección es la estructura, bibliotecas y modelo del que hace uso, a través del Framework se permitirá la separación del sistema en componentes como el modelo, la vista y la lógica de negocio, tales componentes interactuarán entre sí, a su vez permitirán la legibilidad del código y una

metodología estándar para el desarrollo del sistema.

Las librerías incluidas dentro del Framework permitirán tanto la optimización del código como la estandarización del mismo, todo ello a través del uso de métodos incluidos en las librerías del Framework, los métodos permitirán la simplificación de actividades de uso común como el manejo de sesiones, gestión de interfaces y conexiones a la base de datos, entre otras.

A continuación se explicará el flujo implementado en un sistema MVC (Modelo Vista Controlador).

- a) El usuario interactúa con la interfaz, la cual es mostrada a través del navegador Web (por ejemplo, pulsa un botón, realiza una selección, ingresa mediante un enlace).
- b) El controlador recibe la notificación por parte de la interfaz de usuario. Dicha notificación es gestionada dependiendo del tipo de evento recibido.
- c) En caso de que el evento recibido así lo requiera, el controlador accederá al modelo y lo actualizará, modificará o simplemente realizará una consulta del mismo.
- d) Al finalizar el acceso al modelo, el controlador delegará al objeto de la vista la tarea de mostrar la interfaz de usuario con el resultado proveniente del controlador.
- e) Al completarse el ciclo la interfaz de usuario regresa a la espera de un nuevo evento por parte del usuario.

La figura 3.5 muestra la manera en que interactúan los diferentes componentes del Modelo Vista Controlador.

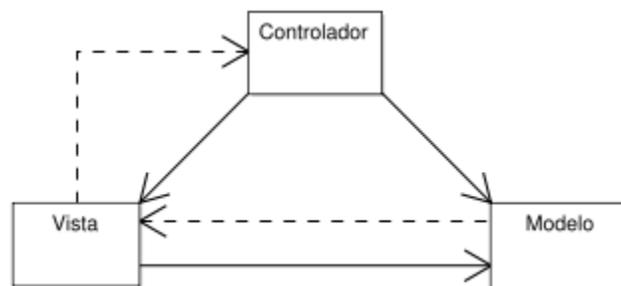


Figura 3.5

Componentes Modelo Vista Controlador

Componentes Modelo Vista Controlador

La aplicación requerirá un servidor Web con soporte para servlets y JSP'S, por ello se ha elegido Tomcat, el cual se integrará con el servidor Web Apache, para gestionar las peticiones que provengan del puerto 80 –comúnmente empleado por la mayoría de servidores Web-, mediante esta integración el servidor Apache gestionará las peticiones dirigidas al servidor Tomcat a través del puerto 80. La integración se realizará mediante la configuración del módulo `apache2_mod_jk`, desarrollado por la Apache Software Foundation.

8.3.7.4 Interfaces de comunicación

El entorno de producción del sistema será exclusivamente la red local del CIDE, constituida por al menos trescientos equipos, en los cuales se emplean diferentes arquitecturas tanto de hardware como software, el común denominador en todos ellos es que cuentan con al menos un navegador Web para establecer comunicación entre el cliente y el servidor por medio el protocolo TCP/IP. El cual permite la transmisión de datos entre redes de computadoras. El protocolo TCP/IP se encuentra formado por un conjunto de protocolos encargados de ofrecer servicios específicos para la transmisión de información en diversos ámbitos, entre ellos podemos mencionar HTTP (HyperText Transfer Protocol) utilizado para acceder a páginas Web, FTP (File Transfer Protocol) para la transferencia de archivos, SMTP (Simple Mail Transfer Protocol) y POP (Post Office Protocol) para el envío de correos electrónicos, sólo por citar los mas comunes.

Dentro de la infraestructura que compone la Intranet se encuentra el servidor principal, encargado de gestionar la conexión entre los equipos del instituto, la conexión se gestionará mediante software protegiendo a los equipos de accesos no autorizados. Las conexiones entre equipos se realizarán mediante *switches*, los cuales se encargan de proveer la conexión al usuario. Esta infraestructura nos permitirá una transferencia de datos de manera eficaz, así como el aseguramiento en el manejo de la información.

8.4 Modelos del sistema

En esta sección se describen los escenarios, casos de uso, modelo objeto y modelos dinámicos que describen el sistema. Se analizará la especificación funcional completa del sistema incluyendo los prototipos de las pantallas del interfaz de usuario y los diagramas que muestren la interacción del usuario con el sistema.

8.5 Escenarios

Acceso al sistema: Se encuentra representado por cualquier usuario que cuente con un nombre de usuario y contraseña válidos, el flujo de este escenario será determinado por los datos ingresados, si ambos datos concuerdan al ser verificados en la base de datos, el acceso al sistema será autorizado, en caso de que el usuario o la contraseña sean erróneos el acceso será denegado.

Captura de requerimientos: Se deberá ingresar la información solicitada, dependiendo del tipo de requerimiento y de los permisos que el usuario posea en relación a su perfil. De acuerdo a la clasificación de perfiles se encuentran dos posibles actores, el primero tiene la facultad de capturar un requerimiento y actualizarlo antes de ser aprobado, el segundo permite realizar las actividades del primero, además de autorizar los requerimientos a favor del área a la que se encuentre adscrito.

Consulta de requerimientos: Permitida a todo usuario con acceso a la captura de requerimientos, de igual manera existen dos actores, el primer actor sólo podrá efectuar consultas de aquellos requerimientos generados por su área, el segundo actor tendrá la capacidad de realizar consultas que impliquen una o más áreas, la definición será gestionada mediante el perfil del usuario.

Captura de adeudos: Los actores son aquellos usuarios adscritos en aquellas áreas en las cuales el alumnado pueda incurrir en una multa, dentro de estas áreas se encuentra la Dirección de Administración Escolar, Idiomas, Biblioteca, etcétera. El usuario deberá registrar un adeudo ingresando el tipo de adeudo, estatus de adeudo y un comentario que indique los motivos del adeudo, la información se ingresará previamente a la búsqueda del alumno. El sistema será capaz de detectar con base al perfil del usuario el área que ha ingresado el adeudo.

Cancelación de adeudos: Intervienen los mismos actores que en la captura de adeudos, una vez que el alumno cumpla con el pago correspondiente al adeudo, el usuario deberá cancelarlo con verificación previa del cumplimiento de su sanción, la cual puede ser de tipo económico. Al cancelar la multa el usuario deberá ingresar un comentario especificando el motivo por el cual se ha cancelado el adeudo.

Consulta buzón de alumnos: En este caso se encontrarán los usuarios que necesiten

conocer el cambio de estatus de aquellos alumnos para los cuales se habilitarán o inhabilitarán los servicios ofrecidos por las diferentes áreas que componen al CIDE. El buzón de alumnos mostrará un listado de alumnos que hayan realizado un cambio de estatus, el listado será firmado de manera electrónica por el usuario.

Cambio / recuperación de contraseña: Se encuentra todo usuario que haya olvidado o perdido su contraseña, la recuperación de la contraseña se realizará a través de la generación de una clave de acceso temporal, la cual se enviará al correo institucional del usuario que solicita el cambio o recuperación de contraseña, una vez que el usuario ingrese haciendo uso de esta llave temporal se le permitirá cambiar de contraseña.

8.6 Modelo de Casos de Uso

8.6.1 Diagramas del Modelo de Casos de Uso. (Diseño del Modelo de Casos de Uso)

Acceso del usuario al sistema

Autores: Encargado de área, empleado.

Descripción: El usuario o encargado de área desean tener acceso a la aplicación para realizar una o diversas tareas definidas dentro del sistema.

Disparador: Dentro del portal de la Intranet del CIDE existirá un enlace que mostrará la interfaz de acceso al sistema, solicitando el nombre de usuario y contraseña para otorgar acceso al sistema.

Precondiciones: Tanto el servidor de la aplicación Web, como el servidor de Bases de Datos iniciarán los servicios para hacerlos accesibles al usuario. El ingreso al sistema estará condicionado, es necesario contar con un usuario y una contraseña válida, misma que será confirmada por el administrador de base de datos.

Post-condiciones: Si el usuario y contraseña ingresados son correctos, el sistema obtendrá el perfil del usuario, con base al perfil el sistema mostrará el menú. En caso de que el usuario o clave sean incorrectos se mostrará un mensaje de error en la pantalla inicial.

Flujo normal:

- ✓ El usuario ingresa a través de la liga mostrada en el portal de la Intranet.
- ✓ Se ingresa el usuario y contraseña proporcionados.

- ✓ Se verifica el usuario y contraseña.
- ✓ Se muestra pantalla con menú principal acorde al perfil del usuario.

Flujo alternativo:

- ✓ El usuario ingresa a través de un marcador que ha creado en el navegador de su preferencia.
- ✓ Ingresa usuario y contraseña.
- ✓ Se verifican tanto el usuario como contraseña proporcionados al sistema.
- ✓ Al ser errónea la contraseña se procede a solicitar la respuesta a la pregunta secreta que ha registrado el usuario.
- ✓ El sistema envía una clave de acceso temporal a la cuenta de correo institucional del usuario.
- ✓ El usuario ingresa con su clave de acceso temporal.
- ✓ Al acceder al sistema con la clave temporal el usuario generará una clave nueva con la cual ingresará de manera habitual.

Excepciones

- ✓ El servidor Web o de Base de Datos no se encuentran funcionando.
- ✓ La cuenta del usuario aún no ha sido activada por el administrador de base de datos.
- ✓ El usuario no recuerda su usuario o contraseña proporcionados por el administrador de base de datos.

Inclusiones

- ✓ Consultar perfil de usuario.
- ✓ Consulta de datos personales del usuario.
- ✓ Consulta de último acceso al sistema.

Extensiones

- ✓ Inicio del tiempo de vida de la sesión
- ✓ Historial de accesos al sistema, en el se almacena la fecha, hora de acceso y la dirección IP de la computadora a través de la cual se ingresó al sistema.

Prioridad: Alta

Frecuencia de uso: 100% de uso

Reglas de negocio: El acceso tanto al sistema como a los módulos debe efectuarse con base a los perfiles de usuario, los cuales han sido previamente definidos, la creación de usuarios nuevos se realizará a partir de petición expresa del titular del área.

Requerimientos especiales: El sistema deberá ser capaz de proveer de manera segura y eficaz un cambio de contraseña, sin implicar una pérdida en cuanto a la seguridad e integridad tanto del sistema como de los usuarios.

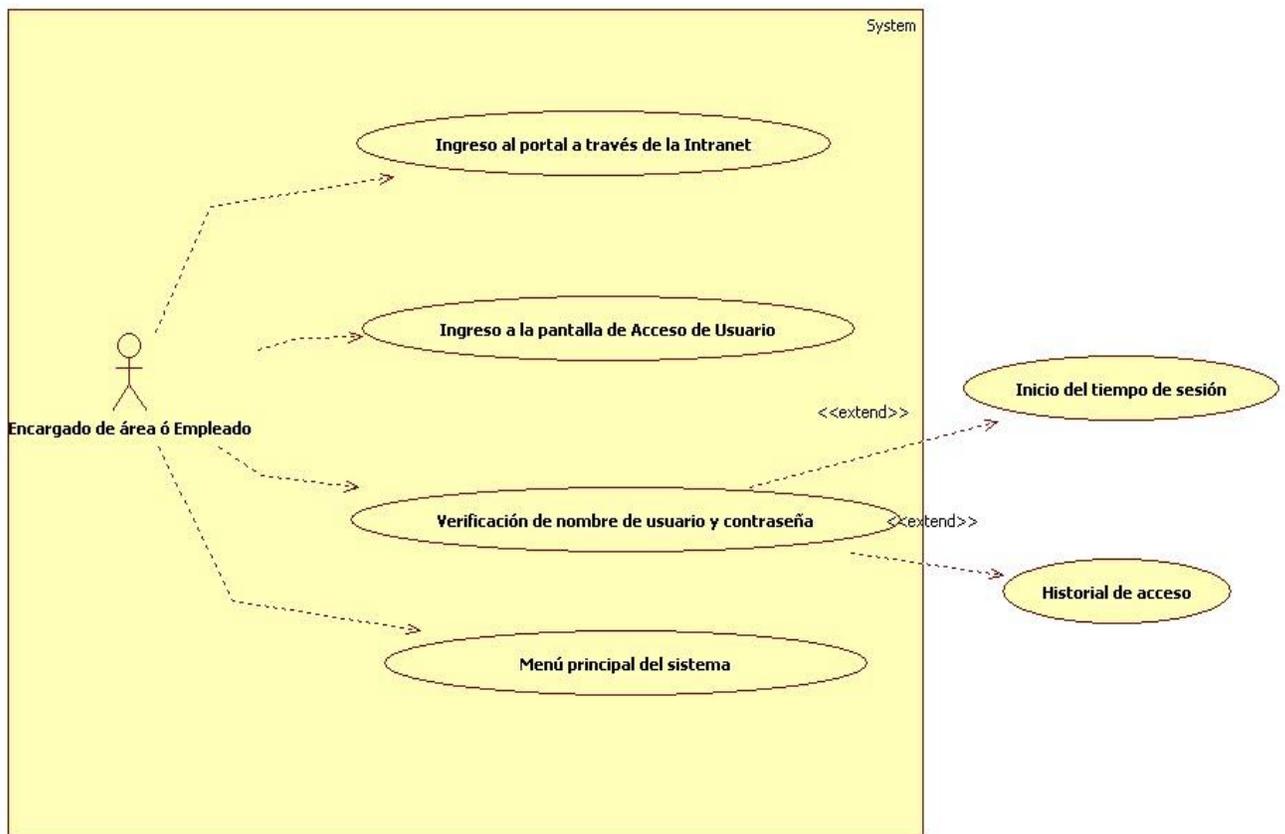


Figura 3.6

Modelo caso de uso acceso al sistema

Captura de requerimiento

Autores: Encargado de área, empleado.

Descripción: El usuario o encargado de área desean realizar la captura de un requerimiento a favor del área en la cual se encuentran adscriptos.

Disparador: El usuario ingresa a la pantalla de captura de requerimientos a través del menú principal.

Precondiciones: El usuario debe de conocer el tipo de requerimiento que desea ingresar, así como otros datos relevantes para efectuar una captura exitosa.

Post-condiciones: Una vez realizada la captura del requerimiento, incluyendo los detalles, se genera un folio oficial, es decir una clave generada a partir de la abreviatura de área y un número consecutivo de requerimientos para esa área dentro del ejercicio fiscal en curso.

Flujo normal:

- ✓ El usuario ingresa a la pantalla de captura de requerimientos a través del menú principal.
- ✓ Selecciona un área, o esta se encuentra preseleccionada en caso de sólo tener acceso a un área.
- ✓ Ingresa mediante una búsqueda la persona que solicita el requerimiento.
- ✓ Selecciona el tipo de requerimiento.
- ✓ Ingresa el concepto general.
- ✓ Ingresa mediante una selección que determina a favor de quién se realizará el requerimiento.
- ✓ Se ingresa la divisa y el tipo de impuesto.
- ✓ Se ingresa tanto la justificación como las observaciones.
- ✓ El ingreso de notas para el requerimiento es opcional.
- ✓ El requerimiento es agregado con un folio temporal, indicando la falta de detalles para el requerimiento.

- ✓ Se agregan los detalles al requerimiento, dichos detalles dependerán del tipo de requerimiento.
- ✓ Una vez agregados los detalles es generado el folio oficial y, se encuentra en espera de ser firmado por el director de área para su aprobación.

Flujo alternativo:

- ✓ El requerimiento se crea a partir de una plantilla la cual contiene los datos de un requerimiento que previamente ha sido establecido como una plantilla.
- ✓ Se actualiza o se agrega información para generar un nuevo requerimiento a partir de una plantilla previamente establecida.
- ✓ Se agregan detalles para completar el requerimiento que generará un folio oficial.
- ✓ Mientras el requerimiento no haya sido aprobado por el responsable de área se permitirá realizar el número de actualizaciones deseadas, ya sea en la pantalla principal del requerimiento o en el detalle.
- ✓ Si el requerimiento fue agregado a través del Sistema de Información Gerencial se permitirá la actualización del mismo en CIDE INTRANET.

Excepciones

- ✓ El usuario no cuenta con los suficientes datos para elaborar el requerimiento.
- ✓ El requerimiento no ha sido oficial pues no se han agregado los detalles del requerimiento.

Inclusiones

- ✓ Consultar número consecutivo temporal de acuerdo al área antes de agregar detalles.
- ✓ Generar folio temporal de acuerdo al tipo de área y número subsecuente.
- ✓ Consultar número consecutivo una vez agregado el requerimiento.
- ✓ Generar folio definitivo una vez agregado el tipo de requerimiento.

Extensiones

- ✓ Registro de requerimiento para ser revisado por parte del encargado de área.

Prioridad: Alta

Frecuencia de uso: 90% de uso

Reglas de negocio: El perfil indicará los tipos de requerimientos, áreas y proveedores accesibles por el usuario. La gestión de permisos se realizará a través de la base de datos mediante la ejecución de un procedimiento almacenado.

Requerimientos especiales: El sistema generará un folio temporal el cual indicará que el requerimiento aún no ha sido completado, por lo que la información capturada será almacenada en una tabla temporal. Una vez completado el requerimiento será generado el folio oficial, el cual es transferido a la tabla de requerimientos; previamente se verificará la secuencia de folios.

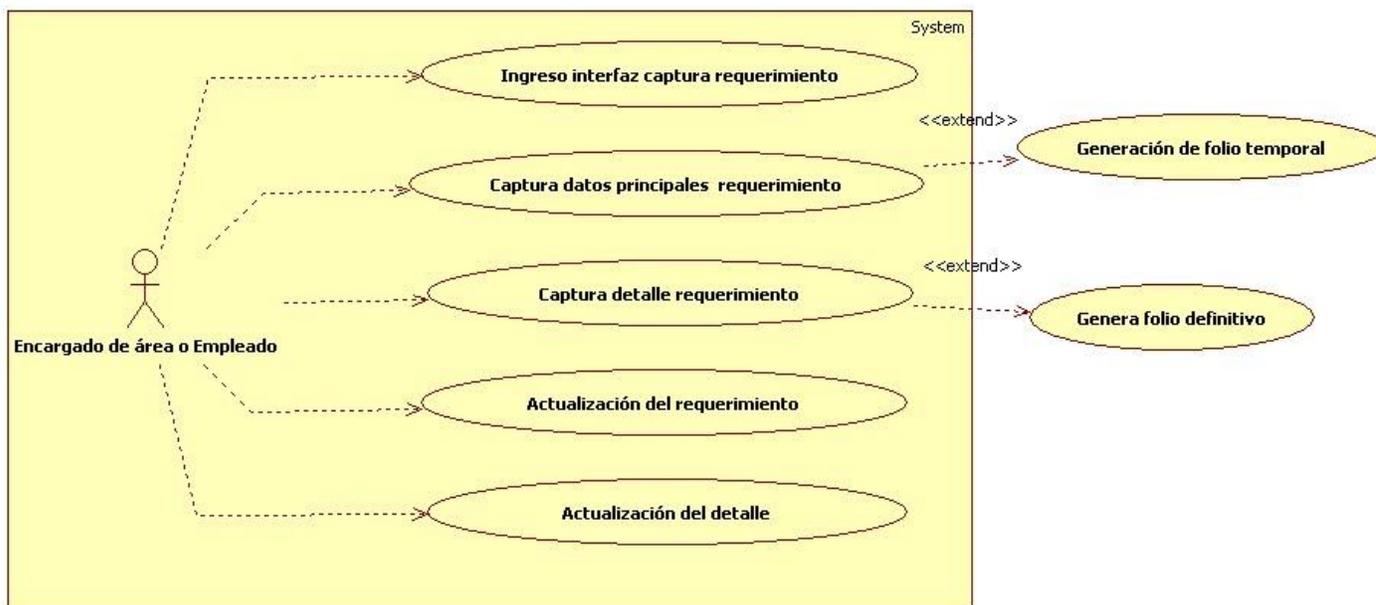


Figura 3.7

Modelo caso de uso captura de requerimientos

Autorización de requerimiento

Autor: Encargado de área.

Descripción: El encargado autorizará o rechazará aquellos requerimientos pertenecientes al área bajo su cargo. El sistema le notificará mediante el menú la cantidad de requerimientos pendientes.

Disparador: El usuario ingresa al sistema y el buzón de requerimientos indica que existen requerimientos pendientes de autorización.

Precondiciones: Se ha capturado de manera íntegra uno o más requerimientos, por lo tanto cuentan con folio oficial.

Post-condiciones: Si el requerimiento es autorizado se canaliza al almacén o al área de presupuesto con la finalidad de surtirlo.

Flujo normal:

- ✓ El usuario mediante el menú principal observa el número de requerimientos por autorizar.
- ✓ El sistema muestra el listado de requerimientos pendientes por autorizar.
- ✓ El usuario revisa la información contenida en el requerimiento.
- ✓ El requerimiento es firmado de manera electrónica.
- ✓ El requerimiento cambia de estatus, por lo tanto la modificación no es permitida una vez que ha sido autorizado.

Flujo alternativo:

Si el requerimiento no cumple con los datos necesarios o no es válido será rechazado.

Si el requerimiento es rechazado el sistema desecha el folio y se continúa con la secuencia establecida de acuerdo al área.

Excepciones

- ✓ El requerimiento ha sido rechazado y se continúa con el número de folio normal.

Inclusiones

- ✓ Una vez autorizado el requerimiento la actualización es bloqueada, por lo que el usuario sólo contará con la opción de consulta.

Extensiones

- ✓ El requerimiento cambia de estatus para su revisión por el almacén o el área de finanzas.

Prioridad: Alta

Frecuencia de uso: 90% de uso

Reglas de negocio: El perfil de usuario determinará su capacidad para autorizar requerimientos, la autorización de un requerimiento se realizará mediante la firma electrónica del usuario, para ello será necesario ingresar el nombre de usuario y contraseña en la parte superior del requerimiento. El requerimiento cambiará de estatus al ser autorizado.

Requerimientos especiales: Un requerimiento autorizado no será susceptible de actualizaciones.

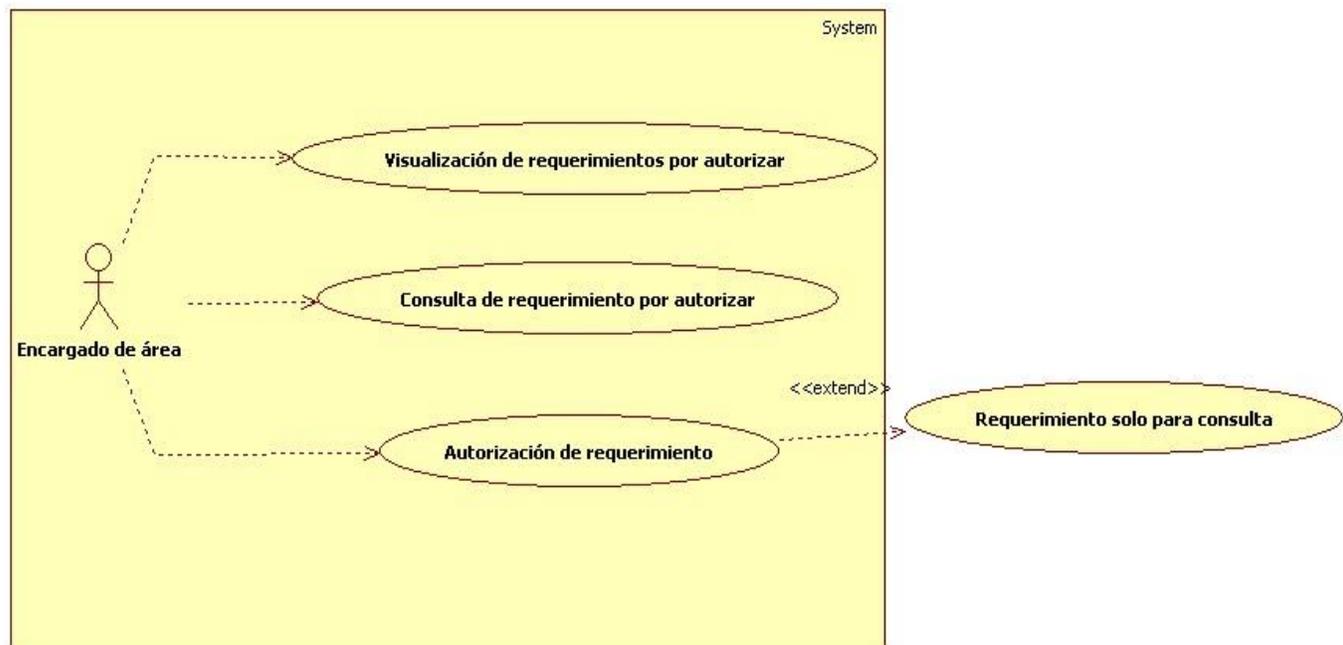


Figura 3.8

Modelo caso de uso autorización de requerimiento

Consulta de requerimiento

Autores: Encargado de área, empleado.

Descripción: Tanto los encargados de área como empleados serán capaces de realizar consultas, las cuales se realizarán con base a los parámetros ingresados y al perfil del usuario. La búsqueda de requerimientos permitirá la impresión del listado obtenido a partir de la búsqueda, a su vez dicho listado se exportará a un archivo con formato PDF.

Disparador: El usuario desea consultar un requerimiento con la finalidad de imprimirlo o modificarlo en caso de que el requerimiento no haya sido autorizado.

Precondiciones: Existen requerimientos capturados a favor del área a la cual se encuentra adscrito el usuario que realiza la búsqueda.

Post-condiciones: El resultado de la búsqueda muestra aquellos requerimientos que coincidan con los parámetros ingresados por el usuario.

Flujo normal:

- ✓ El usuario ingresa a la búsqueda de requerimientos a través del menú principal.
- ✓ El usuario ingresa uno o más parámetros de búsqueda.
- ✓ Se muestran los requerimientos que concuerden con el patrón de búsqueda introducido por el usuario.
- ✓ El usuario selecciona el requerimiento del cual desea conocer los detalles.
- ✓ Los detalles del requerimiento seleccionado por el usuario son mostrados.

Flujo alternativo:

El usuario realiza una búsqueda avanzada, la cual incluye una cantidad mayor de parámetros de búsqueda, entre los cuales se encuentran: proveedor, concepto general, nombre del solicitante, etc.

Excepciones

La búsqueda no produce ningún resultado, debido a que los parámetros ingresados no concuerdan con algún requerimiento almacenado en la base de datos.

Inclusiones

Generación de reportes en formato XLS a partir del listado generado por la búsqueda del usuario.

Prioridad: Media

Frecuencia de uso: 60% de uso

Reglas de negocio: La consulta de requerimientos será restringida dependiendo del perfil del usuario, por lo tanto antes de realizar cualquier búsqueda el sistema consultará la visibilidad de áreas a las cuales tenga acceso el usuario que realiza la búsqueda.

Requerimientos especiales: El listado de resultados deberá ser exportado en formato XLS, con la finalidad de ser impreso o almacenado por el usuario.

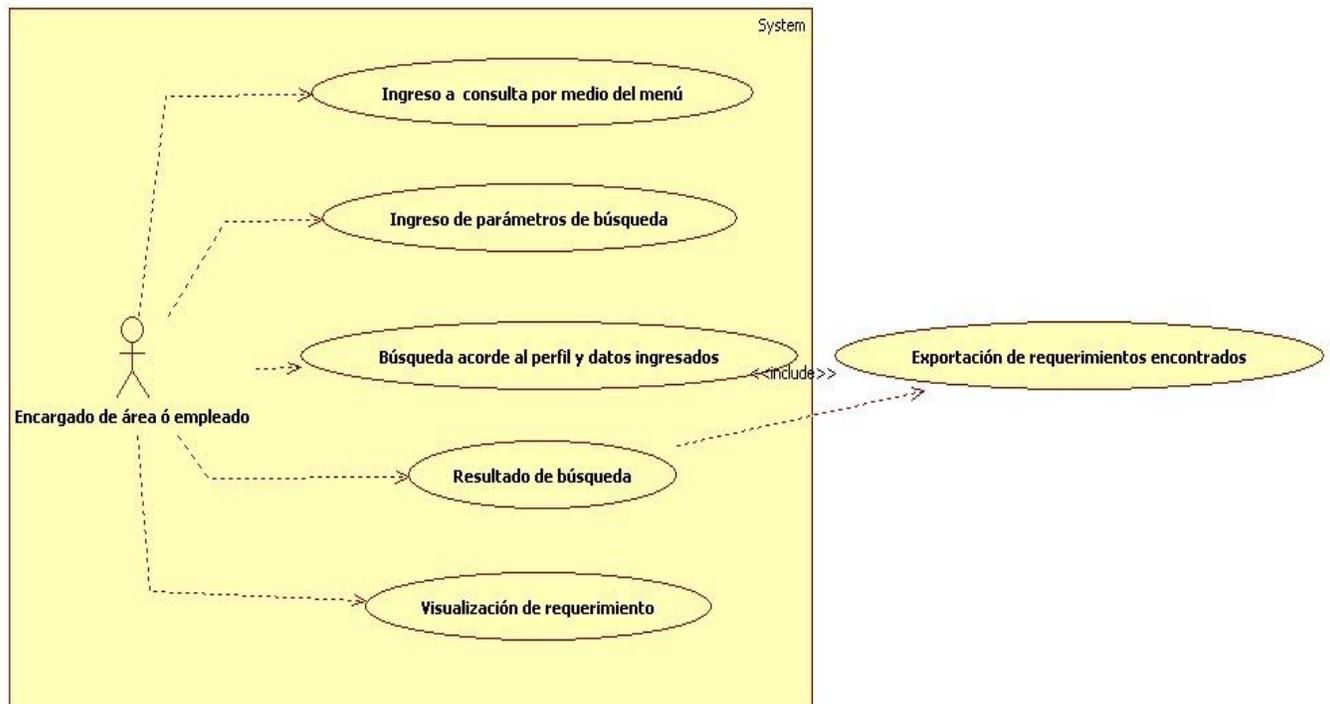


Figura 3.9

Modelo caso de uso consulta de requerimientos

Captura de adeudos

Autor: Encargado de área.

Descripción: El usuario realiza la búsqueda del alumno, una vez encontrado el alumno se ingresa el tipo y descripción del adeudo.

Disparador: El alumno ha incurrido en un adeudo en cualquiera de las áreas del Centro de Investigación y Docencia Económicas.

Precondiciones: El perfil del usuario cuenta con acceso al módulo de captura de adeudos.

Post-condiciones: El adeudo es almacenado en la base de datos.

Flujo normal:

1. El usuario ingresa a la pantalla de adeudos a través del menú principal.
2. Realiza una búsqueda del alumno mediante el uso de diferentes criterios de búsqueda como lo son: nombre, apellido paterno, materno y matrícula.
3. Una vez localizado el alumno se procederá a ingresar el tipo de adeudo generado por el alumno así como una breve descripción.
4. El sistema muestra el listado de adeudos del alumno.

Flujo alternativo:

En caso de que el alumno cuente con un adeudo ya registrado, los siguientes adeudos se agregarán de manera consecutiva.

3.1. Si el perfil del usuario cuenta acceso a más de un área, el sistema le permitirá la capturar y borrado de adeudos en múltiples áreas.

Excepciones

El perfil del usuario no cuenta con los suficientes privilegios para agregar un adeudo.

Inclusiones

Al finalizar la captura del adeudo el sistema mostrará un listado indicando el tipo de adeudo capturado.

Extensiones

La consulta de adeudos mostrará el listado de adeudos generados por uno o más alumnos en una o más áreas.

Prioridad: Alta

Frecuencia de uso: 60% de uso

Reglas de negocio: Sólo usuarios encargados de área contarán con acceso al módulo de adeudos, la cancelación de un adeudo sólo será permitida por el área que haya realizado la captura del mismo.

Requerimientos especiales: Los usuarios con acceso a más de un área tendrán la facultad de capturar y cancelar adeudos en múltiples áreas.

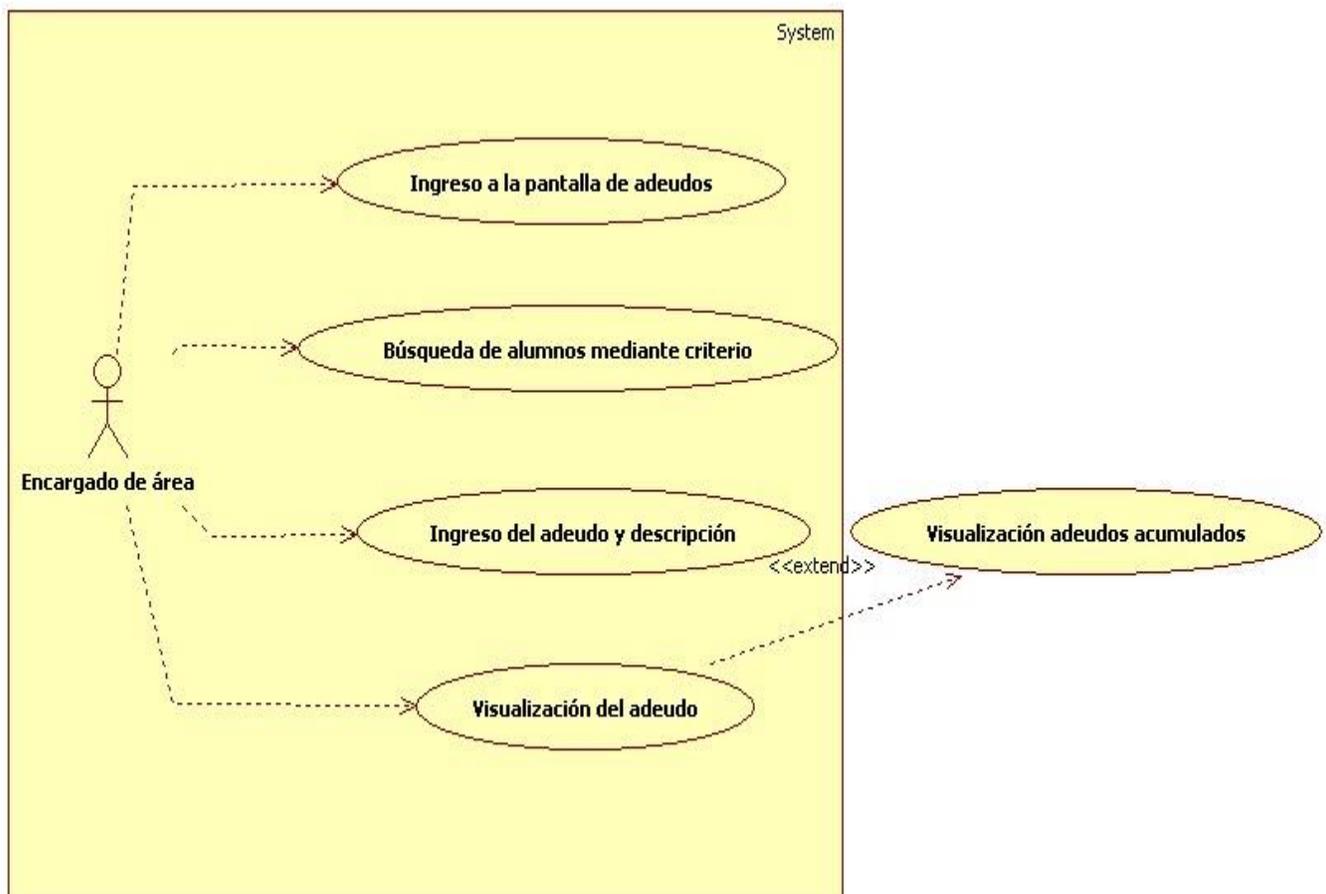


Figura 4.1

Modelo caso de uso captura de adeudos

Buzón de alumnos

Autor: Encargado de área.

Descripción: El buzón de alumnos se encargará de mantener informados a los usuarios de los alumnos con un cambio de estatus, el conocimiento del cambio de estatus permitirá habilitar/inhabilitar los servicios otorgados a los alumnos del instituto.

Disparador: El usuario desea conocer los alumnos que han registrado un cambio de estatus.

Precondiciones: La Dirección de Administración Escolar se encargará de actualizar la información de los alumnos con cambio de estatus.

Post-condiciones: Una vez registrada la firma electrónica del usuario, el buzón de alumnos reiniciará el contador ubicado en la parte superior del menú.

Flujo normal:

1. El usuario ingresa al sistema.
2. El menú principal muestra el contador que indica la cantidad de alumnos que han registrado un cambio de estatus.
3. El usuario accede al listado de alumnos con cambio de estatus mediante la parte superior del menú, en donde es mostrada la información detallada del alumno.
4. El usuario firma de manera electrónica el listado de alumnos con cambio de estatus.
5. El contador ubicado en el menú es reiniciado hasta existir algún cambio de estatus.

Flujo alternativo:

- 3.1. El usuario accede al listado de alumnos mediante el apartado de alumnos ubicado en la parte inferior del menú.

Excepciones

- ✓ Si el buzón de alumnos no es mostrado en la parte superior es debido a que el usuario no cuenta con los suficientes permisos para visualizarlo.

Inclusiones

- ✓ El contador mostrado en el buzón de alumnos se actualizará cada vez que el usuario

revise el listado en donde aparecerá el cambio de estatus.

Extensiones

- ✓ Una vez firmado el cambio es generada una bitácora indicando: el nombre de usuario, fecha y hora en la que es registrada la firma electrónica.

Prioridad: Alta

Frecuencia de uso: 40% de uso

Reglas de negocio: El sistema mostrará el compendio de alumnos con cambio de estatus de manera individual, la actualización del buzón dependerá de la periodicidad con que sea revisado por el usuario. La bitácora indicará la fecha y hora en la cual el usuario registra su firma electrónica.

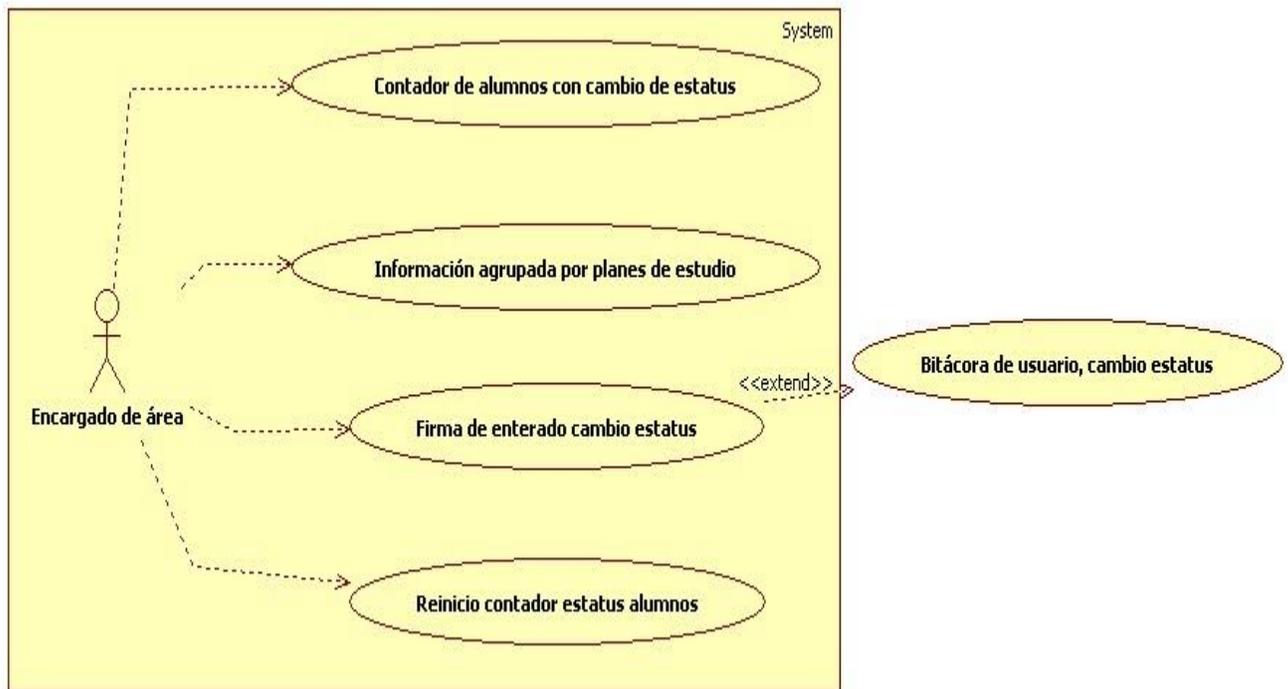


Figura 4.2

Modelo caso de uso buzón de alumnos

CIDE INTRANET
Arquitectura de Software

9. Introducción a la arquitectura de software

El presente capítulo describe la especificación completa de la Arquitectura de Software utilizada en la aplicación CIDE INTRANET. Además, se define el conjunto de especificaciones estructurales y aspectos arquitectónicos acordes al Análisis de Requerimientos.

9.1 Alcance

El objetivo de la arquitectura de software es proporcionar una visión global que permita establecer un medio de comunicación entre el arquitecto de software y otros miembros del equipo de desarrollo, abarcando un conjunto de decisiones importantes sobre la organización y metodologías empleadas en el desarrollo de CIDE INTRANET, incluyendo:

- ✓ Definición del lenguaje de desarrollo.
- ✓ Definición de elementos estructurales e interfaces.
- ✓ Definición del framework.
- ✓ Facilidad de uso.
- ✓ Flexibilidad
- ✓ Rendimiento

10. Resumen Arquitectónico

10.1 Hechos más Importantes

El conocimiento y dominio de la arquitectura de software resulta ser un factor primordial en el desarrollo de todo sistema, a continuación se realizará una pequeña introducción en la cual se definirá de manera más precisa los objetivos y alcances de la Arquitectura de software y los elementos que la componen.

En sentido amplio, el término Arquitectura de Software se puede definir como el diseño de más alto nivel en la estructura de un sistema, programa o aplicación; esta arquitectura tiene como finalidad realizar un seguimiento adecuado de un conjunto de patrones y abstracciones coherentes que proporcionen el marco de referencia necesario para guiar la construcción de cualquiera de los sistemas de información mencionados con anterioridad.

Los patrones y abstracciones definidos por la arquitectura de software incluyen los componentes que realicen toda interacción del usuario con el sistema, sus interfaces y la comunicación entre ellas, así

mismo, se define la implementación de tipo física, en la cual se especificará la distribución y funcionalidad de cada equipo implicado en el sistema.

La arquitectura de software establece los fundamentos seguidos por el equipo de desarrollo, los cuales tienen como meta seguir una línea común que permita alcanzar los objetivos establecidos por la visión del sistema. Además de cumplir con los objetivos de tipo funcional es necesario llevar a cabo una serie de objetivos generales como: mantenibilidad, auditabilidad, flexibilidad e interacción con otros sistemas de información.

La selección de la Arquitectura de Software es realizada con base en los objetivos y restricciones, ambas condiciones varían de acuerdo al sistema analizado y, principalmente refieren a aquellas limitaciones que deriven directamente de las condiciones y tecnologías que determinen la viabilidad y éxito de la arquitectura elegida.

A continuación se enlistan las responsabilidades establecidas dentro de la Arquitectura de Software:

- ✓ Ubicación del hardware.
- ✓ Protocolos de interacción y comunicación.
- ✓ Control y flujo de datos.
- ✓ Definición de los módulos principales.
- ✓ Definición de la funcionalidad de los módulos principales.
- ✓ Definición de la interacción entre módulos.

El objetivo principal de la Arquitectura de Software radica en la importancia de establecer un medio común que permita la comunicación entre los integrantes del equipo, así mismo, se establecen los elementos que faciliten la construcción de abstracciones, las cuales se pueden definir como la representación gráfica de una parte de funcionalidad fraccionada del sistema.

Estas abstracciones son representadas mediante diagramas de flujo, estado, etcétera. La elaboración es realizada mediante un lenguaje de modelado unificado (**UML**, por sus siglas en inglés, *Unified Modeling Language*) considerado como un estándar para la visualización, especificación, construcción y documentación en sistemas de información.

10.2 Estilo Arquitectónico

De acuerdo a las condiciones e infraestructura del Centro de Investigación y Docencia Económicas, el proyecto será desarrollado en un entorno de red de área local que proporciona los siguientes beneficios:

- ✓ Acceso controlado
- ✓ Transferencia segura e íntegra de la información
- ✓ Se evita la redundancia de la información

La red de área local permitirá el desarrollo de la aplicación haciendo uso de la arquitectura cliente/servidor, en la cual el cliente (usuario que desea acceder a un recurso) realiza peticiones a un servidor (equipo encargado de ofrecer recursos informáticos).

La arquitectura cliente/servidor permite la distribución de procesos entre el cliente y servidor, así como la gestión y centralización de la información; esta separación es realizada a nivel lógico, logrando la comunicación de los equipos a través de los protocolos de comunicación TCP/IP.

En la figura 5.1 se muestra el funcionamiento de la arquitectura cliente/servidor, en la que el cliente inicia el diálogo enviando una petición a través del protocolo HTTP, la petición es recibida por el servidor, es procesada y envía la respuesta al cliente.

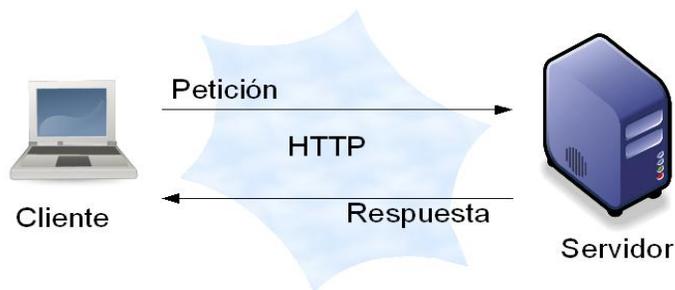


Figura 5.1

Diagrama cliente/servidor

El modelo cliente/servidor permite la distribución de procesamiento en dos componentes independientes que cooperan entre sí, una de las debilidades en el modelo radica en la falta de asignación clara de las funciones de cada componente, lo cual implica un mayor tiempo de procesamiento debido a la gestión de recursos. tanto para el servicio de aplicaciones como para el sistema de gestión de bases de datos, por lo tanto y de acuerdo al modelo de datos se ha optado por un modelo descentralizado que retoma los beneficios de la arquitectura cliente/servidor, además de permitir la adecuada gestión de recursos con la finalidad de obtener un menor tiempo de procesamiento.

La arquitectura de tres niveles es una especialización de la arquitectura cliente/servidor que conjuga sus virtudes y añade un tercer nivel, a través del cual establece una separación expresa entre los componentes; cada componente se especializa en una determinada función, permitiendo el diseño de aplicaciones escalables, esto se traduce en la facilidad en que pueden ampliarse en caso de así ser requerido.

10.3 Objetivos de la Arquitectura

La arquitectura de tres niveles proporciona los siguientes beneficios:

- ✓ Mayor grado de flexibilidad.
- ✓ Mayor seguridad. (Independiente en cada nivel)
- ✓ Mejor rendimiento. (El sistema es distribuido logrando independencia en cada proceso)

11. Componentes Significativos de la Arquitectura del Sistema

A continuación se enlistan y describen las capas que conforman la arquitectura de tres niveles, así como la funcionalidad implicada en cada una de ellas.

Cliente: Es el equipo que solicita los recursos mediante una interfaz de usuario, se encarga de comunicar y capturar la información que será enviada al servidor de aplicaciones.

Servidor de aplicaciones: Comúnmente llamado software intermedio, entre sus funciones se encuentran la recepción y respuesta de las solicitudes realizadas por el cliente, así como la interacción con el servidor de bases de datos para almacenar o consultar datos.

Servidor de datos: Encargado de almacenar y proporcionar los datos requeridos por el servidor de aplicaciones. Puede estar formado por uno o más Sistemas de Gestión de Bases de Datos (SGBD).

En la figura 5.2 se muestra el diagrama que ilustra la arquitectura de tres capas y tres niveles, se hace referencia al número de niveles por la separación física existente en cada una de las capas, esto significa que existe físicamente un servidor para cada una de las capas de aplicación.

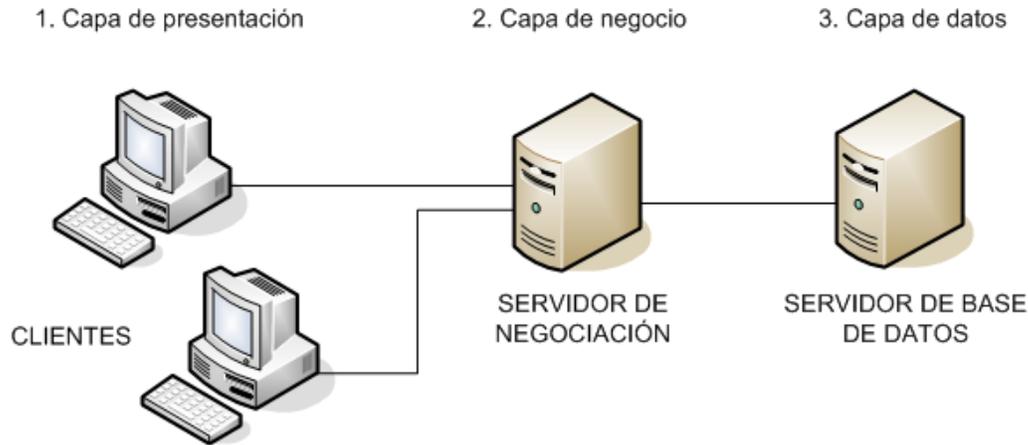


Figura 5.2

Arquitectura de tres capas establecida en tres niveles

Un *framework* es la representación de una arquitectura de software, se encuentra formado por una estructura definida. Se incluye un conjunto bibliotecas que permiten el desarrollo de software de manera organizada mediante un conjunto de metodologías y procedimientos definidos.

El propósito de todo *framework* radica en la capacidad de facilitar y estandarizar el desarrollo de software, esto permite un mayor tiempo para el análisis de requerimientos y un menor tiempo en la implementación de las características de bajo nivel (conexiones a bases de datos, estructura jerárquica y archivos de configuración).

Cabe mencionar que un *framework* no necesariamente se encuentra ligado a un lenguaje de programación específico, por el contrario, los *frameworks* son orientados al tipo de aplicación que se desea desarrollar (aplicaciones Web, aplicaciones de escritorio, etcétera).

A continuación se presentan de manera concreta las ventajas obtenidas a partir de la implementación de un *framework*:

- ✓ Definición concreta de la estructura global y metodologías de desarrollo.
- ✓ Facilidad en integración y colaboración de grupos de desarrollo.
- ✓ Implementación de utilerías, librerías y herramientas de terceros.
- ✓ Permite la escalación del sistema.

11.1 Lenguaje y framework de desarrollo

Una vez presentados los antecedentes que detallan las ventajas de la elección correcta de aspectos como: el tipo de arquitectura de software, *framework* o marco de trabajo adecuado a las metodologías establecidas por el tipo de arquitectura elegida, es imprescindible elegir de forma correcta un lenguaje de programación capaz de seguir la metodología establecida por uno o más *frameworks*.

Después de un estudio detallado en el que se encuentran implicados los factores mencionados se concluye que *Java Enterprise Edition* es el lenguaje de programación idóneo para realizar el proyecto. A continuación se expone un breve análisis de sus características y ventajas sobre otros lenguajes de programación.

Java es un lenguaje orientado a objetos, esto indica que se ajusta al paradigma de programación conocido como Programación Orientada a Objetos (**POO** u **OOP** según sus siglas en inglés) y consiste en la **usabilidad** de objetos y su interacción entre ellos, esto permite el uso de ciertas características que facilitan el desarrollo de sistemas, entre ellas se encuentran:

Abstracción: Característica que permite definir y modelar un objeto abstracto, mismo que se encarga de efectuar una serie de acciones como el cambio de estado, comunicación con otros objetos, etcétera.

Encapsulación: Tiene como objetivo mantener los atributos de los objetos pertenecientes a una misma entidad como privados y proporcionar el acceso a los mismos a través de métodos públicos, esto permite la protección de datos, facilidad y la flexibilidad en el mantenimiento de las aplicaciones.

Herencia: Es la capacidad de crear clases que adquieren de manera automática los atributos y métodos de otra clase previamente definida, además, permite construir una clase nueva, en la que se pueden agregar métodos y atributos propios.

Poliformismo: Se define como la posibilidad de utilizar una misma expresión para invocar a diferentes versiones de un mismo método; permite un comportamiento diferente a pesar de compartir el mismo nombre.

Modularidad: Establece la división de una aplicación en componentes diferentes (clases, paquetes o bibliotecas) logrando una disminución en el grado de dificultad del problema y permitiendo una mayor comprensión.

La POO considera a los objetos como entidades independientes con la capacidad de combinar atributos como: estado, comportamiento e identidad.

El estado se refiere a la composición de datos mediante uno o más atributos a los que se asignan valores concretos.

El comportamiento es definido por los métodos y procedimientos que pueden operar sobre dicho objeto, esto quiere decir que cada objeto cuenta con una serie de acciones a través de las cuales puede ser manipulado.

La identidad es la propiedad que identifica y diferencia cada objeto de manera análoga, es considerado como el identificador de una variable.

La base de la programación orientada a objetos radica en el conjunto de reglas y metodologías que rigen la manera en que son tratados los objetos y su colaboración entre ellos para realizar procesos conjuntos, a continuación se describen de manera breve algunos de los conceptos de mayor uso en los lenguajes orientados a objetos.

Clase: Archivo en donde se define el comportamiento de un objeto y las propiedades que lo componen, a través de la clase se permite la instanciación, esto hace referencia a la lectura y creación de objetos en clases externas.

Objeto: Entidad conformada por un conjunto de atributos acompañados de un comportamiento específico; responde a una serie de eventos. La manipulación de un objeto es realizada por la instancia de una clase, utiliza una variable para acceder a los métodos a través del operador, esto permite hacer la llamada a los métodos del objeto.

Método: Conjunto de instrucciones agrupadas en una clase, encargadas de la definición y comportamiento de los objetos pertenecientes a una clase, los métodos pueden acceder a campos o atributos propios de clase con la finalidad de almacenar información sobre el objeto.

Evento: Suceso originado a partir de la interacción del usuario con el sistema, mismo que responde con la ejecución de una rutina vinculada al tipo de evento. La rutina será ejecutada cada vez que se origine el evento en el sistema.

Mensaje: Comunicación dirigida hacia un objeto, tiene como finalidad la ejecución de un método y puede ir acompañado de cero o más parámetros de ejecución.

Atributo o propiedad: Contenedor de alguno de los diferentes tipos de datos válidos para una variable. El conjunto de atributos define las características de un método.

Java es desarrollado por la empresa *Sun Microsystems* y, además de brindar todas las ventajas de la POO cuenta con una serie de características que lo hacen ideal para el desarrollo de una arquitectura en capas con total independencia de la plataforma en la que se desarrolle o ejecute el sistema, esto es posible debido a la implementación de la máquina virtual de Java (**JVM**, *Java Virtual Machine*, por sus siglas en inglés).

La máquina virtual de Java se encarga de compilar el código fuente escrito en lenguaje Java. A diferencia de un compilador, el cual genera un código capaz de ser interpretado directamente por una computadora, la *JVM* compila el código y produce lo que se conoce como un código intermedio denominado “*bytecode*,” que sólo es una simplificación de instrucciones máquina específicas de la plataforma Java.

El *bytecode* es una pieza de código independiente de la arquitectura en que fue generado, por lo tanto puede ser interpretado por cualquier máquina virtual (JVM), esta independencia restringe la ejecución directa de *bytecodes* por cualquier sistema operativo. La fase de ejecución comprende la interpretación de los archivos *.class* (archivos producto de la compilación) y consiste en la traducción de *bytecodes* a código ejecutable por el Sistema Operativo.

Cada Sistema Operativo proporciona su propia implementación de la JVM, la cual ejecuta de manera similar la interpretación del *bytecode*, sin embargo, cada una realiza la interpretación de acuerdo a las características del sistema operativo en el que se ejecuta la aplicación.

En la figura 5.3 se muestra un esquema en donde se ilustra el proceso de compilación y ejecución de aplicaciones.

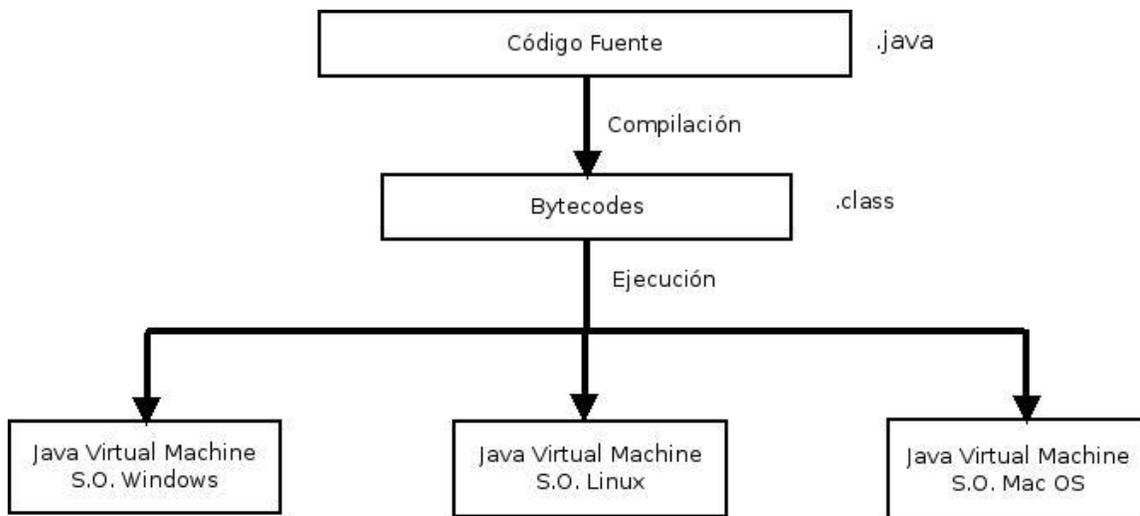


Figura 5.3

Compilación y ejecución de Java en diferentes plataformas.

Ediciones Java

Una de las principales características de Java es el hecho de disponer de una amplia variedad de librerías (conjunto de clases) para la realización de aplicaciones de acuerdo al tipo de aplicación a desarrollar. Este compendio de clases se encuentra organizado en tres grupos, conocidos como ediciones Java.

Java Platform, Standar Edition (JSE). En este grupo se encuentran las clases de uso general (tratamiento de cadenas, funciones matemáticas, colecciones, acceso a datos, etcétera), es decir, aquellas librerías que se utilizan en cualquier tipo de aplicación. Asimismo se incluyen los paquetes para la creación de entornos gráficos y aplicaciones ejecutadas por navegadores Web (applets).

Java Micro Edition (JME). Proporciona la especificación de un subconjunto de paquetes y tecnologías orientadas a la creación de aplicaciones en dispositivos de capacidades limitadas (teléfonos celulares, agendas electrónicas, etcétera).

Java Platform, Enterprise Edition (JEE). Conjunto de paquetes y tecnologías para el desarrollo y ejecución de software de aplicaciones en arquitecturas distribuidas de N niveles, la plataforma JEE tiene como base el uso de componentes modulares, mismos que son ejecutados en un servidor de aplicaciones.

Una vez analizadas las diferentes versiones de Java se concluye que *Java Enterprise Edition* es la edición adecuada para realizar el proyecto debido a su capacidad de ejecución en múltiples plataformas, así como permitir el desarrollo de aplicaciones siguiendo una arquitectura distribuida, una cantidad significativa de *frameworks* apegados al desarrollo de aplicaciones multicapa y una amplia variedad de servidores web.

Framework

Después de un análisis en donde se evaluaron aspectos como el tipo de arquitectura implementada, conjunto de bibliotecas disponibles, soporte e integración con otros sistemas se eligió para el desarrollo de CIDE INTRANET el *framework Struts*.

Apache Struts es un framework de aplicaciones Web de libre distribución; permite el desarrollo de aplicaciones bajo la plataforma JEE mediante el patrón de diseño Modelo Vista Controlador (MVC), considerado como una arquitectura de tres capas y, puede desarrollarse en un entorno de dos o más niveles.

El modelo es representado tanto por el sistema de gestión de bases de datos como por la información con la que opera el sistema, la vista es considerada como la interfaz de usuario y el controlador refiere a la navegación y flujo del sistema.

En la figura 5.4 se muestran las relaciones entre las capas modelo, vista y controlador de la arquitectura MVC.

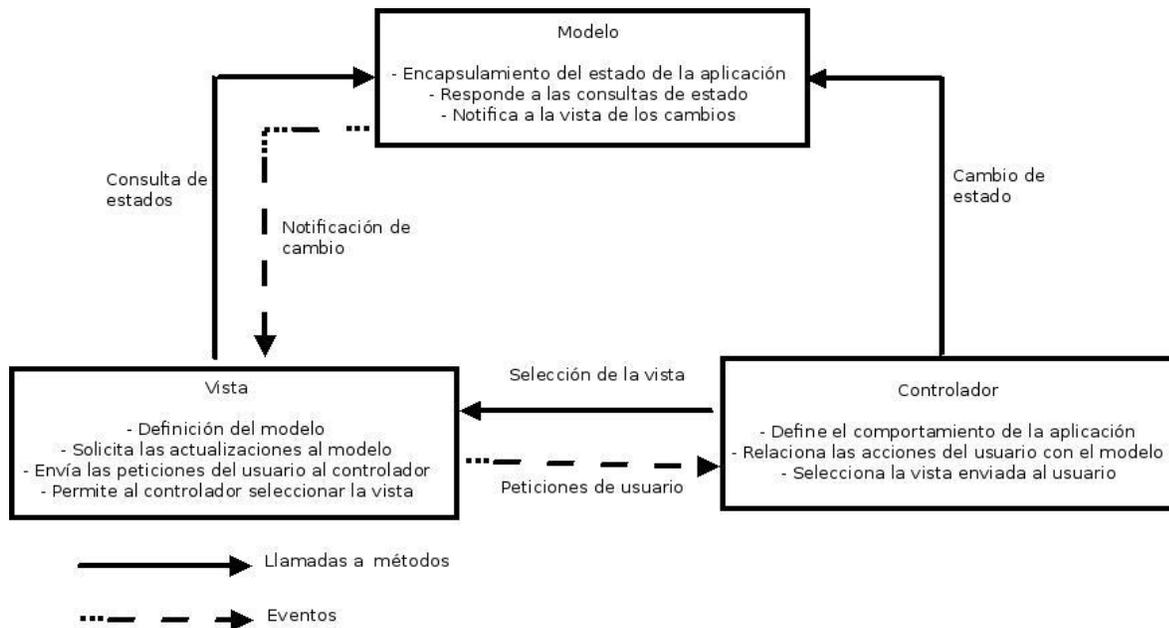


Figura 5.4

Relaciones entre capas del MVC

Struts proporciona tres componentes claves para el desarrollo de sistemas:

Request (Petición): Conductor encargado del mapeo de acuerdo a la norma URI (Uniform Resource Identifier, Identificador Unificado de Recurso). Consiste en una cadena corta de caracteres que identifica un recurso.

Response (Respuesta): Conductor encargado de transferir el control a otro recurso capaz de completar la petición.

Tag library (Conjunto de marcadores): Permiten la creación interactiva de formularios para aplicaciones Web.

A continuación se detalla el funcionamiento de la arquitectura MVC implementada en el *framework Struts*

Funcionamiento del controlador de Struts

Cuando un usuario realiza una petición es manejada por la clase *Action Servlet*, interceptando la solicitud por medio de la URL que se encuentra asociada con una acción mediante un archivo de configuración en formato XML. La clase *Action Servlet* se encarga de establecer la comunicación con la capa de abstracción de datos (modelo).

Funcionamiento de la vista en Struts

La vista es la capa encargada de presentar la información de acuerdo a la solicitud ingresada por el usuario. La información proviene del modelo y es mostrada por archivos conocidos como Java Server Pages (JSP) que son complementados con el uso del lenguaje Java Script.

Funcionamiento del modelo en Struts

El modelo se encarga de proporcionar las interfaces a las bases de datos, no tiene un formato definido, es por ello que es posible la reutilización del código.

En la figura 5.5 se muestra el diagrama de flujo de una aplicación realizada haciendo uso del Framework Struts.

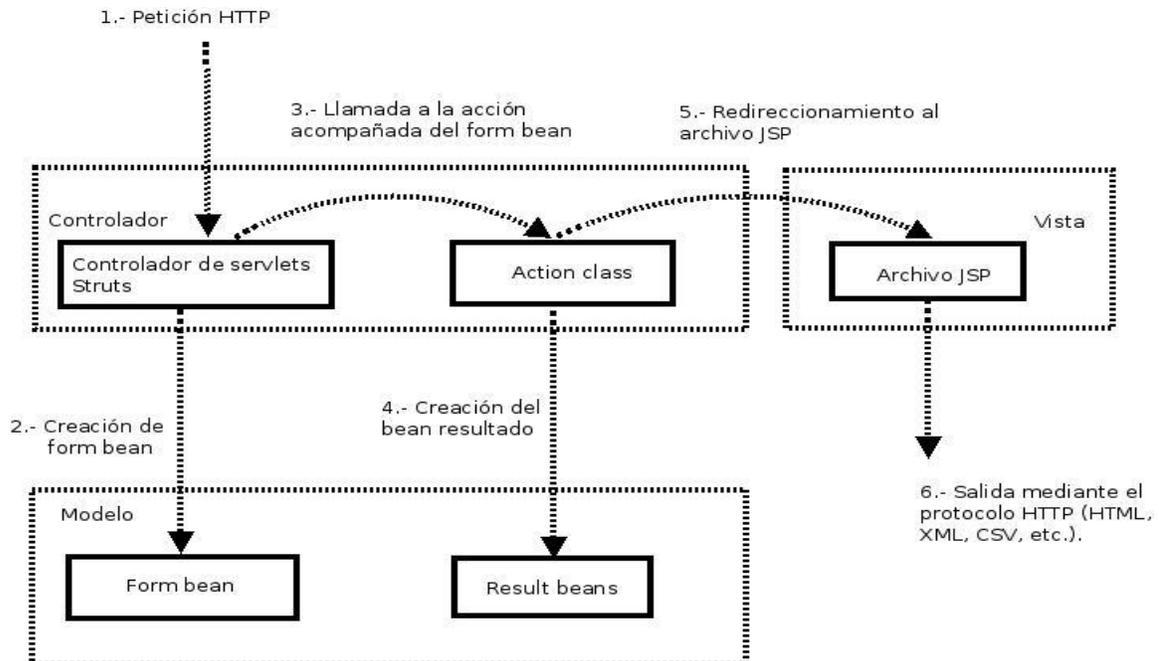


Figura 2.3

Diagrama de flujo de una aplicación realizada con Struts Framework

11.2 Servidor Web

Un servidor Web es un programa diseñado para realizar transferencia de datos a través del protocolo HTTP (*HyperText Transfer Protocol*, protocolo de transferencia de hipertexto, por sus siglas en inglés). El servidor se mantiene a la espera de peticiones por parte de un cliente (navegador Web), una vez recibida la petición es procesada por el servidor Web, enviando la respuesta mediante una página Web, la cual será mostrada en el navegador del usuario que realizó la petición.

Debido a sus características e integración con el *framework Struts* se llevó a cabo la elección del servidor Web *Tomcat* que es distribuido y desarrollado por la Apache Software Foundation.

El servidor Web Tomcat cumple con la especificación para la ejecución tanto de servlets¹ como de JavaServer Pages (JSP).

11.3 Servidor de base de datos

El servidor de base de datos es considerado como un elemento de vital importancia en la arquitectura de tres niveles, su función consiste en almacenar de manera sistemática un conjunto de datos pertenecientes a un mismo contexto.

El Sistema de Gestión de Base de Datos (SGBD) tiene como propósito realizar un manejo claro, sencillo y ordenado del conjunto de datos almacenados, en este caso se utilizará el SGBD *SQL SERVER 2008*, implementado en una plataforma Windows Server 2008; esta separación permitirá el cumplimiento de la arquitectura de tres capas establecida en tres niveles, obteniendo un mejor rendimiento y mayor seguridad de la información.

El desarrollo de la aplicación no interferirá con el actual sistema gestor de base de datos, CIDE INTRANET se ajustará al modelo actual, por lo tanto se permitirá el acceso y actualización a registros realizados con anterioridad.

12. Vista de Casos de Uso

Captura de requerimientos: El usuario ingresa al sistema, selecciona la opción captura de requerimientos en el menú principal, selecciona el tipo de requerimiento y, el sistema muestra los campos acordes al requerimiento seleccionado. Una vez que el sistema valida los campos, el requerimiento es almacenado con un folio temporal. El usuario ingresa el detalle del requerimiento

¹ Un servlet es un programa escrito en lenguaje Java capaz de ofrecer contenido dinámico dependiendo del tipo de petición realizada por uno o más clientes, en tanto que un JSP es una página Web construida a base de etiquetas y código Java; es interpretada con la finalidad de construir un servlet que producirá un documento con formato HTML como respuesta a la petición del cliente.

actualizando el folio temporal a un folio definitivo.

Consulta de requerimientos: El usuario ingresa mediante la opción de consulta en el menú de requerimientos, ingresa alguno de los parámetros de búsqueda y, el sistema muestra los resultados que coincidan con los parámetros ingresados y perfil de usuario.

Actualización de requerimientos: El usuario ingresa mediante la consulta de requerimientos o al momento de generar un folio oficial. El usuario actualiza uno o más campos pertenecientes al requerimiento o detalle. El sistema permitirá realizar un número indeterminado de actualizaciones mientras el requerimiento no sea autorizado por el titular de área.

Autorización de requerimientos: El usuario ingresa a través de una búsqueda o el menú de requerimientos pendientes por autorizar. El usuario firma de manera electrónica ingresando su nombre de usuario y contraseña en la parte superior del requerimiento que será autorizado.

La figura 6.1 muestra el modelado del caso de uso correspondiente al módulo de requerimientos.

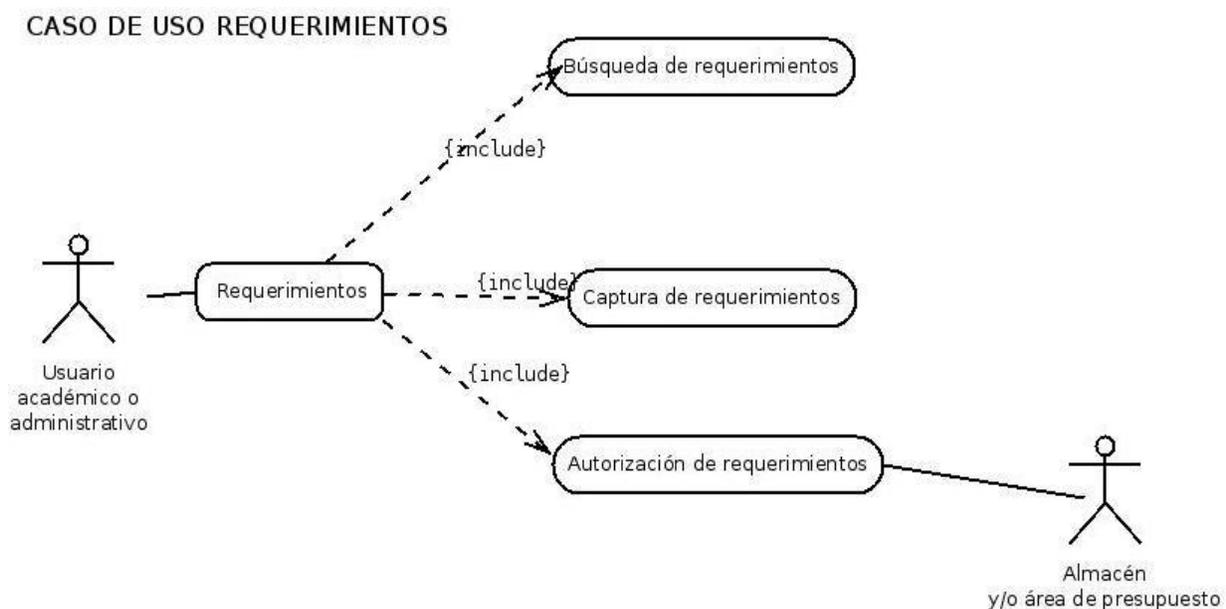


Figura 6.1

Modelado caso de uso requerimientos

Consulta de adeudos: El usuario ingresa al sistema, selecciona la opción consulta de alumnos e ingresa los parámetros de búsqueda deseados. El sistema busca las coincidencias de acuerdo a los parámetros ingresados, en caso de encontrar una o más coincidencias el sistema mostrará los adeudos, de lo contrario emitirá un mensaje indicando que el alumno no tiene adeudos.

Captura de adeudos: El usuario realiza la búsqueda de alumnos, selecciona el tipo de adeudo,

comentario y en caso de ser requerido el monto del adeudo el sistema registra el adeudo del alumno.

Modificación de adeudos: El usuario realiza la búsqueda de alumnos, el sistema muestra los resultados que coincidan con los parámetros de búsqueda. El usuario modifica el adeudo actualizándolo o cancelándolo, en caso de ser cancelado el sistema registra la fecha, usuario y motivo de cancelación del adeudo.

La figura 6.2 muestra el modelado del caso de uso correspondiente al módulo de adeudos.

CASOS DE USO ADEUDOS

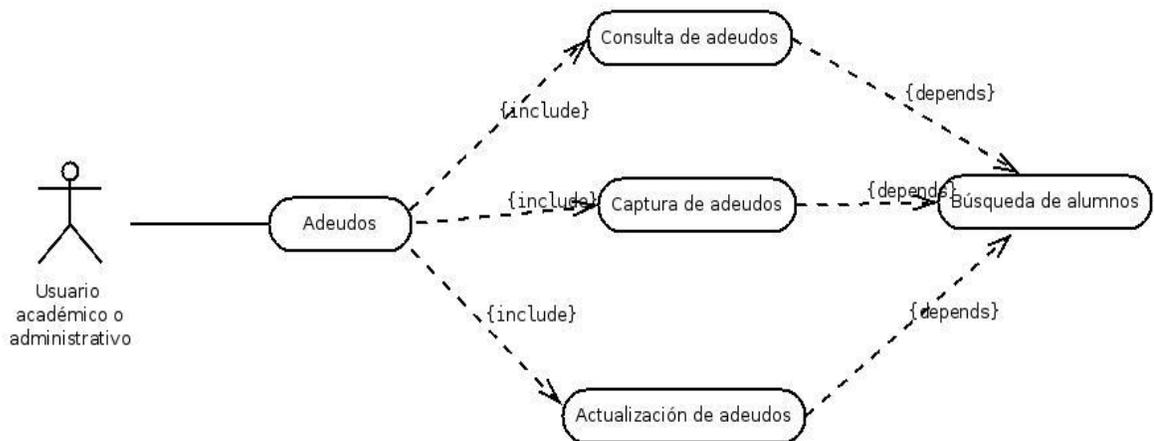


Figura 6.2

Modelado caso de uso adeudos

Buzón de alumnos: El usuario ingresa al sistema, el sistema muestra el número de alumnos con cambio de estatus agrupados por niveles (licenciatura, maestría, doctorado). El usuario visualiza el nombre del alumno, plan de estudios y motivo del cambio de estatus; el sistema registra la firma electrónica del usuario.

La figura 6.3 representa el modelado del buzón de alumnos.

CASO DE USO BUZÓN DE ALUMNOS

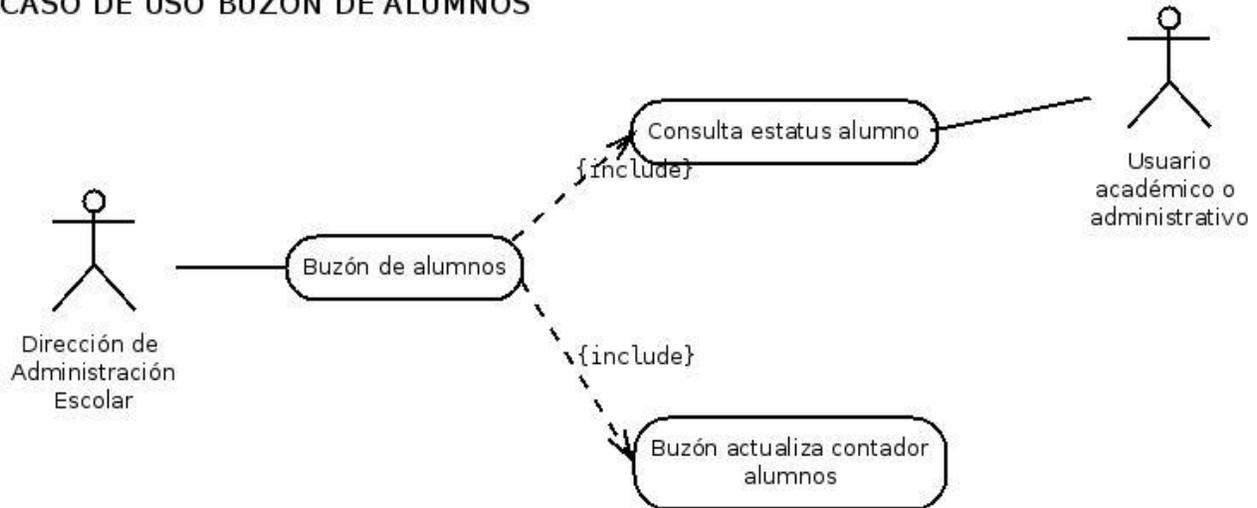


Figura 6.3

Modelado caso de uso buzón de alumnos.

13. Vista Lógica

Como se mencionó CIDE INTRANET será realizado haciendo uso de una arquitectura de tres niveles, utilizando el patrón de diseño Modelo Vista Controlador implementado a través del Framework Struts, a continuación se mostrará un diagrama realizado mediante UML indicando el flujo y relaciones entre los componentes del MVC.

La figura 7.1 muestra las relaciones entre los componentes de la Arquitectura de Software unidas a través de flechas indicando el sentido del flujo del sistema.

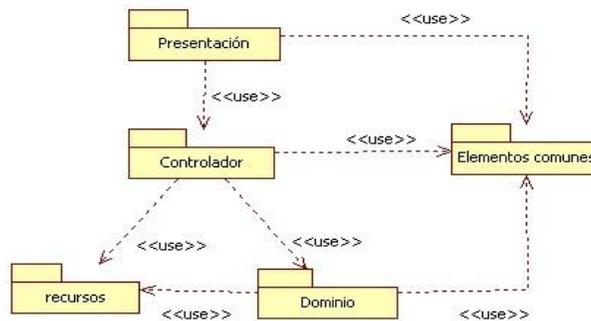


Figura 7.1

Dependencias de componentes

Capa	Función
Presentación	Muestra información al usuario, es representada por los archivos JSP'S.
Controlador	Gestión y acceso a la capa de dominio.
Dominio(Lógica de negocios)	Gestión de accesos a la capa de recursos.
Recursos(Integración)	Encargada de acceder a los sistemas de información solicitados.
Elementos de uso común	Conjunto de clases reutilizables en las capas anteriores

13.1 Repartición del Procesamiento

A continuación se muestra el modelado de las principales clases implementadas en CIDE INTRANET. El diagrama muestra del lado izquierdo el nombre del paquete que contiene o hace referencia a las clases ubicadas a la derecha; incluyendo los principales métodos en cada clase.

La figura 7.2 representa modelo de análisis de clases correspondientes al módulo de requerimientos, haciendo uso del subsistema Perfil encargado de consultar y establecer los permisos de usuario dentro del sistema. La clase de *ProcesaRequerimientos* se encarga del almacenamiento, consulta y autorización de requerimientos.

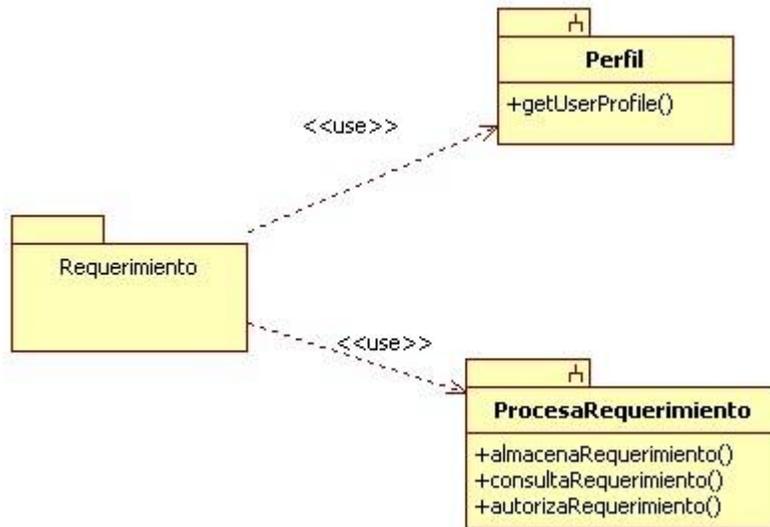


Figura 7.2

Resumen modelo análisis requerimientos

La figura 7.3 muestra el modelo de análisis de clases correspondientes al módulo de adeudos. El

sistema utiliza la clase Perfil para obtener los datos del usuario, la clase *ProcesamientoAdeudo* se encarga de realizar las operaciones de consulta de alumnos, almacenamiento, consulta y actualización de adeudos.

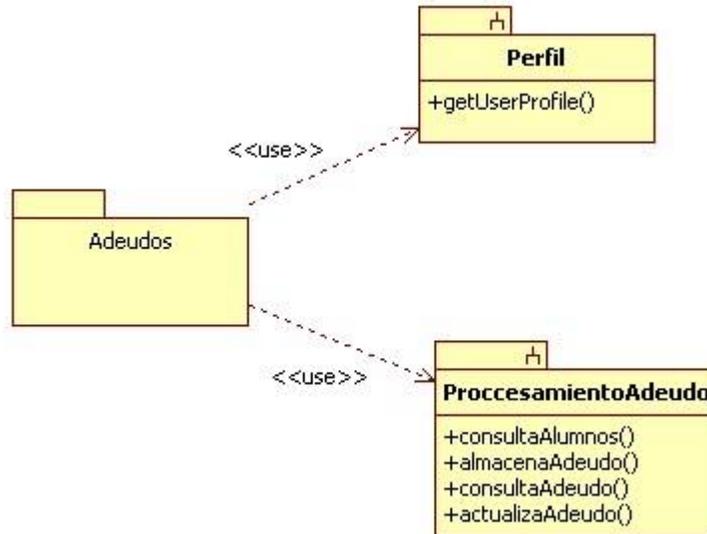


Figura 7.3

Resumen modelo análisis de adeudos

La figura 7.4 representa el modelado de clases correspondientes al módulo buzón de alumnos, conformado por la clase *ProcesaBuzonAlumnos* encargada de realizar las siguientes funciones: consulta de alumnos con cambio de estatus, actualización de estatus y validación del perfil de usuario.

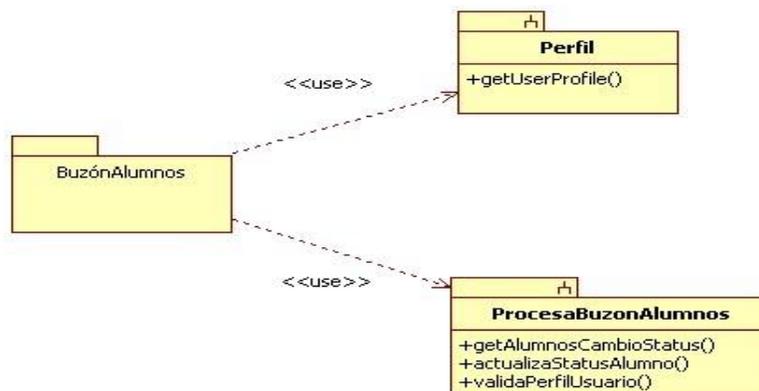


Figura 7.4

Resumen modelo análisis buzón de alumnos

14. Aseguramiento de la Calidad

El aseguramiento de la calidad dentro del ambiente de la aplicación estará determinado mediante los siguientes objetivos de calidad.

Objetivo	Descripción	Solución
Escalabilidad	El sistema debe contar con la capacidad de implementar nuevas funcionalidades en el momento en que las demandas del usuario aumenten.	Tanto el lenguaje Java como el servidor de aplicaciones Web cuentan con métodos y técnicas para el balanceo de cargas en caso de incremento abrupto de transacciones realizadas por el sistema.
Disponibilidad	El sistema debe encontrarse en servicio un 99% del día.	En caso de falla el servidor de pruebas será configurado para establecer conexión con el servidor de base de datos de producción, manteniendo la operación del sistema.
Portabilidad	El sistema deberá contar con la capacidad de ser implementado en entornos de producción diferentes.	El diseño implementado en la Arquitectura de Software permitirá la ejecución del sistema en cualquier entorno soportado por la <i>Java Virtual Machine</i> .
Seguridad	El sistema deberá contar con políticas de aseguramiento y manejo de información.	La seguridad estará implementada a través de la definición de perfiles de usuario dentro de la base de datos, los cuales serán almacenados en una sesión creada durante el ingreso del usuario al sistema.

CIDE INTRANET
Implementación

15. Objetivo de la implementación

La etapa de implementación tiene como objetivo poner en marcha el sistema desarrollado, siendo sometido a un proceso de evaluación dictaminado por la captura de requerimientos contenidos en el capítulo: Análisis de Requerimientos.

La etapa de implementación culmina una vez que el sistema cumple con los siguientes objetivos:

Puesta en marcha del sistema: El sistema es desplegado en el entorno de producción y se encuentra al alcance del usuario final.

Capacitación de usuarios: Tiene como finalidad mostrar con detalle el funcionamiento y capacidades del sistema.

La implementación se realizará de manera paralela, esto quiere decir que, SIGCIDE y CIDE INTRANET operarán juntos hasta que el usuario esté familiarizado con el funcionamiento del nuevo sistema. Este tipo de implementación permitirá realizar una transición segura, ya que si el sistema nuevo falla, la operación no se verá afectada.

A continuación se enlistan los apartados contenidos en la etapa de implementación. Cada uno de ellos será detallado con el mayor grado posible.

Calendario de trabajo: Especifica las etapas del sistema; implica cada una de las actividades que componen el proyecto de manera iterativa.

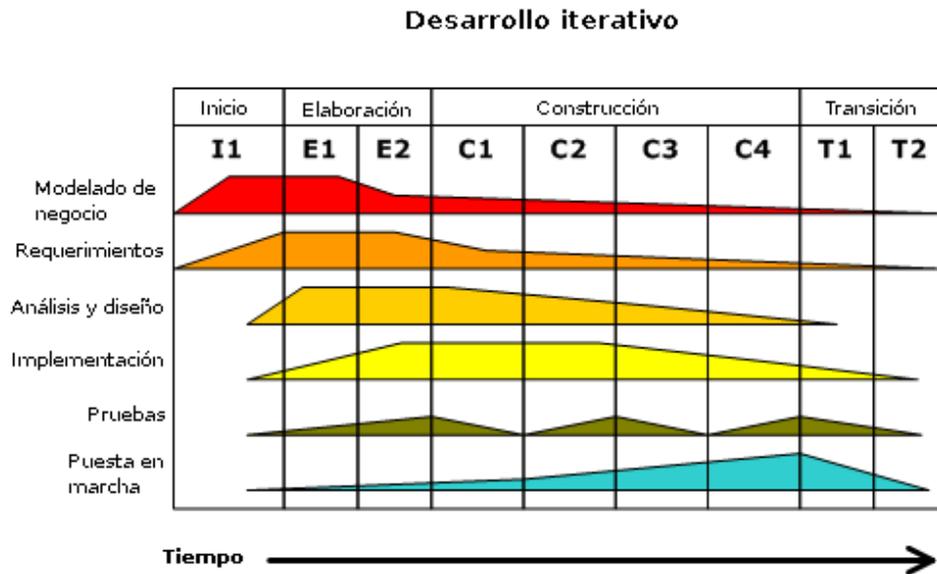
Guía de instalación: Especifica a nivel técnico los pasos para instalar el sistema, incluye los detalles correspondientes a la compilación y despliegue en el servidor web Tomcat.

Especificación de la aplicación: Muestra detalladamente el funcionamiento de los módulos de acuerdo al flujo del sistema.

Administración del sistema: Detalla la administración del sistema, identifica los archivos de configuración y esquema de la base de datos.

15.1 Calendario de trabajo Iterativo

A continuación se muestra de manera gráfica la representación del calendario de trabajo a modo iterativo. En la siguiente tabla se especifican las etapas: inicio, finalización y prioridad de los procesos implicados en el desarrollo de CIDE INTRANET.



Calendario de actividades

Etapa	Comienzo	Finalización	Prioridad
Modelado del Negocio			
Modelo de Casos de Uso del Negocio y Modelo de Objetos del Negocio	Semana 1	Semana 3	Normal
Requerimientos			
Visión	Semana 2	Semana 3	Normal
Modelo de Casos de Uso	Semana 3	Semana 4	Alta
Especificación de Casos	Semana 3	Semana 4	Alta

de Uso			
Análisis / Diseño			
Modelo de Análisis / Diseño	Semana 2	Siguiente fase	Normal
Modelo de Datos	Semana 2	Siguiente fase	Normal
Implementación			
Prototipos de Interfaces de Usuario	Semana 4	Semana 6	Alta
Modelo de Implementación	Semana 6	Siguiente fase	Normal
Pruebas			
Casos de Pruebas Funcionales	Semana 12	Semana 13	Critica
Puesta en marcha			
Despliegue del sistema, implementación a usuarios finales	Semana 13	Semana 14	Critica
Gestión de Cambios y Configuración	Durante todo el proyecto		Normal

16. Guía de instalación

El proceso de instalación comprende dos etapas, en la primera se compila el código fuente produciendo un archivo con extensión .WAR (Web Archive Resource); básicamente es un empaquetado de .jars producto de la compilación del código fuente, en la segunda etapa el archivo .WAR es copiado al servidor de producción para ser desplegado y accesible por los usuarios.

Requerimientos de compilación

En esta sección se especifican los requisitos mínimos para la compilación y ejecución de CIDE INTRANET en el servidor Web.

Requerimientos de hardware

- ✓ Microprocesador Pentium 2.6 Mhz o compatible.
- ✓ 512 MB en RAM, se recomienda 1 GB.
- ✓ Disco duro con 1 GB de espacio libre.

Esta es la configuración mínima necesaria para realizar la compilación de CIDE INTRANET.

Requerimientos de software

- ✓ Sistema operativo de 32 o 64 bits.
- ✓ Java Enterprise Edition version 1.5 o superior.
- ✓ IDE NETBEANS
- ✓ Servidor Web Tomcat

16.1 Compilación y ejecución de la aplicación

La etapa de compilación requiere el uso del entorno de desarrollo integrado NetBeans. El proyecto será compilado y empaquetado en un archivo .WAR, este archivo contendrá las clases compiladas y librerías utilizadas por el proyecto.



Figura 8.1

Inicio de IDE NETBEANS 6.5

La apertura del proyecto se realizará a través de la siguiente ruta: File → Open Project. La figura 8.2 muestra el cuadro de apertura de proyecto en NetBeans.

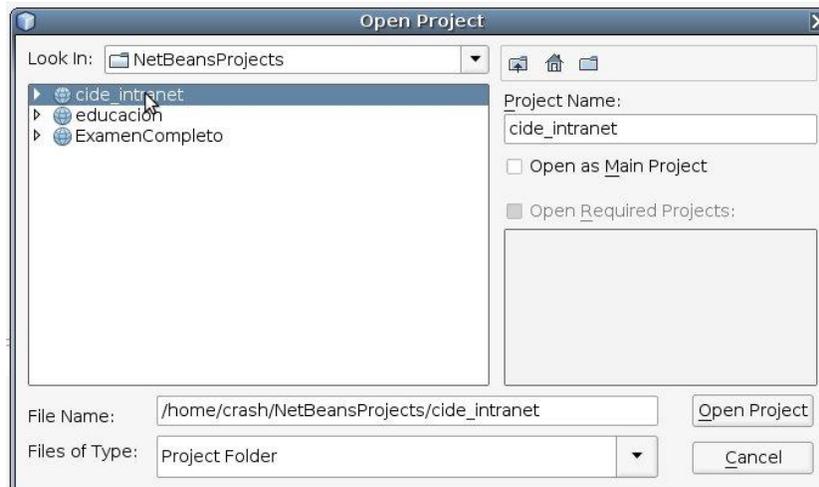


Figura 8.2

Cuadro de diálogo apertura de proyecto en NetBeans

Una vez abierto el proyecto se muestra la vista principal del IDE, en la parte superior se ubica el menú de herramientas para compilación y ejecución del proyecto, el lado izquierdo contiene la lista de paquetes pertenecientes al proyecto, al centro el editor de archivos y, en la parte inferior se muestran los mensajes emitidos por el compilador.

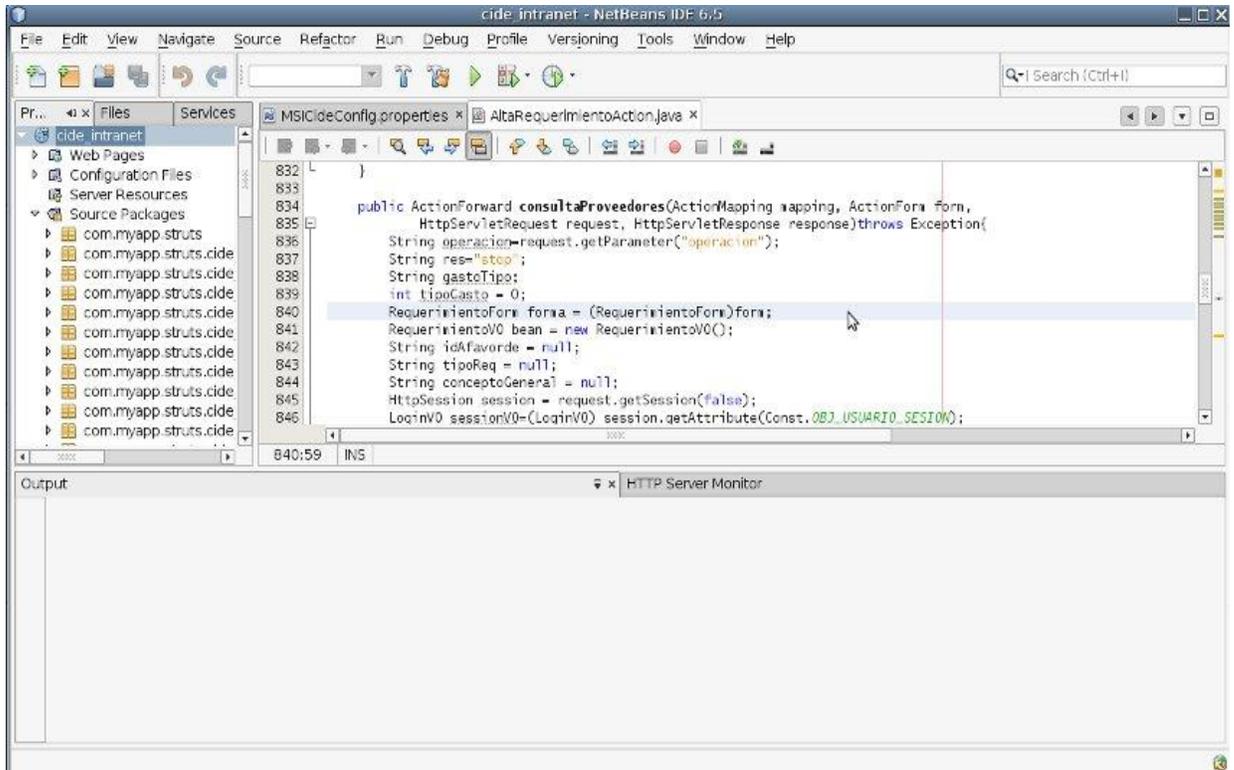


Figura 8.3
Entorno de desarrollo NetBeans

La compilación y empaquetado se realizará al presionar el icono con forma de martillo ubicado en la parte superior. Al finalizar la compilación y empaquetado el entorno de desarrollo mostrará un mensaje en la parte inferior, indicando si la compilación fue correcta o existió algún error.

En la figura 8.4 el lado izquierdo muestra el icono para realizar la compilación, mientras que el lado derecho se observa el mensaje producido al compilar el proyecto.



Figura 8.4
Icono y mensaje de compilación en NetBeans

Al término de la compilación se producirá el archivo cide_intranet.war, ubicado en la ruta: cide_intranet→dist. La figura 8.5 muestra el árbol de archivos contenidos en el proyecto.

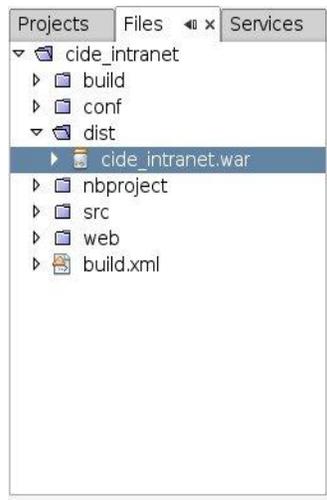


Figura 8.5
Archivo .war producto de la compilación

El archivo `cide_intranet.war` será copiado al servidor de producción a través del programa `winscp` en caso de ser compilado en un entorno Windows, si el código es compilado en un entorno Unix o Linux el archivo será copiado haciendo uso del comando `rsync`.

La figura 8.6 muestra ambos procedimientos de copiado, la primera imagen muestra el programa `winscp`, la imagen inferior muestra el copiado utilizando `rsync` en una terminal.

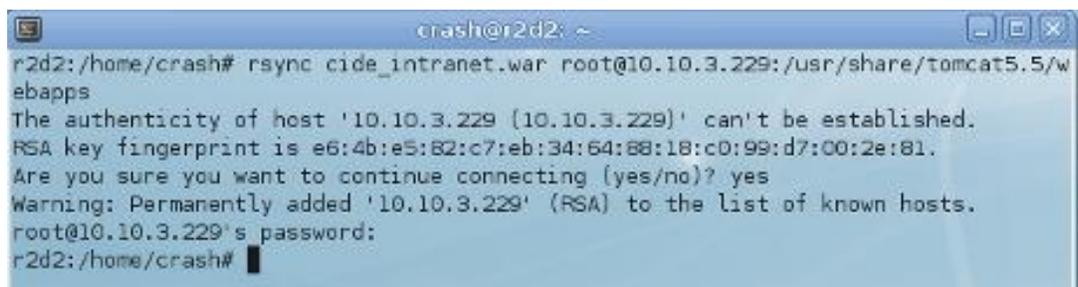
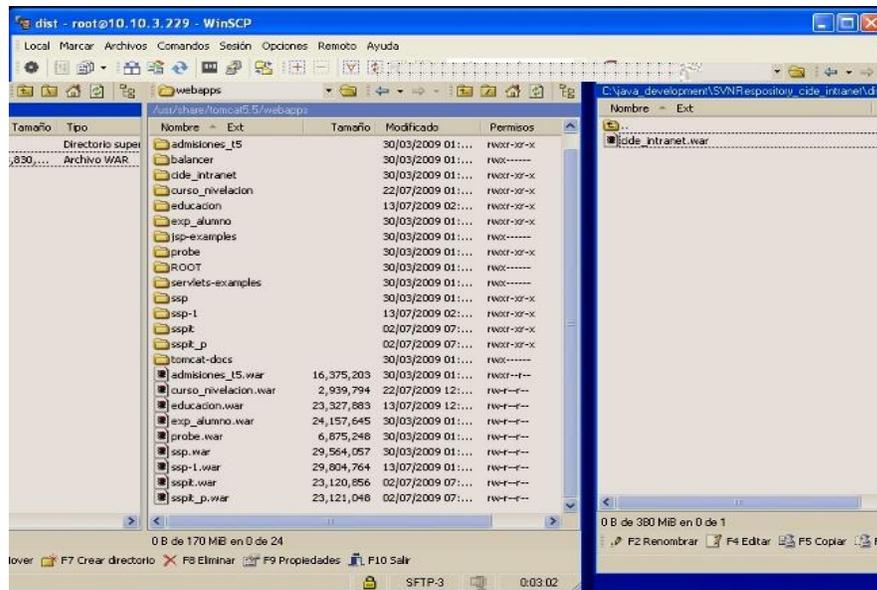


Figura 8.6

Copiado del archivo `cide_intranet.war` en diferentes arquitecturas

Una vez copiado el archivo `cide_intranet.war` en la ruta `/usr/share/tomcat` se deberá detener e iniciar el servidor web tomcat con la finalidad de desplegar la aplicación.

El servidor web tomcat será detenido ejecutando el script `shutdown.sh` ubicado en el directorio `/usr/share/tomcat5.5/bin`. En la figura 8.7 se muestra la manera en que se detiene el servidor Web Tomcat.

```
r2d2:/usr/share/tomcat5.5/webapps# /usr/share/tomcat5.5/bin/shutdown.sh
Using CATALINA_BASE:   /usr/share/tomcat5.5
Using CATALINA_HOME:   /usr/share/tomcat5.5
Using CATALINA_TMPDIR: /usr/share/tomcat5.5/temp
Using JRE_HOME:        /usr/lib/jvm/java-1.5.0-sun-1.5.0.19
```

Figura 8.7

Ejecución del script `shutdown.sh`

La figura 8.8 muestra la ejecución del script `startup.sh` con la finalidad de iniciar el funcionamiento del servidor Tomcat

A screenshot of a terminal window titled "crash@r2d2: ~". The terminal shows the execution of the script `/usr/share/tomcat5.5/bin/startup.sh`. The output displays the environment variables used by the script: `CATALINA_BASE`, `CATALINA_HOME`, `CATALINA_TMPDIR`, and `JRE_HOME`. The prompt returns to `r2d2:/usr/share/tomcat5.5/webapps#` after the script execution.

```
crash@r2d2: ~
r2d2:/usr/share/tomcat5.5/webapps# /usr/share/tomcat5.5/bin/startup.sh
Using CATALINA_BASE:   /usr/share/tomcat5.5
Using CATALINA_HOME:   /usr/share/tomcat5.5
Using CATALINA_TMPDIR: /usr/share/tomcat5.5/temp
Using JRE_HOME:        /usr/lib/jvm/java-1.5.0-sun-1.5.0.19
r2d2:/usr/share/tomcat5.5/webapps#
```

Figura 8.8

Ejecución del script `startup.sh`

Nota: La ejecución de los scripts `shutdown.sh` y `startup.sh` deberá realizarse por el superusuario o por un usuario con permisos en el directorio de instalación del servidor web Tomcat.

Para verificar el correcto despliegue de la aplicación teclearemos la dirección ip del servidor, seguido de una diagonal y el nombre de la aplicación, en este caso `cide_intranet`.

Ejemplo: http://10.10.3.229:8084/cide_intranet



Figura 8.9

Despliegue de la aplicación CIDE INTRANET en una máquina cliente

16.2 Especificación de la aplicación

Acceso al sistema

El ingreso al sistema se realizará a través del enlace mostrado en el portal interno del CIDE o al ingresar la dirección del servidor web donde reside la aplicación. En cualquiera de los dos casos el usuario accederá a la pantalla mostrada en la figura 9.1 donde le serán requeridos su nombre de usuario y contraseña.



Figura 9.1

Pantalla de acceso CIDE INTRANET

En caso de que el usuario ingrese de manera errónea su usuario o clave, el sistema enviará un mensaje indicando el error. El sistema es indistinto a mayúsculas y minúsculas, esto quiere decir que el usuario podrá ingresar su cuenta y clave de cualquier forma, siempre y cuando ambos datos sean correctos, en caso contrario se mostrará el siguiente mensaje.



Figura 9.2

Mensaje error de cuenta o clave de usuario

En caso de que ambos datos sean correctos el sistema mostrará la pantalla inicial acorde al perfil de usuario que ingresa al sistema. La figura 9.3 muestra la pantalla de inicio acorde a un usuario con acceso a los módulos de *Requerimientos*, *Adeudos* y *Alumnos*.

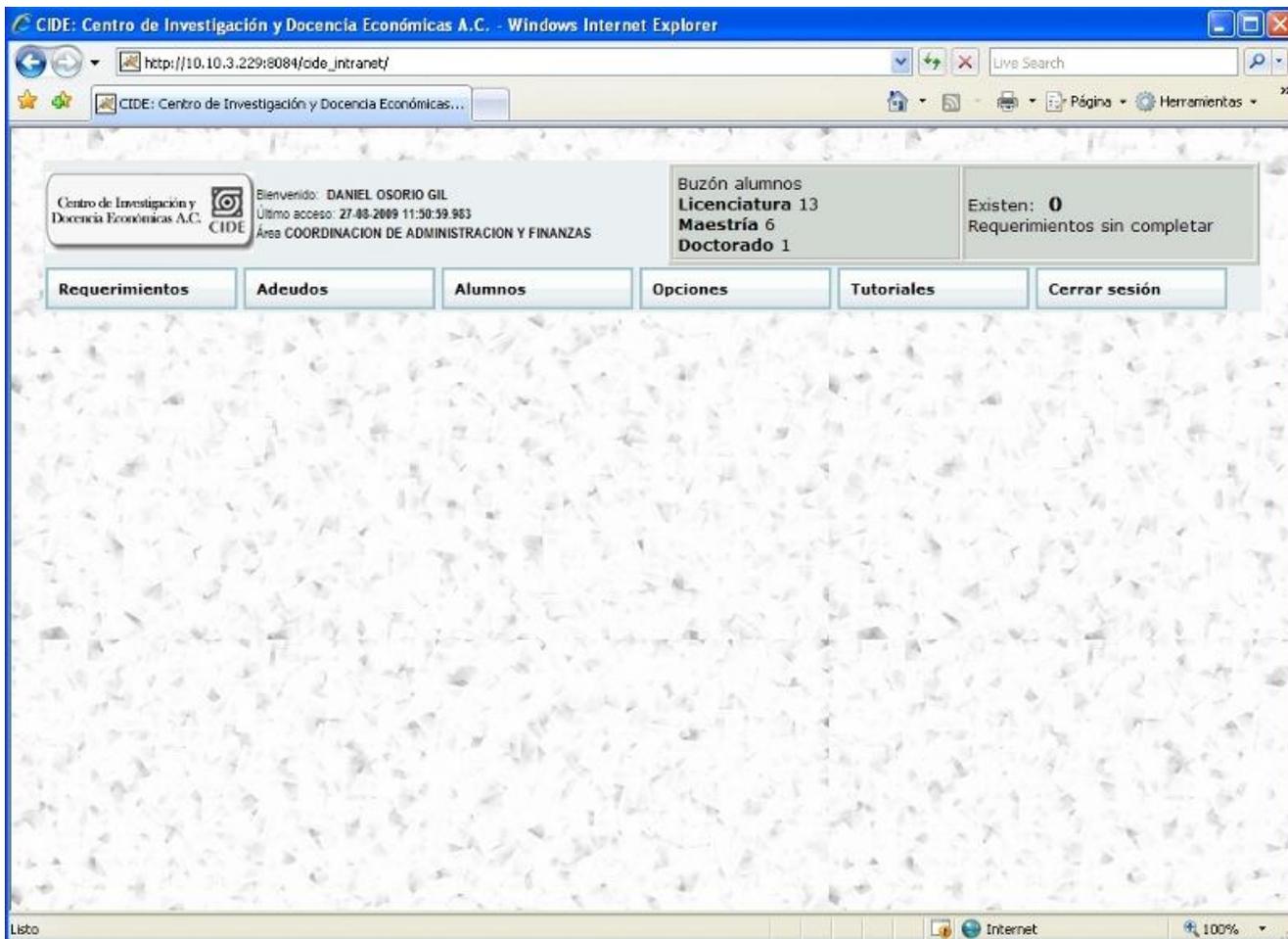


Figura 9.3

Menú inicial CIDE INTRANET

Modulo de Requerimientos

El módulo de requerimientos permitirá capturar, consultar, modificar, autorizar y rechazar los diferentes tipos de requerimientos establecidos para cada una de las áreas del CIDE. En cada requerimiento el usuario deberá de ingresar los datos acordes al tipo de requerimiento seleccionado.

La figura 9.4 muestra el menú de requerimientos.



Figura 9.4

Menú de requerimientos

Captura de requerimientos

Esta opción permite al usuario capturar requerimientos de gasto, el sistema automáticamente mostrará el Ejercicio fiscal y la Fecha actual del requerimiento. En la parte superior se mostrará una barra indicando las distintas etapas del requerimiento, a medida que el usuario complete las distintas etapas la barra será iluminada.



Figura 9.5

Barra de progreso captura de requerimientos

En caso de que el usuario tenga acceso a más de un área el sistema mostrará un campo de selección en el cual se deberá de especificar el área que realiza el requerimiento. Después de seleccionar el área a favor de la cual se realiza el requerimiento, el sistema mostrará los usuarios con posibilidad de autorizar el requerimiento.

La figura 9.6 muestra la selección de área, así como la barra de progreso indicando el avance del requerimiento.

The screenshot displays a web application interface. At the top, there is a header section with the logo of 'Centro de Investigación y Docencia Económicas A.C. CIDE' and user information: 'Bienvenido: DANIEL OSORIO GIL', 'Último acceso: 31-08-2009 09:48:27.03', and 'Área COORDINACION DE ADMINISTRACION Y FINANZAS'. To the right, it shows 'Buzón alumnos' with counts for 'Licenciatura 13', 'Maestría 6', and 'Doctorado 1', and a status 'Existen: 0 Requerimientos sin completar'. Below the header is a navigation menu with buttons for 'Requerimientos', 'Adeudos', 'Alumnos', 'Opciones', 'Tutoriales', and 'Cerrar sesión'. A progress bar below the menu shows four steps: 'Paso 1: Seleccione el área' (highlighted in blue), 'Paso 2: Llene los campos obligatorios', 'Paso 3: Guarde el requerimiento', and 'Paso 4: Agregue los detalles'. The main form area contains several fields: 'No. Folio:' with value 'CAF/122', 'Ejercicio Fiscal:' with value '2009', and 'Fecha:' with value '31-08-2009'. There are 'AGREGAR' and 'SALIR' buttons at the top of the form. The 'Área' field is a dropdown menu with 'COORDINACION DE ADMINISTRACION Y FINANZAS' selected. The 'Quien Autoriza (Área)' field is a dropdown menu with 'CELIS GARCIA PABLO JAVIER.' selected. The 'Solicita' field is a text input with a red asterisk and a magnifying glass icon to its right. The 'Tipo de Requerimiento' field is a dropdown menu with a red asterisk. The 'Observaciones' field is a text input. At the bottom of the form, there is an 'AGREGAR NOTAS' button and another set of 'AGREGAR' and 'SALIR' buttons.

Figura 9.6

Barra de progreso y selección de Área

Una vez seleccionada la persona que autorizará el requerimiento se deberá ingresar el nombre de la persona que solicita el requerimiento, para ello será necesario realizar una búsqueda seleccionando el icono con forma de lupa ubicado al extremo derecho del campo "Solicita", a continuación se desplegará una ventana para realizar la búsqueda. En caso de haber realizado una asignación errónea se deberá presionar el icono ubicado en la parte

inferior o presionar nuevamente el ícono de lupa y realizar una nueva búsqueda.



Figura 9.7

Búsqueda de usuario solicitante

Después de ingresar los campos mencionados se deberá ingresar el tipo de requerimiento; la selección del tipo determinará los campos que serán mostrados para completar la pantalla principal del requerimiento.



Figura 9.8

Selección tipo de requerimiento

Una vez seleccionado el tipo de requerimiento el sistema mostrará los campos necesarios para completar la pantalla principal, entre los cuales se encuentran:

- ✓ **Proyecto:** Tipo de proyecto al que pertenece el requerimiento
- ✓ **Concepto general:** Descripción de los motivos para solicitar el requerimiento.
- ✓ **A favor de:** Persona a la cual se le asignará el recurso solicitado en caso de ser aprobado.
- ✓ **Divisa:** Tipo de divisa usada en el requerimiento.
- ✓ **Impuesto único para todos los detalles:** Tipo de impuesto aplicable a los detalles que serán capturables.
- ✓ **Justificación:** Justificación del requerimiento.
- ✓ **Observaciones:** Información adicional del requerimiento.

En la figura 9.9 se muestra la captura de un requerimiento de tipo gastos, en la parte superior se visualiza la barra de progreso completada hasta el tercer paso.

Paso 1:	Paso 2:	Paso 3:	Paso 4:
Seleccione el área	Llene los campos obligatorios	Guarde el requerimiento	Agregue los detalles

<input type="button" value="AGREGAR"/> <input type="button" value="SALIR"/>					
No. Folio:	CAF/122	Ejercicio Fiscal:	2009	Fecha:	31-08-2009
Área	* COORDINACION DE ADMINISTRACION Y FINANZAS	Proyecto	* Institucional		
Quien Autoriza (Área)	DIAZ BARRIOS MAGDALENO				
Solicita	* OSORIO GIL DANIEL				
Tipo de Requerimiento	* GASTOS				
Concepto General	* Gasolina periodo 1 al 31 agosto				
A Favor de	* Proveedor <input type="radio"/> Empleado <input checked="" type="radio"/> Estudiante <input type="radio"/> General <input type="radio"/>				
	* OSORIO GIL DANIEL				
Divisa	<input type="checkbox"/> Dólar, Valor de caja	<input checked="" type="checkbox"/> Impuesto único para todos los detalles	IVA 15%		
	PESOS				
Justificación	* Gasolina empleada para transporte CIDE SANTA FE - CIDE CONSTITUYENTES				
Observaciones	Gastos de gasolina				
<input type="button" value="AGREGAR NOTAS"/>					
<input type="button" value="AGREGAR"/> <input type="button" value="SALIR"/>					

Figura 9.9

Requerimiento de tipo gastos

Al completar la pantalla principal el sistema agregará el requerimiento con un folio temporal acorde al área que solicita el requerimiento; en la imagen 10.1 se muestra el mensaje indicando el folio temporal con el cual es agregado el requerimiento.



Una vez que el requerimiento tiene folio temporal el botón *Agregar* es remplazado por el botón *Ir al detalle*, permitiendo acceder al detalle del requerimiento; en el caso de un requerimiento sin detalles la pantalla lucirá como la figura 10.2

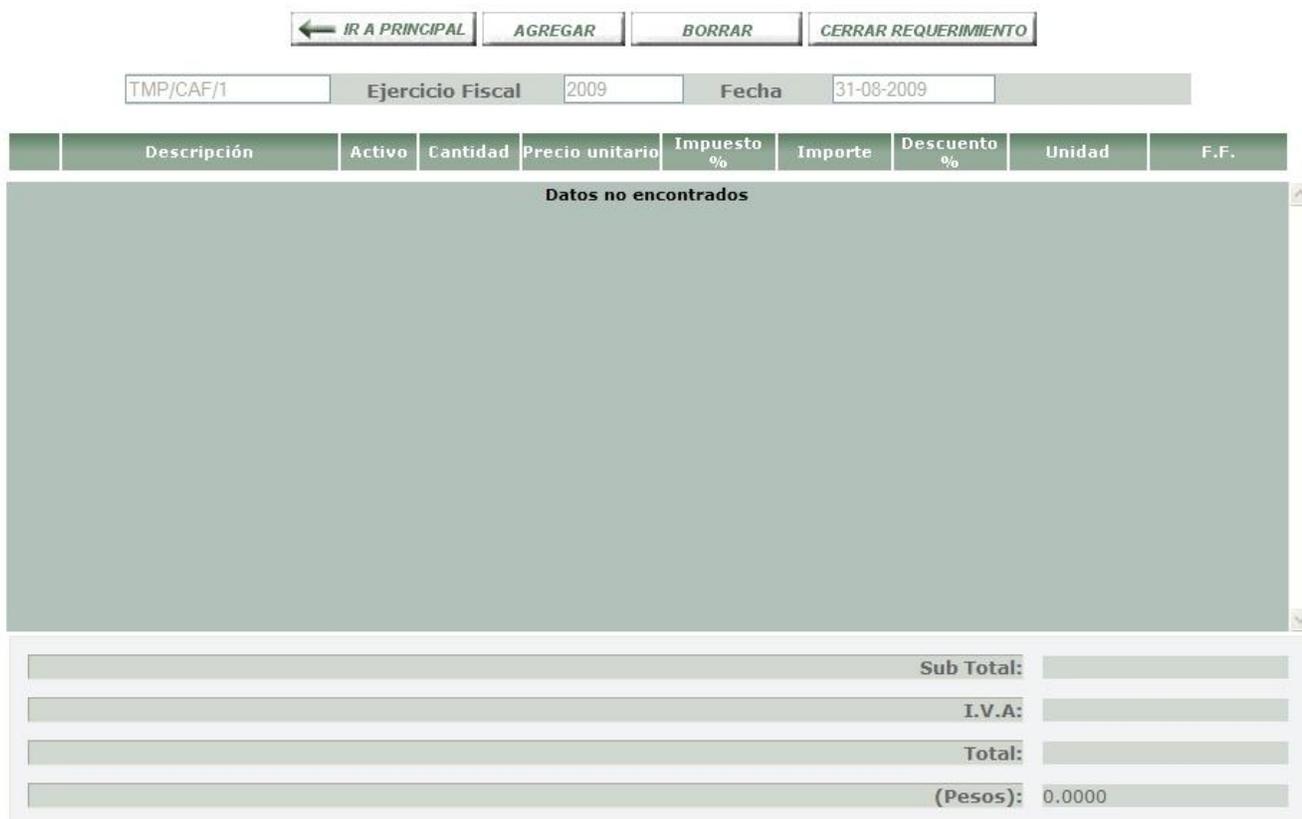


Figura 10.2

Detalle de requerimientos

La pantalla de detalle mostrará en la parte superior los botones de las acciones permitidas, el botón *IR A PRINCIPAL* regresará a la pantalla principal; *AGREGAR* permitirá añadir un detalle nuevo; *BORRAR* eliminará un detalle seleccionado; *CERRAR REQUERIMIENTO* almacenará los detalles y generará un folio oficial al requerimiento.

En la figura 10.3 se muestra el formulario para agregar elementos al detalle del requerimiento, la unidad deberá ser seleccionada acorde al tipo de requerimiento.

Field	Left Screenshot	Right Screenshot
Descripción	[Empty]	Gasolina periodo 1 agosto al 30 agosto 2009
Unidad	Seleccione	LITRO
Cantidad	[Empty]	25
Precio unitario	[Empty]	7.20
Impuesto	IVA 16%	IVA 15%
Proveedor	Empleado: OSORIO GIL DANIEL	Empleado: OSORIO GIL DANIEL

Figura 10.3

Formulario detalle de requerimiento

Una vez que el requerimiento es agregado la pantalla del detalle muestra un compendio con la información desglosada.

DAE/87	Ejercicio Fiscal	2009	Fecha	31-08-2009				
Descripción	Activo	Cantidad	Precio unitario	Impuesto %	Importe	Descuento %	Unidad	F.F.
<input type="checkbox"/> Pago registros de cédula	No	10.00	150.00	0.15	1500.00	0.00	PZA.	Recursos Fiscales
Sub Total:						1500.0		
I.V.A.:						225.00002		
Total:						1725.0		
(Pesos):						1725.0000		

Figura 10.4

Detalles del requerimiento

Al cerrar el requerimiento el sistema asigna el folio oficial y envía un mensaje de confirmación indicando el folio oficial.



Figura 10.5

Mensaje confirmación folio oficial

✓ *Autorización / rechazo de requerimientos*

La autorización / rechazo de requerimientos será realizada una vez que el requerimiento cuente con un folio oficial, el usuario designado en el campo "Quién Autoriza" será notificado al ingresar al sistema.

El menú principal mostrará un apartado en el cual se indicará la cantidad de requerimientos pendientes por autorizar.



Figura 10.6

Menú de requerimientos por autorizar

Al presionar sobre el recuadro "Requerimientos sin autorizar" se proporcionará la lista de requerimientos pendientes por autorizar como se muestra en la siguiente figura.



Resultado de la búsqueda para Requerimientos pendientes de Autorización						
Folio	Nombre solicitante	Área solicitante	Fecha	Descripción	Importe total	Observaciones
CAF/122	OSORIO GIL DANIEL	COORDINACION DE ADMINISTRACION Y FINANZAS	31-08-2009	Gasolina periodo 1 al 31 agosto	207.00	Gastos de gasolina

El usuario deberá seleccionar el folio del requerimiento que desea consultar, el sistema mostrará la pantalla principal del requerimiento, a través de la cual se podrán consultar los detalles del requerimiento o autorizar o rechazar un requerimiento.

En la figura 10.7 se muestra el menú ubicado en la parte superior del requerimiento en el cual se podrá seleccionar la autorización o rechazo del requerimiento.

Figura 10.7

Menú autorización / rechazo de requerimientos

En caso de que el usuario quiera autorizar el requerimiento será necesario presionar el botón "Autorizar", solicitando la firma electrónica del usuario, el nombre de usuario y contraseña utilizada para ingresar al sistema.

Figura 10.8

Firma autorización requerimiento

En caso de ser errónea la cuenta o clave, el sistema informará mediante una notificación, misma que se ejemplifica en la figura 10.9



Figura 10.9

Error firma electrónica

El requerimiento será autorizado al ingresar de manera correcta la cuenta y clave de usuario mostrando una pantalla con los datos más importantes del requerimiento, así como un mensaje indicando el usuario que ha autorizado el requerimiento.

REQUERIMIENTO DE GASTOS	
Área	CAF
Folio	CAF/122
Fecha	31-08-2009
Ejercicio fiscal	2009
Prioridad	Normal
¿Quién Autoriza?	DIAZ BARRIOS MAGDALENO
Proyecto	INSTITUCIONAL
Solicita	OSORIO GIL DANIEL
Tipo requerimiento	GASTOS
Tipo de persona	Empleado
A favor de	OSORIO GIL DANIEL
Divisa	PESOS = 1.000000000000000000
Concepto general	Gasolina periodo 1 al 31 agosto
Un solo proveedor para todos los detalles	Sí
Impuesto único para todos los detalles	Sí
Cotización	
Observaciones	Gastos de gasolina
Nota	
Justificación	Gasolina empleada para transporte CIDE SANTA FE - CIDE CONSTITUYENTES

Figura 11.1

Requerimiento autorizado

Rechazo de requerimientos

Los requerimientos que no cumplan con los lineamientos especificados serán rechazados, para ello será necesario ingresar un breve comentario indicando el motivo de rechazo.



The screenshot shows a web interface titled "Firmas". On the left, there is a form titled "Introduzca su Firma electrónica" with fields for "Cuenta:", "Clave:", and "Causa del rechazo:". Below these fields are two buttons: "RECHAZAR" and "CANCELAR". To the right of the form, there are two buttons: "Autorizar" and "Rechazar", with the text "Responsable de Área" positioned between them.

Figura 11.2

Firma rechazo de requerimiento

Cuando el requerimiento es autorizado o rechazado el buzón de *requerimientos sin autorizar* será actualizado. En la figura 11.3 se muestra la actualización indicando la falta de requerimientos por autorizar.



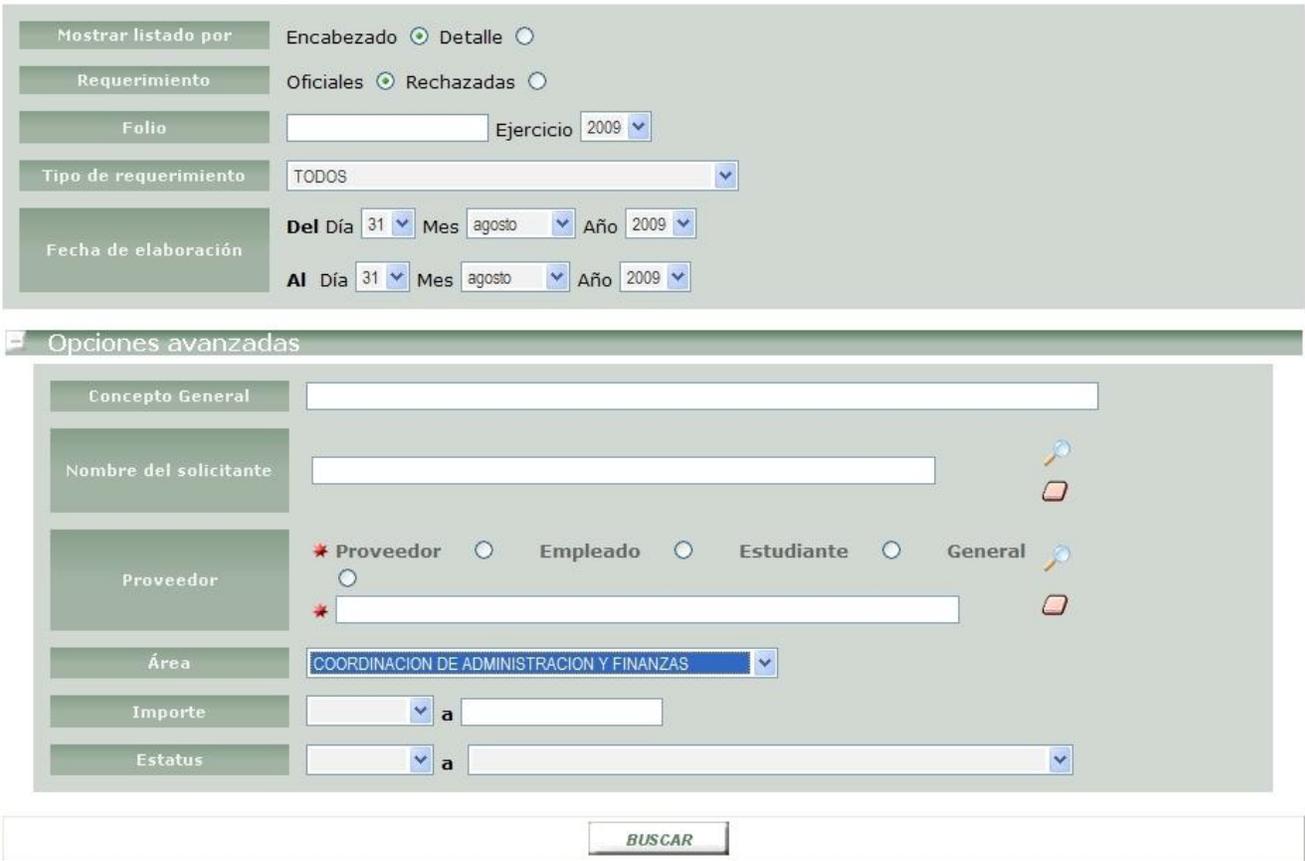
The screenshot displays a user dashboard. At the top left, it shows the logo for "Centro de Investigación y Docencia Económicas A.C. CIDE" and a welcome message for "DIAZ BARRIOS MAGDALENO" from the "Área COORDINACION DE ADMINISTRACION Y FINANZAS". In the top right, there are statistics: "Buzón alumnos" with "Licenciatura 13", "Maestría 6", and "Doctorado 1"; "Usted tiene: 0 Requerimientos sin autorizar"; and "Existen: 0 Requerimientos sin completar". Below these are navigation buttons for "Requerimientos", "Adeudos", "Alumnos", "Opciones", "Tutoriales", and "Cerrar sesión". The main content area is titled "Resultado de la búsqueda para Requerimientos pendientes de Autorización" and contains a table with columns: "Folio", "Nombre solicitante", "Área solicitante", "Fecha", "Descripción", "Importe total", and "Observaciones". The table is currently empty, with the text "Datos no encontrados" displayed in the center.

Figura 11.3

Buzón de requerimientos sin autorizar

Consulta de requerimientos

La consulta de requerimientos permitirá localizar un requerimiento o varios a través de diversos parámetros. En caso de que el usuario quiera realizar una búsqueda más detallada deberá de hacer clic en el botón , desplegándose un menú en donde el usuario podrá ingresar un mayor número de parámetros. El menú de opciones avanzadas podrá contraerse haciendo clic en el botón. 



Mostrar listado por Encabezado Detalle

Requerimiento Oficiales Rechazadas

Folio Ejercicio 2009

Tipo de requerimiento TODOS

Fecha de elaboración Del Día 31 Mes agosto Año 2009
Al Día 31 Mes agosto Año 2009

Opciones avanzadas

Concepto General

Nombre del solicitante  

Proveedor Proveedor Empleado Estudiante General  

Área COORDINACION DE ADMINISTRACION Y FINANZAS

Importe a

Estatus a

Figura 11.4

Pantalla de consulta requerimientos

El usuario ingresará los criterios de búsqueda acordes al tipo de requerimiento que desea localizar, el sistema realizará la búsqueda mediante los parámetros ingresados y perfil de usuario; limitando así los resultados.

La figura 11.5 muestra el resultado de una consulta de requerimientos utilizando un rango de fechas, en la parte superior e inferior se muestran los botones con las posibles acciones a realizar para un requerimiento.

<input type="button" value="CONVERTIR A PLANTILLA"/> <input type="button" value="IMPRIMIR NOTA"/> <input type="button" value="IMPRIMIR"/> <input type="button" value="EXCEL"/> <input type="button" value="REGRESAR"/> <input type="checkbox"/> Ver folio asociado							
Folio	Área	Fecha	Proveedor	Cheque	Tipo requerimiento	Descripción estatus	Vista
DVE/197	DIVISION DE ECONOMIA	26-08-2009	AURA ERENDIRA MARTINEZ ORIOL		COMPROBACIÓN DE GASTO	Requerimiento a revisar por parte del Responsable de Area	 <input type="radio"/>
DRF/395	DIRECCION DE RECURSOS FINANCIEROS	26-08-2009	AMERICAN EXPRESS COMPANY MEXICO S.A. DE C.V.		COMPROBACIÓN DE GASTO	Requerimiento a revisar por parte del Responsable de Area	 <input type="radio"/>
SSG/1030	SUBDIRECCION DE SERVICIOS GENERALES	26-08-2009			COMPROBACIÓN DE GASTO	Requerimiento a revisar por parte del Responsable de Area	 <input type="radio"/>
CAF/120	COORDINACION DE ADMINISTRACION Y FINANZAS	26-08-2009	VEGA PACHECO MARIA DE LOURDES		COMPROBACIÓN DE GASTO	Requerimiento a revisar por parte del Responsable de Area	 <input type="radio"/>
DRF/396	DIRECCION DE RECURSOS FINANCIEROS	26-08-2009	AMERICAN EXPRESS COMPANY MEXICO S.A. DE C.V.		COMPROBACIÓN DE GASTO	Requerimiento a revisar por parte del Responsable de Area	 <input type="radio"/>
CAF/121	COORDINACION DE ADMINISTRACION Y FINANZAS	26-08-2009	VEGA PACHECO MARIA DE LOURDES		COMPROBACIÓN DE GASTO	Requerimiento a revisar por parte del Responsable de Area	 <input type="radio"/>

Figura 11.5

Resultado consulta de requerimiento

El resultado de la búsqueda de requerimientos será mostrado mediante columnas, en la primera se muestra el folio del requerimiento, la segunda muestra el área a favor de la cual es realizado el requerimiento, la tercera columna muestra la fecha, la cuarta indica el nombre del proveedor a favor del cual se realiza el requerimiento; la columna cheque indica el número de cheque emitido en caso de representar un pago, la siguiente columna especifica el tipo de requerimiento, la descripción de estatus indica la etapa en que se encuentra el requerimiento, el campo ubicado al final permitirá el acceso a la consulta de requerimientos sin posibilidad de modificación.

REQUERIMIENTO DE GASTOS	
Área	CAF
Folio	CAF/122
Fecha	31-08-2009
Ejercicio fiscal	2009
Prioridad	Normal
¿Quién Autoriza?	DIAZ BARRIOS MAGDALENO
Proyecto	INSTITUCIONAL
Solicita	OSORIO GIL DANIEL
Tipo requerimiento	GASTOS
Tipo de persona	Empleado
A favor de	OSORIO GIL DANIEL
Divisa	PESOS = 1.00000000000000000000
Concepto general	Gasolina periodo 1 al 31 agosto
Un solo proveedor para todos los detalles	Sí
Impuesto único para todos los detalles	Sí
Cotización	
Observaciones	Gastos de gasolina

Figura 11.6

Consulta de requerimiento sin posibilidad de modificación

Actualización de requerimientos

La actualización se realizará por medio de la búsqueda de requerimientos, el usuario deberá de seleccionar el folio del requerimiento que desea actualizar. Una vez seleccionado el folio el sistema mostrará la pantalla principal del requerimiento.

Folio
DVE/197
DRF/395
SSG/1030
CAF/120
DRF/396
CAF/121

Figura 11.7

Resultado folio consulta de requerimientos

En caso de que el folio seleccionado aún no haya sido autorizado el sistema mostrará la pantalla principal del requerimiento; mostrada en la figura 11.8

The screenshot shows a web application interface for managing requirements. At the top, there are buttons for 'Modificar', 'Ir al detalle' (with a right arrow), and 'SALIR'. Below this, the form is organized into several sections:

- No. Folio:** TMP/CAF/1
- Ejercicio Fiscal:** 2009
- Fecha:** 31-08-2009
- Área:** COORDINACION DE ADMINISTRACION Y FINANZAS (with a red asterisk)
- Proyecto:** Institucional (with a red asterisk)
- Quien Autoriza (Área):** DIAZ BARRIOS MAGDALENO
- Solicita:** OSORIO GIL DANIEL (with a red asterisk)
- Tipo de Requerimiento:** GASTOS (with a red asterisk)
- Concepto General:** Gasolina periodo 1 al 31 agosto (with a red asterisk)
- A Favor de:** Proveedor (selected), Empleado, Estudiante, General (all with radio buttons). Below it, OSORIO GIL DANIEL (with a red asterisk).
- Divisa:** Dólar, Valor de caja; PESOS. Impuesto único para todos los detalles. IVA 15% (dropdown).
- Justificación:** Gasolina empleada para transporte CIDE SANTA FE - CIDE CONSTITUYENTES (with a red asterisk)
- Observaciones:** Gastos de gasolina

At the bottom, there is a button 'AGREGAR NOTAS' and another set of buttons: 'Modificar', 'Ir al detalle' (with a right arrow), and 'SALIR'.

Figura 11.8

Actualización de requerimiento

En caso de que el usuario intente actualizar un requerimiento previamente autorizado el sistema enviará un mensaje notificando el motivo por el cual no es posible actualizar el requerimiento.



Figura 11.9

Mensaje requerimiento actualizado

Módulo adeudos

El módulo de adeudos tiene como objetivo realizar de manera rápida y eficiente la captura, consulta y actualización de adeudos en diferentes áreas, en la figura X.X se mostrará el menú de Adeudos.



Figura 12.1

Menú adeudos

Búsqueda alumnos deudores

La búsqueda de alumnos deudores permitirá localizar mediante el ingreso de parámetros a un alumno o grupo de alumnos sancionados por alguna de las áreas del CIDE.

Alumnos con Adeudo (Créditos Educativos)

Criterios de búsqueda

Matrícula: Alumno(s) Deudores Alumno(s) no deudores Ambos

Nombre: Apellido paterno: Apellido materno: Generación: ▼

Figura 12.2

Formulario búsqueda alumnos deudores

El usuario deberá ingresar los parámetros que crea pertinentes para realizar la búsqueda, en caso de que se ingrese la matrícula y el alumno no registre adeudo, el sistema notificará mediante una leyenda mostrada en la figura 12.3

Alumnos con Adeudo (Créditos Educativos)

Criterios de búsqueda

Matrícula: Alumno(s) Deudores Alumno(s) no deudores Ambos

Nombre: Apellido paterno: Apellido materno: Generación

**El alumno no tiene registrados
Adeudos (pendientes y finiquitados)
asociados al área de DIRECCION DE ADMINISTRACION ESCOLAR**

Figura 12.3

Alumno no deudor

En caso de que la búsqueda de alumnos produzca resultados, se mostrará un listado de alumnos que hayan coincidido con los parámetros de búsqueda ingresados.

La figura 12.4 muestra el resultado de la búsqueda realizada; el listado incluirá los siguientes datos:

Matrícula del alumno: Permitirá al usuario acceder al detalle de adeudos del alumno seleccionado.

Nombre del alumno: Nombre completo del alumno organizado por apellido paterno, apellido materno y nombres.

Generación: Generación a la cual pertenece el alumno.

Programa: Carrera en que se encuentra inscrito el alumno.

Compendio de adeudos: Muestra los adeudos indicando en la parte superior el área que registró el adeudo.

Criterios de búsqueda

Matrícula: Alumno(s) Deudores Alumno(s) no deudores Ambos

Nombre: Apellido paterno: Apellido materno: Generación

Datos generales de alumnos sin adeudos			
Matrícula	Nombre	Generación	Programa
0211087	Martínez Barranco Kukutschka Roberto	2002	LIC. EN CIENCIA POL y REL.INT.
0311128	Martínez Zárate Rosalinda	2003	LIC. EN DERECHO
0412027	Martínez Méndez Juan Ismael	2004	M. EN ADMINISTRACIÓN Y POLÍTICAS PÚBLICAS
0412028	Martínez Sánchez Enrique	2004	M. EN ADMINISTRACIÓN Y POLÍTICAS PÚBLICAS
0512008	Martínez Ríos Omar Francisco	2005	M. EN ECONOMÍA
0112014	Martínez Solorzano Bessy Carolina	2005	M. EN ECONOMÍA
0112039	Martínez Medrano Carlos Iván	2005	M. EN ADMINISTRACIÓN Y POLÍTICAS PÚBLICAS
0611042	Martínez Oriol Aura Eréndira	2006	LIC. EN CIENCIA POL y REL.INT.
0711015	Martínez Lechuga Isabel	2007	LIC. EN ECONOMÍA
0712015	Martínez Castillo Raúl	2007	M. EN ECONOMÍA
0712016	Martínez Magaña Erick Rafael	2007	M. EN ECONOMÍA
0811018	Martínez Roa Gabriel Alberto	2008	LIC. EN ECONOMÍA

Figura 12.4

Resultado búsqueda alumnos deudores

El usuario accederá al detalle de adeudo seleccionando la matrícula del alumno. La figura 12.5 muestra el detalle de adeudos pertenecientes a un alumno.

Matrícula	0211087	Nombre del alumno	Martínez Barranco Kukutschka Roberto
Generación	2002	Programa	LIC. EN CIENCIA POL y REL.INT.

Adeudos pendientes del alumno							
ADEUDO	IMPORTE	ESTATUS	REGISTRO	PAGO-AUT-CAN	DESCRIPCIÓN	INGRESO	RECIBO
MULTA BIB	330.0000	SIN PAGO	21-11-2008 00:00:00		MATERIAL VENCIDO 27/10/08		

Adeudos finiquitados del alumno							
ADEUDO	IMPORTE	ESTATUS	REGISTRO	PAGO-AUT-CAN	DESCRIPCIÓN	INGRESO	RECIBO
MULTA BIB	50.0000	PAGADO EN CAJA	10-04-2007 00:00:00	10-05-2008 00:00:00	mat. venc. 05/03/2007	65204	40124
COLEGIATURA	5500.0000	PAGADO EN CAJA	12-08-2005 00:00:00	12-08-2005 00:00:00	PAGO 1		
COLEGIATURA	8000.0000	PAGADO EN CAJA	12-08-2005 00:00:00	12-08-2005 00:00:00	PAGO 2		
COLEGIATURA	8147.0000	PAGADO EN CAJA	12-08-2005 00:00:00	12-08-2005 00:00:00	ADEUDO COLEGIATURA	7729	13336

Figura 12.5

Detalle alumno deudor

El detalle de alumnos deudores en la parte superior muestra los datos del alumno, el resto de la página está dividido en tres secciones desglosadas de la siguiente manera:

Agregar adeudo: Por omisión se encuentra minimizado, el usuario podrá expandir el formulario haciendo clic en el signo (+)

Adeudos pendientes: Listado de adeudos no finiquitados, se muestran los aspectos más relevantes del adeudo: área, importe, estatus, descripción.

Adeudos finiquitados: Historial de adeudos generados por el alumno.

Captura de adeudo

El usuario podrá agregar un adeudo a través del detalle o al ingresar a la opción “*Adeudo Nuevo*” ubicada en el menú del sistema.

La figura 12.6 muestra el formulario para registrar un adeudo.



El formulario, titulado "Registrar adeudo", contiene los siguientes elementos:

- Un campo de texto vacío.
- Un botón "AGREGAR ALUMNO".
- Un menú desplegable "Adeudo" con "MULTA DAE" seleccionado.
- Un menú desplegable "Estatus" con "SIN PAGO" seleccionado.
- Un campo de texto "Importe".
- Un campo de texto "Descripción".
- Un botón "AGREGAR" en la parte inferior.

Figura 12.6

Captura de adeudos

El usuario deberá de ingresar el nombre del alumno realizando una búsqueda a través del botón “Agregar alumno” desplegándose una ventana de búsqueda de alumnos.

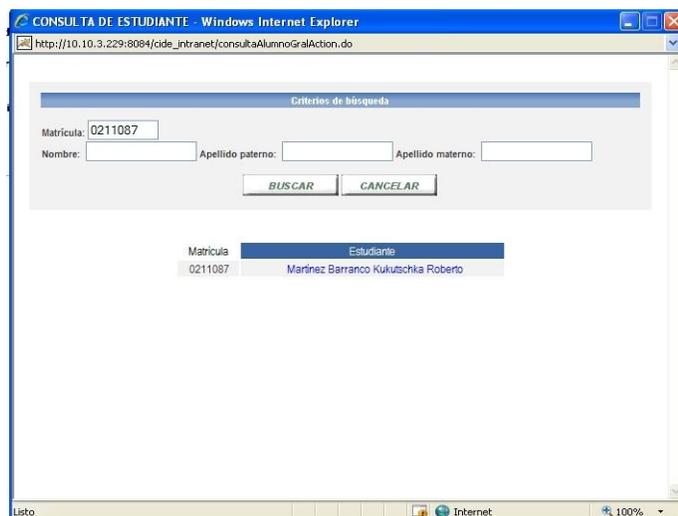


Figura 12.7

Búsqueda alumnos para registro de adeudos

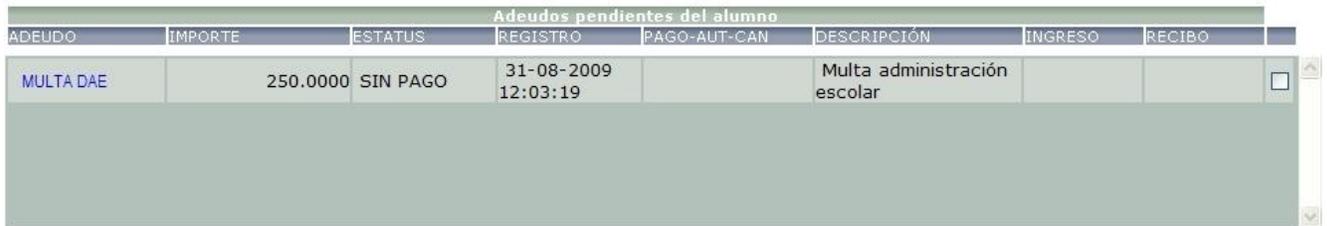
En caso de que el sistema encuentre coincidencias acordes a los criterios de búsqueda se mostrará un listado con las matriculas y nombres de alumnos. El usuario seleccionará el nombre del estudiante para el cual desea agregar un adeudo, una vez seleccionado el estudiante la ventana se cerrará, mostrando el nombre del alumno en el formulario de captura de adeudos.

The screenshot shows a form titled 'Registrar adeudo'. At the top, there is a green header bar with the text 'Registrar adeudo'. Below the header, there is a text input field containing the name 'Martínez Barranco Kukutschka Roberto' and a button labeled 'AGREGAR ALUMNO'. Below this, there are two dropdown menus: 'Adeudo' with 'MULTA DAE' selected, and 'Estatus' with 'SIN PAGO' selected. Below the 'Estatus' dropdown, there is a list of options: 'SIN PAGO' (highlighted), and 'DEBE MATERIAL O CONTRATO'. Below the dropdowns, there is an 'Importe' text input field and a 'Descripción' text input field. At the bottom of the form, there is a button labeled 'AGREGAR'.

Figura 12.8

Alumno agregado en la pantalla de adeudos

Una vez seleccionado el nombre del alumno, es necesario seleccionar el tipo de adeudo, estatus, así como ingresar el importe en caso de ser un adeudo económico, y una descripción o motivos del adeudo. En caso de que los datos ingresados sean correctos se mostrará la pantalla del detalle con el adeudo agregado.



Adeudos pendientes del alumno							
ADEUDO	IMPORTE	ESTATUS	REGISTRO	PAGO-AUT-CAN	DESCRIPCIÓN	INGRESO	RECIBO
MULTA DAE	250.0000	SIN PAGO	31-08-2009 12:03:19		Multa administración escolar		<input type="checkbox"/>

Figura 12.9

Registro de adeudos actualizado

Actualización de adeudos

La actualización de adeudos se realizará una vez que el usuario haya ingresado al detalle de adeudo. En caso de que el alumno tenga adeudos pendientes el sistema mostrará al final de la columna un cuadro de selección que permitirá realizar las siguientes actualizaciones: autorización, cancelación o entrega de material.

En la figura 13.1 se muestra el registro de un adeudo así como las opciones de actualización.

Matrícula 0211087 Nombre del alumno Martínez Barranco Kukutschka Roberto
 Generación 2002 Programa LIC. EN CIENCIA POL Y REL.INT.

Agregar Adeudo

Registrar adeudo

Adeudo MULTA BIB Estatus SIN PAGO

Importe

Descripción

AGREGAR

Adeudos pendientes del alumno

ADEUDO	IMPORTE	ESTATUS	REGISTRO	PAGO-AUT-CAN	DESCRIPCIÓN	INGRESO	RECIBO
MULTA BIB	330.0000	SIN PAGO	21-11-2008 00:00:00		MATERIAL VENCIDO 27/10/08		

AUTORIZACIÓN CANCELACIÓN MATERIAL ENTREGADO REGRESAR

Adeudos finiquitados del alumno

ADEUDO	IMPORTE	ESTATUS	REGISTRO	PAGO-AUT-CAN	DESCRIPCIÓN	INGRESO	RECIBO
MULTA BIB	50.0000	PAGADO EN CAJA	10-04-2007 00:00:00	10-05-2008 00:00:00	mat. venc. 05/03/2007	65204	40124
COLEGIATURA	5500.0000	PAGADO EN CAJA	12-08-2005 00:00:00	12-08-2005 00:00:00	PAGO 1		
COLEGIATURA	8000.0000	PAGADO EN CAJA	12-08-2005 00:00:00	12-08-2005 00:00:00	PAGO 2		
COLEGIATURA	8147.0000	PAGADO EN CAJA	12-08-2005 00:00:00	12-08-2005 00:00:00	ADEUDO	7729	13336

REGRESAR

Figura 13.1

Adeudo multa de biblioteca

Una vez seleccionado el adeudo que se desea modificar, es necesario presionar el botón que corresponda a la acción que se desea ejecutar. En cualquier caso la actualización requerirá ingresar una descripción del cambio.

Adeudos pendientes del alumno							
ADEUDO	IMPORTE	ESTATUS	REGISTRO	PAGO-AUT-CAN	DESCRIPCIÓN	INGRESO	RECIBO
MULTA DAE	300.0000	SIN PAGO	31-10-2008 09:36:13		REINCRIPCIÓN TARDÍA		<input checked="" type="checkbox"/>

Ingrese la descripción

Pago en caja CIDE

Figura 13.2

Cancelación de adeudo

En caso de haber ingresado una descripción correcta el sistema envía un mensaje de confirmación indicando el éxito de la actualización. En caso contrario el sistema mostrará una leyenda referente a la falta de descripción.



Figura 13.3

Mensaje actualización adeudo

En caso de que el adeudo haya sido cancelado, la sección de adeudos pendientes mostrará una leyenda notificando la falta de adeudos.

La figura 13.4 muestra la sección de adeudos pendientes después de cancelar el único adeudo correspondiente al área de administración escolar.

Matrícula 0411153 Nombre del alumno Martínez Mata Fabiola
 Generación 2004 Programa LIC. EN DERECHO

Agregar Adeudo

Registrar adeudo

Adeudo **MULTA DAE** Estatus SIN PAGO

Importe 250

Descripción Multa administración escolar

AGREGAR

Adeudos pendientes del alumno

ADEUDO	IMPORTE	ESTATUS	REGISTRO	PAGO-AUT-CAN	DESCRIPCIÓN	INGRESO	RECIBO
No se encontraron adeudos pendientes asociados al área de DIRECCION DE ADMINISTRACION ESCOLAR							

Adeudos finiquitados del alumno

ADEUDO	IMPORTE	ESTATUS	REGISTRO	PAGO-AUT-CAN	DESCRIPCIÓN	INGRESO	RECIBO
MULTA DAE	300.0000	CANCELADO	12-08-2005 00:00:00	12-08-2005 00:00:00	MULTA ADMINISTRACION ESCOLAR	15882	14749
MULTA DAE	400.0000	PAGADO EN CAJA	13-09-2007 10:56:43	28-01-2008 00:00:00	REINSCRIPCIÓN TARDÍA	59139	25499
MULTA DAE	250.0000	PAGADO EN CAJA	02-04-2008 15:14:30	22-08-2008 00:00:00	REINSCRIPCIÓN TARDÍA	69963	27346

REGRESAR

Figura 13.4

Sección adeudos pendientes

Reportes

El usuario podrá obtener reportes de alumnos deudores seleccionando la opción “Reportes” ubicada en el menú principal. El reporte será generado a partir de los parámetros ingresados por el usuario.

El sistema desplegará la lista de alumnos con adeudos correspondientes a la búsqueda realizada por el usuario, teniendo la posibilidad de ser exportado a una hoja de cálculo.

La figura 13.5 muestra el cuadro de diálogo para abrir o guardar el archivo generado a través de la búsqueda de alumnos deudores.

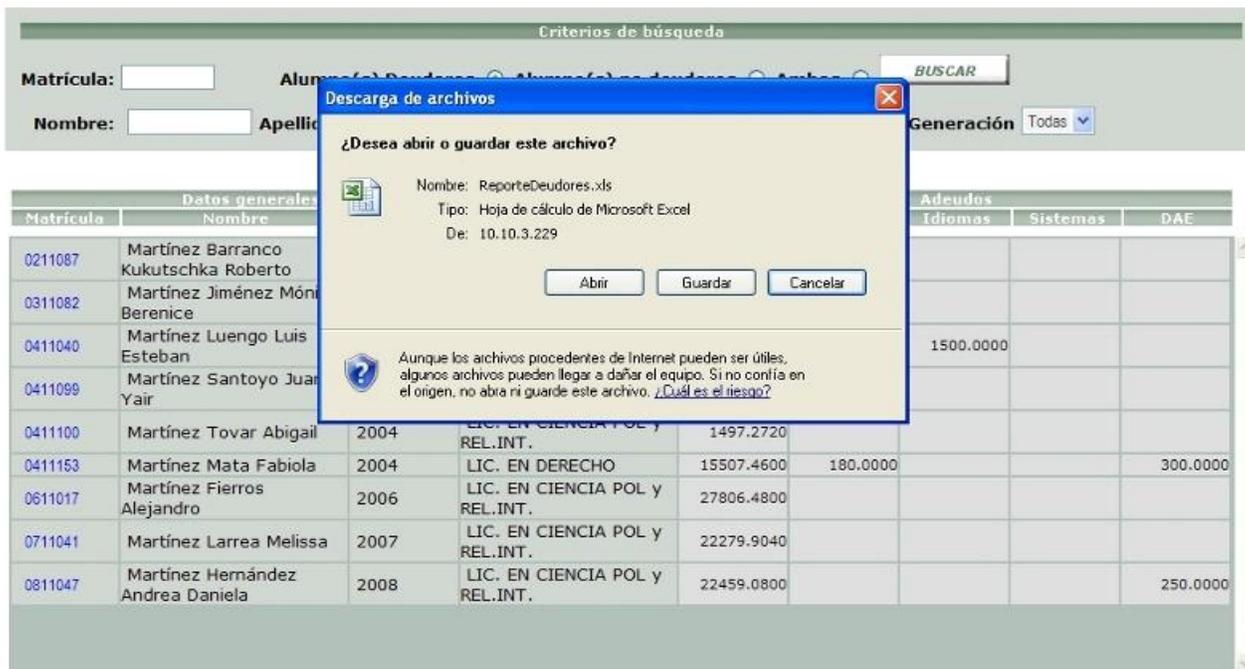


Figura 13.5

Cuadro de diálogo abrir o guardar reporte

En la figura 13.6 se muestra el reporte generado con formato de hoja de cálculo.

Matrícula	Generación	Programa	Nombre	Colegatura	Biblioteca	Idiomas	Sistemas
0211087	2009	LIC. EN CIENCIA POL Y REL. INT.	Martínez Barranco Kukutschka Roberto	1497.2720	333.0000		
0311082	2009	LIC. EN CIENCIA POL Y REL. INT.	Martínez Jiménez Mónica Berenice		41.0000		
0411040	2004	LIC. EN ECONOMÍA	Martínez Luengo Luis Esteban	3449.6534		600.0000	
0411099	2004	LIC. EN CIENCIA POL Y REL. INT.	Martínez Santoyo Juan Yair		21.0000		
0411100	2004	LIC. EN CIENCIA POL Y REL. INT.	Martínez Tovar Abigail	1497.2720			
0411153	2004	LIC. EN DERECHO	Martínez Mata Fabiola	15507.4600	180.0000		300.0000
0611017	2006	LIC. EN CIENCIA POL Y REL. INT.	Martínez Fierros Alejandro	27806.4800			
0711041	2007	LIC. EN CIENCIA POL Y REL. INT.	Martínez Larrea Melissa	22279.9040			
0811047	2008	LIC. EN CIENCIA POL Y REL. INT.	Martínez Hernández Andrea Daniela	22459.0800			250.0000

Figura 13.6

Reporte alumnos deudores

Módulo Buzón de alumnos

El módulo buzón de alumnos tiene como finalidad mantener informado al usuario de los cambios de estatus de alumnos del CIDE. En caso de existir un cambio de estatus en uno o más alumnos el buzón notificará al usuario mediante un recuadro ubicado en la parte superior del menú.

La figura 13.7 muestra el buzón de alumnos que aparece en el menú.

The screenshot shows a user interface for the 'Buzón alumnos' (Students Mailbox) module. At the top left, there is a logo for 'Centro de Investigación y Docencia Económicas A.C. CIDE'. To its right, a welcome message reads: 'Bienvenido: DANIEL OSORIO GIL', 'Último acceso: 31-08-2009 10:05:16.247', and 'Área COORDINACION DE ADMINISTRACION Y FINANZAS'. The main notification box on the right contains the text: 'Buzón alumnos', 'Licenciatura 13', 'Maestría 6', 'Doctorado 1', and 'Existen: 0 Requerimientos sin completar'. Below this, a horizontal menu contains buttons for 'Requerimientos', 'Adeudos', 'Alumnos', 'Opciones', 'Tutoriales', and 'Cerrar sesión'.

Figura 13.7

Buzón de alumnos

Los usuarios serán notificados mediante el buzón ubicado en el menú, agrupándolos acorde al nivel de estudios. Para conocer el listado de alumnos con cambio de estatus se deberá presionar el recuadro titulado “Buzón alumnos” o ingresar a través de la opción con el mismo nombre ubicada en el menú de Alumnos.

Enterado cambio estatus

Alumnos pendientes de firma cambio estatus						
Licenciatura						
Matrícula	Generación	Nombre	Estatus	Fecha mov.	Comentarios	
0811072	2008	BRUNO VIRGILIO TORRES CARBAJAL	Baja académica	07/16/09		
0811070	2008	VÍCTOR MANUEL TAPIA HERNÁNDEZ	Baja temporal	06/16/09		
0811032	2008	GABRIELA GIOVANA BARRERA FLORES	Baja con re-ingreso	07/15/09		

Maestría						
Matrícula	Generación	Nombre	Estatus	Fecha mov.	Comentarios	
0912015	2009	MIGUEL ÁNGEL MAYA CONTRERAS	Baja temporal	08/26/09		
0812041	2008	JUAN CARLOS TERRÓN VEGA	Baja académica	07/14/09		
0812035	2008	JAIME EMMANUEL LORENZANA GÓMEZ	Baja académica	07/14/09		

Doctorado						
Matrícula	Generación	Nombre	Estatus	Fecha mov.	Comentarios	
0713006	2007	BERNARDO JORGE ALMARAZ CALDERÓN	Baja temporal	08/25/09	Baja Temporal	

Enterado cambio estatus

Centro de Investigación y Docencia Económicas A.C. Bienvenido: DANIEL OSORIO GIL

Último acceso: 31-08-2009 10:05:16.247

Área: COORDINACION DE ADMINISTRACION Y FINANZAS

Buzón alumnos

Licenciatura 0

Maestría 0

Doctorado 0

Existen: **0**

Requerimientos sin completar:

Requerimientos

Adeudos

Alumnos

Opciones

Tutoriales

Cerrar sesión

Enterado cambio estatus

Alumnos pendientes de firma cambio estatus						
Licenciatura						
Matrícula	Generación	Nombre	Estatus	Fecha mov.	Comentarios	
Datos no encontrados						

Maestría						
Matrícula	Generación	Nombre	Estatus	Fecha mov.	Comentarios	
Datos no encontrados						

Doctorado						
Matrícula	Generación	Nombre	Estatus	Fecha mov.	Comentarios	
Datos no encontrados						

Enterado cambio estatus

Figura 13.8

Consulta listado buzón de alumnos

El listado de alumnos con cambio de estatus muestra en la parte superior e inferior el botón “Enterado cambio estatus”. Cuando el usuario hace clic en el botón se firma de manera electrónica, por lo tanto se asume que el usuario ha quedado enterado del cambio realizado por la Administración Escolar.

Una vez firmado el cambio de estatus, el contador ubicado en el buzón de alumnos es reiniciando a cero, sólo para el usuario que realizó la firma. La cantidad de alumnos mostrada en el buzón de alumnos variará de acuerdo a la periodicidad con la cual sea revisado.

Figura 13.9

Buzón de alumnos vacío

En la figura 13.9 se muestra el contenido vacío del buzón de alumnos después de ser firmado por el usuario.

Consulta de estatus alumnos

La consulta de estatus de alumnos se realizará al seleccionar la opción “Consulta estatus alumnos” ubicada en el menú “Alumnos”.



Figura 14.1

Opción consulta estatus alumnos

En la figura 14.2 se muestra el formulario de consulta de estatus alumnos, teniendo como opciones de búsqueda el número de matrícula, plan académico, nombre, apellido paterno, apellido materno y generación.

Figura 14.2

Búsqueda estatus alumno

En el siguiente ejemplo el usuario ingresa el número de matrícula de un alumno que desea conocer su estatus.

Matrícula	Generación	Nombre	Licenciatura	Estatus	Fecha mov.	Comentarios
0611004	2007	ABIMELEK ÁVILA MARROQUÍN		Baja académica	07/15/09	SI EL COMITÉ DE DOCENCIA LO AUTORIZA

Figura 14.3

Resultado consulta estatus alumno

Opciones

El menú de opciones contiene las opciones: cambiar contraseña y cambiar pregunta de seguridad.

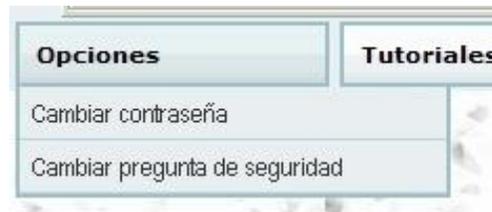


Figura 14.4

Menú opciones

Cambio de contraseña

El usuario podrá realizar el cambio de contraseña en el momento que lo decida, para ello será necesario ingresar su contraseña actual; este método es realizado para prevenir un descuido por parte del usuario al dejar una sesión abierta.

Una vez ingresada la contraseña actual, el usuario ingresará su nueva contraseña, repitiéndola en el campo mostrado en la parte inferior. En caso de que las contraseñas no coincidan el sistema notificará al usuario que debe ingresar la misma contraseña en ambos campos.

A screenshot of a web form titled 'Cambio de contraseña'. The form is contained within a light green header bar. It consists of three rows of input fields. The first row is labeled 'Ingrese su clave actual:', the second 'Ingrese su nueva clave:', and the third 'Confirmar la nueva clave:'. Each row has a text input field. Below the third input field is a 'Guardar' button.

Figura 14.5

Formulario cambio de contraseña

En la figura 14.6 se muestra el mensaje de confirmación al ingresar la contraseña actual, la clave nueva y repetirla correctamente.



Figura 14.6

Mensaje éxito actualización contraseña

Cambiar pregunta de seguridad

En caso de que el usuario quiera actualizar su pregunta de seguridad deberá ingresar a la opción “Cambiar pregunta de seguridad” ubicada en el menú de “Opciones”, la pregunta de seguridad permitirá asignar una clave temporal en caso de olvidar la clave de usuario. Esta clave temporal será enviada al correo institucional una vez respondida de manera correcta la pregunta seleccionada.

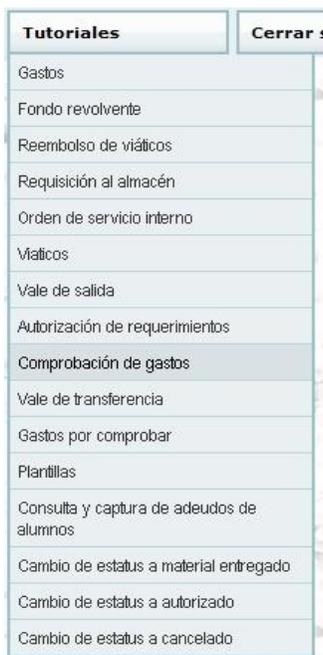
Generación de la pregunta de seguridad	
Seleccione una pregunta:	¿Cuál es tu equipo deportivo favorito? ▼
Ingrese su respuesta:	Pumas!!!
<input type="button" value="GUARDAR"/>	

Figura 14.7

Cambio de pregunta seguridad

Tutoriales

En caso de que un usuario desconozca o haya olvidado el funcionamiento del sistema podrá recurrir a la opción Tutoriales, desplegándose la lista de tutoriales contenidos dentro del sistema.



Tutoriales	Cerrar
Gastos	
Fondo revolvente	
Reembolso de viáticos	
Requisición al almacén	
Orden de servicio interno	
Viáticos	
Vale de salida	
Autorización de requerimientos	
Comprobación de gastos	
Vale de transferencia	
Gastos por comprobar	
Plantillas	
Consulta y captura de adeudos de alumnos	
Cambio de estatus a material entregado	
Cambio de estatus a autorizado	
Cambio de estatus a cancelado	

Figura 14.8

Menú tutoriales

Los video tutoriales tendrán la opción de pausar, adelantar, regresar y serán narrados por medio de cuadros de dialogo explicando cada una de las acciones realizadas.

Ejemplo: requerimiento de gastos.

Seleccione el área Llene los campos obligatorios Guarde el requerimiento Agregue los detalles

AGREGAR SALIR

NO. CURSOS: DAE/84 Ejercicio Fiscal: 2007 Fecha: 15-06-2007

Área: * DIRECCION DE ADMINISTRACION ESCOLAR * Proyecto: Propios

Quién Autoriza (Área): GUERRERO ALONSO MARY JULY

Solicita: CASAB CASTILLEJOS RUTH

Tipo de Requerimiento: * GASTOS

Concepto General: Pago de revisión de títulos

A Favor de: * Proveedor Empleado Estudiante General

* GUERRERO ALONSO MARY JULY

Divisa: PESOS 1.00 Impuesto único para todos los detalles: IVA 15%

Justificación: Pago de títulos

Observaciones: Requerimiento para pago de títulos

AGREGAR NOTAS

AGREGAR SALIR

En caso de que requiera anexar alguna nota al requerimiento, presione este botón.

Figura 14.9

Tutorial requerimiento gastos

16.3 Administración del sistema

El propósito del siguiente apartado es enlistar los archivos de configuración, así como describir de una manera breve el esquema de base de datos incluyendo algunas de las tablas más importantes dentro del sistema.

16.4 Archivos de configuración

Los archivos de configuración se encargan de establecer la funcionalidad del sistema acorde al contexto del proyecto, esto significa que dependiendo del entorno (producción o desarrollo) la configuración cambiará.

El archivo `web.xml` define el Servlet *ActionServlet* componente esencial del *Framework* Struts, definido como la representación del Controlador desde la perspectiva del patrón de diseño MVC.

El parámetro *config* define la localización del archivo que define el flujo de la aplicación, por omisión toma la ruta `/WEB-INF/struts-config.xml`

Al final del archivo se declaran las librerías de Tags utilizadas dentro de los archivos JSP.

El archivo `struts-config.xml` se encuentra dividido mediante los siguientes tags:

`<form-beans>` Dentro de este apartado se definen los Java Beans utilizados dentro de la aplicación, la definición se encuentra compuesta por el nombre de la clase Java, así como un nombre corto o de referencia utilizado como propiedad de tipo “*name*” en la sección `<action-mappings>`

`<action-mappings>` En esta sección se establece el flujo del sistema, se especifican las acciones solicitadas por el cliente, así como la clase que atenderá dicha petición y los diferentes archivos enviados como respuesta a partir del procesamiento de la petición realizada por el cliente.

ApplicationResource.properties: En este archivo se almacenan las etiquetas que serán desplegadas en caso de existir un error al validar un formulario.

MSICIDEConfig.properties: Archivo de especificación para el driver de la base de datos y parámetros de conexión.

Log4j.properties: Archivo de configuración para envío de mensajes de salida al ejecutar la aplicación dentro del servidor Web.

17. Esquema de la base de datos

En la figura 15.1 se muestra el diagrama entidad-relación de algunas tablas del sistema.

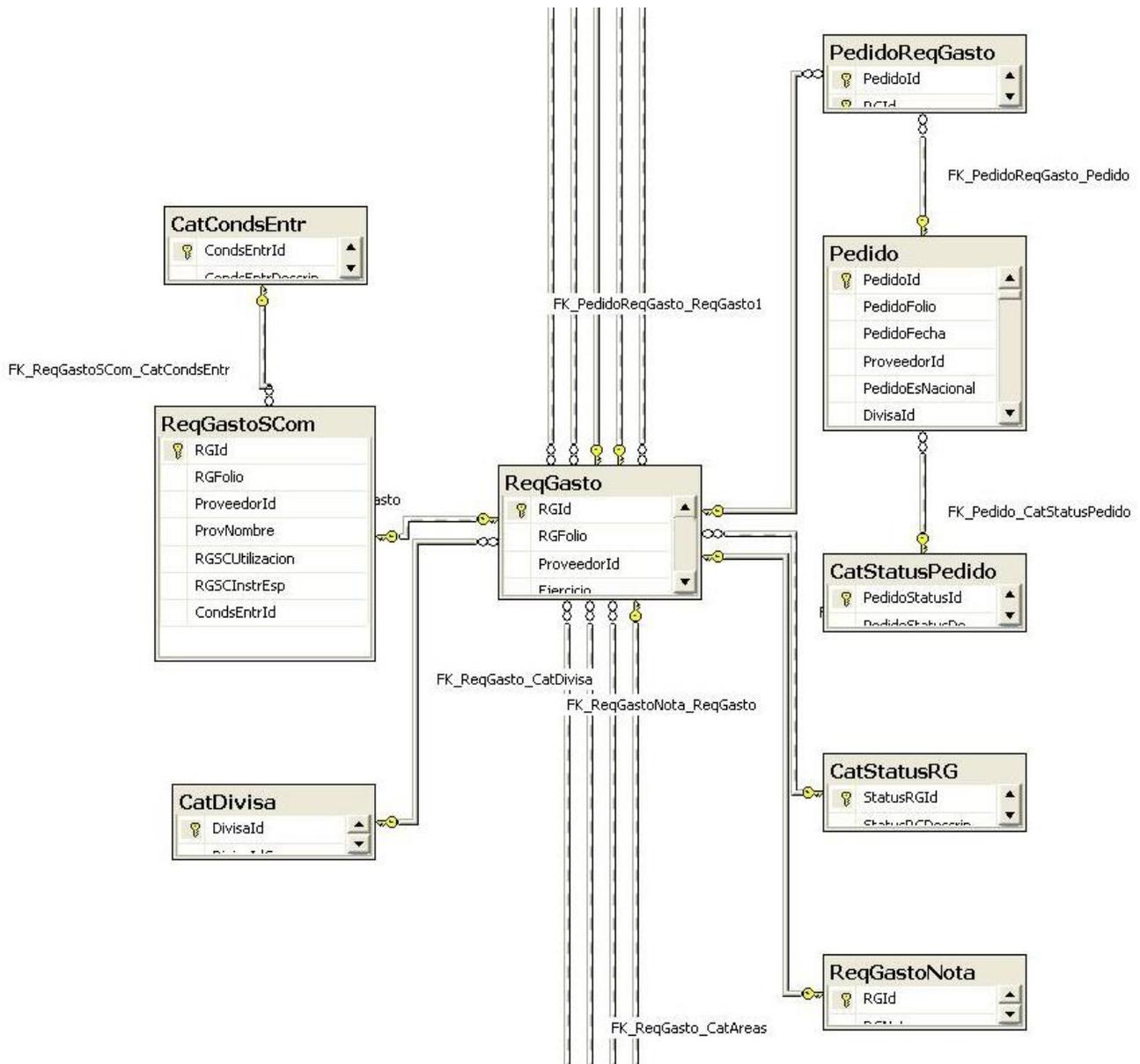


Figura 15.1

Diagrama entidad relación tablas del sistema

Tabla: ReqGasto

Descripción: Tabla de almacenamiento requerimientos.

Modulo: Requerimientos

Nombre del Campo	Tipo de dato	Longitud	Descripción del dato almacenado
RGId	int	8	Identificador único del requerimiento
RGFolio	varchar	15	Folio del requerimiento
ProveedorId	smallint	5	Identificador del proveedor del requerimiento
Ejercicio	smallint	5	Año del ejercicio fiscal
ReqGastoTipoId	tinyint	3	Identificador del tipo de requerimiento
RGFechaReg	datetime	8	Fecha de registro requerimiento
RGAFavorDe	varchar	120	Nombre del área o persona favorecida con el requerimiento.
RGConceptoGral	varchar	300	Concepto del requerimiento
RGJustificacion	varchar	500	Justificación del requerimiento
RGEsNacional	bit	1	Compra nacional o extranjera
ImpuestoId	tinyint	3	Identificador tipo de impuesto
DivisaId	int	4	Identificador tipo de divisa
RGSumaParcial	money	8	Suma parcial detalles del requerimiento
RGDescuento	decimal	9	Descuento aplicable al total del requerimiento
RGSubtot	money	8	Subtotal detalle requerimiento
RGImporteTotal	money	8	Importe total detalles del requerimiento
RGImporteImpuesto	money	8	Importe cálculo impuestos requerimiento

RGPrioridad	bit	1	Identificador estatus prioridad
RGSolicita	varchar	50	Nombre de la persona solicitante del requerimiento
MontoActId	tinyint	3	Identificador monto de actuación
RGFirElecArea	smallint	5	Identificador de usuario firma electrónica
RGFirElecAdqMat	smallint	5	Identificador de usuario firma electrónica área de adquisición materiales
RGFirElecPr	smallint	5	Identificador de usuario firma electrónica área de presupuesto
RGFirElecAreaStatus	smallint	5	Identificador de usuario firma electrónica estatus área solicitante
RGFirElecCRFM	smallint	5	Identificador de usuario firma electrónica Coordinación de Recursos Financieros y Materiales
RGFirElecDA	smallint	5	Identificador de usuario firma electrónica
RGAsociadoRG	int	4	Folio asociado del requerimiento
RGCotizacion	varchar	20	Cotización de la compra
StatusRGId	tinyint	3	Estatus del requerimiento
AreaId	varchar	4	Abreviación área solicitante del requerimiento
RGObs	varchar	500	Observaciones del requerimiento
RGRechazado	bit	1	Identificador requerimiento rechazado
RGCausaRechazo	varchar	255	Descripción motivos del rechazo
RGFirmasFisicas	bit	1	Identificador firma de requerimiento
AreaCIId	int	4	Identificador área solicitante
AFavorDeId	int	4	Identificador área o persona favorecida con el requerimiento
AFavorDeTipo	tinyint	3	Identificador del tipo de área favorecida

Tabla: CatDivisa

Descripción: Catálogo de divisas y valor de cambio.

Módulo: Requerimientos

Nombre del campo	Tipo de dato	Longitud	Descripción del dato almacenado
DivisaId	int	4	Identificador id tabla Catálogo divisas
DivisaIdG	int	4	Identificador divisa
DivisaTipoCambio	decimal	9	Tipo de cambio de la divisa
DivisaFecha	datetime	8	Fecha del tipo de cambio

Tabla: ReqGastoDetalle

Descripción: Tabla detalle del requerimiento

Módulo: Requerimientos

Nombre del campo	Tipo de dato	Longitud	Descripción del dato almacenado
RGDId	int	4	Identificador detalle requerimiento
RGId	int	4	Identificador requerimiento
FuenFinId	tinyint	3	Identificador fuente de financiamiento
PartidaId	smallint	5	Identificador partida presupuestal
ActInstId	smallint	5	Identificador actividad institucional
RGDDescrip	varchar	620	Descripción detalle
ProveedorId	smallint	3	Identificador proveedor
GiroId	tinyint	3	Identificador giro proveedor
UnidadId	smallint	3	Identificador tipo de unidad detalle
RGDCantidad	decimal	9	Cantidad solicitada

RGDPrecioUni	decimal	9	Precio unidad
ImpuestoId	tinyint	3	Identificador de impuesto
RGDPorEntregar	decimal	9	Número de piezas por entregar
RGDImporte	money	8	Importe total
RGDDescuento	decimal	9	Descuento
RGDImporteImpuesto	money	8	Importe del impuesto
RGDUltFechaModif	datetime	8	Fecha de modificación
RGDImporteImpuestoReal	decimal	9	Impuesto real
RGDImporteReal	decimal	9	Importe real
MoviCajaId	int	4	Identificador pago en caja

Tabla: Usuarios

Descripción: Tabla de usuarios del sistema.

Módulo: Requerimientos, adeudos.

Nombre del campo	Tipo de dato	Longitud	Descripción del dato almacenado
UsuarioId	smallint	5	Identificador de usuario
UsuarioCuenta	varchar	20	Nombre de cuenta usuario
EmpleadoId	int	4	Identificador número de empleado
UsuarioNombre	varchar	100	Nombre de usuario
UsuarioClave	varchar	20	Clave de usuario
AreaId	varchar	4	Identificador de área
UsuarioEmail	varchar	60	Correo electrónico del usuario
FirmaElectronica	bit	1	Autorización firma electrónica

Iniciales	varchar	5	Iniciales área usuario
NivelVisible	int	4	Nivel visibilidad
AreaCIId	int	4	Identificador área del usuario
Puesto	varchar	250	Puesto
Fotografia	varchar	100	Ruta archivo fotografía
Fec_cuenta	datetime	8	Fecha de alta cuenta
imagen	image		Imagen usuario

Tabla: UsuariosAreas

Descripción: Tabla de áreas y permiso de visibilidad usuarios.

Modulo: Requerimientos, adeudos.

Nombre del campo	Tipo de dato	Longitud	Descripción del dato almacenado
UsuarioId	int	4	Identificador usuario
AreaCID	int	4	Identificador de área
NivelVisible	int	4	Nivel de acceso usuario

Tabla: Deudores

Descripción: Vista de alumnos deudores

Módulo: Adeudos

Nombre del campo	Tipo de dato	Longitud	Descripción del dato almacenado
AdeudoId	int	4	Identificador del adeudo
MatriculaId	int	4	Identificador matricula del alumno
cve_adeu	smallint	5	Clave del adeudo
importe	money	8	Importe del adeudo
sta_adeu	smallint	5	Estatus adeudo
descripcion	varchar	200	Descripción del adeudo
MoviCajaId	int	4	Identificador pago adeudo caja
fec_reg	datetime	8	Fecha registro adeudo
fec_pago	datetime	8	Fecha pago adeudo

Tabla: CatTipoDeuda

Descripción: Catálogo tipos de adeudo

Módulo: Adeudos

Nombre del campo	Tipo de dato	Longitud	Descripción del dato almacenado
cve_adeu	smallint	5	Clave adeudo
desc_adeu	varchar	30	Descripción adeudo
AreaCId	int	4	Identificador área

Tabla: Alumnos

Descripción: Registro de alumnos del CIDE

Módulo: Requerimientos, adeudos, buzón de alumnos.

Nombre del campo	Tipo de dato	Longitud	Descripción del dato almacenado
MatriculaId	int	4	Identificador de la matrícula
Matricula	varchar	20	Número de matrícula
ClaveGeneracion	varchar	8	Clave de generación
Programa	varchar	100	Programa académico
Grupo	varchar	255	Grupo actual del alumno
ApellidoPaterno	varchar	25	Apellido paterno
ApellidoMaterno	varchar	25	Apellido materno
Nombre	varchar	100	Nombre del alumno
Becario	int	4	Identificador becario
Activo	bit	1	Estatus del alumno
EdoCivil	varchar	15	Estado civil
RFC	varchar	15	RFC
CURP	char	20	CURP
FechaAlta	datetime	8	Fecha de alta
FechaNacimiento	datetime	8	Fecha de nacimiento
Sexo	varchar	1	Sexo
Pais	varchar	25	País de origen
Entidad	varchar	30	Entidad

Municipio	varchar	30	Municipio
Calle	varchar	100	Calle domicilio particular
Colonia	varchar	100	Colonia domicilio
CodigoPostal	varchar	10	Código postal
Telefono	varchar	150	Teléfono particular
CalleProvincia	varchar	100	Calle domicilio estudiante foráneo
ColoniaProvincia	varchar	100	Colonia domicilio estudiante foráneo
PaisDomProv	varchar	25	País domicilio estudiante foráneo
EntidadProv	varchar	25	Entidad estudiante foráneo
MunicipioProv	varchar	30	Municipio estudiante foráneo
Gentilicio	varchar	25	Gentilicio
CodigoPostalProv	char	10	CP estudiante foráneo
TelProv	varchar	100	Teléfono domicilio estudiante foraneo
Email	varchar	100	Dirección de correo electrónico
Edad	tinyint	3	Edad
Escuela	varchar	600	Escuela de procedencia
Estatus	varchar	18	Estatus alumno
NoSemestre	int	4	Número de semestre
Semestre	varchar	255	Semestre
MunicipioId	smallint	5	Identificador del municipio
folio	int	4	Número de folio solicitud electrónica

Tabla: buzónAlumnos, buzónAlumnosMaestria, buzónAlumnosDoctorado

Descripción: Tabla registro usuarios buzón de alumnos licenciatura.

Módulo: Buzón de alumnos.

Nombre del campo	Tipo de dato	Longitud	Descripción del dato almacenado
id	int	4	Identificador de tabla
folio	varchar	20	Folio de alumno
usuarioId	varchar	255	Identificador usuario registro buzón alumnos
fechaRegistro	datetime	8	Fecha registro del cambio

CIDE INTRANET

Conclusiones

18. Conclusiones

El presente trabajo concluye a la par de un gran aprendizaje. Si bien la universidad nos presenta y establece las bases para la integración laboral, resulta difícil un acoplamiento perfecto con la curva de aprendizaje, es por tal motivo que, con el paso del tiempo y la inclusión en el ámbito laboral se pone en práctica el conocimiento adquirido durante la formación académica y se desarrollan nuevas habilidades propias del ambiente, sin embargo, el proceso suele ser complicado en un inicio.

El desarrollo de CIDE INTRANET implicó un esfuerzo continuo a nivel personal y colectivo, comenzando por comprender y conocer los motivos que justificarían el desarrollo de un sistema nuevo que reemplazaría el anterior. Después de recabar la opinión de los usuarios, se determinó la viabilidad del proyecto, iniciando el análisis del código fuente y base de datos, etapa en la cual se replicó el entorno de desarrollo utilizado por el equipo de SIGCIDE, ésta etapa fue en particular difícil. Al ser el inicio del proyecto no se habían definido de manera clara los objetivos, ni se contaba con el apoyo del equipo de desarrollo de SIGCIDE.

Una vez analizado el código y la base de datos, se evaluó la cantidad de usuarios y módulos sobre los cuales se tendría un mayor impacto, determinando sólo desarrollar el módulo de requerimientos y alumnos deudores; la elección se tomó en conjunto con el equipo de desarrollo de CIDE INTRANET, considerando que, migrar un sistema completo con más de cinco años de desarrollo y módulos muy específicos utilizados por un porcentaje muy bajo de usuarios, representaría un gasto de recursos muy significativo.

Después de definir los alcances del sistema la siguiente tarea fue elegir la arquitectura acorde al tipo de requerimientos y, a la infraestructura proporcionada por el CIDE; en esta etapa los integrantes del equipo de desarrollo expusimos ventajas y desventajas de arquitecturas, metodologías y componentes a utilizar para el desarrollo de CIDE INTRANET, llegando a un acuerdo común sobre la arquitectura, metodología y lenguaje de programación que serían utilizados.

En la definición de arquitectura aprendí que, la etapa de análisis a pesar de no representar una línea de código funcional, es un factor decisivo para el éxito o fracaso de todo proyecto, resultando de suma importancia conocer y manejar un amplio espectro de

tecnologías y metodologías, con la finalidad de implementar aquella que sea la más viable de acuerdo a las necesidades del proyecto, infraestructura, costos, o tiempo estimado para la implementación. Cada uno de los factores mencionados tienen una proporción en el proyecto, en caso de descuidar uno o más aspectos, los defectos en cada etapa se acarrearán en las etapas subsecuentes, concluyendo en un cúmulo de errores que afectarán aspectos como: rendimiento, funcionamiento o escalabilidad del sistema.

Por tales razones, es indispensable hacer un análisis acorde al tipo y dimensiones del proyecto que se realizará, por lo tanto, es necesario utilizar una metodología específica, y para ello fue necesario evaluar las siguientes metodologías: *Programación Extrema*, *Espiral*, *Cascada*, *RUP*, cada una compuesta por diferentes fases, algunas más robustas o centradas en aspectos de menor análisis, pero mayor énfasis en etapas de implementación y pruebas.

La metodología RUP es realizada de manera iterativa, cada fase se realiza hasta completar la anterior, incluyendo la revisión del artefacto producido, y evitando la propagación de errores en las etapas subsecuentes del desarrollo. Otro punto de vital importancia en la metodología RUP es el modelado de negocios, el cual representa los problemas actuales de la organización e identifica las áreas de mejora potencial. El modelado de negocios permite garantizar la comprensión común entre los clientes, usuarios finales y desarrolladores, asegurando el preciso entendimiento de las funciones, responsabilidades e interacciones entre los elementos que constituyen la organización.

En el caso de RUP el modelado es realizado haciendo uso de UML, representando la funcionalidad del sistema a través de la mayor abstracción posible en diagramas de estado, secuencia y casos de uso. Los casos de uso deberán encajar en la arquitectura de software, permitiendo la implementación requerida actualmente y en el futuro.

Otra característica esencial dentro de la metodología RUP es el énfasis realizado en la Arquitectura de Software, etapa en la que se involucran factores estáticos y dinámicos, los más significativos del sistema; se definen aspectos estructurales y arquitectónicos acordes al análisis de sistemas, así como la consideración de elementos relacionados con el aseguramiento de calidad en diversas estructuras del sistema.

Una vez que conocí de manera detallada las características, ventajas y desventajas de la

metodología RUP, en conjunto con el equipo de desarrollo y de común acuerdo con el director de área responsable del proyecto, decidimos emplearla con la finalidad de establecer un medio de comunicación entre los usuarios y el equipo de desarrollo, así como evitar errores durante las fases del proyecto.

Después de definir los alcances del proyecto y la metodología, se continuó con la asignación de tareas. En esta etapa se asignaron actividades con base en la experiencia y capacidades de cada integrante del equipo, lo cual permitió un desarrollo rápido y eficiente.

En la etapa de implementación se obtuvieron comentarios favorables y otros no tanto, las respuestas favorables provinieron -en su mayoría- de usuarios nuevos o con poco tiempo de uso en SIGCIDE, los comentarios adversos fueron realizados por usuarios totalmente acostumbrados al sistema anterior, por lo tanto, este último grupo recibió mayor atención para realizar la transición de un sistema a otro.

Una vez liberado el sistema se han realizado actualizaciones, principalmente al módulo de requerimientos, esta serie de actualizaciones me ha permitido corregir, automatizar y mejorar la funcionalidad del sistema, aumentando el rendimiento y mejorando el aspecto del sistema, además, me permite detectar y evitar errores similares en proyectos subsecuentes.

Como profesional, el análisis diseño e implementación de CIDE INTRANET significó un gran aprendizaje que implicó la puesta en marcha del conocimiento adquirido durante mi etapa académica, por otro lado, comprendí y desarrollé nuevas habilidades, que si bien fueron adquiridas de manera empírica, en esta ocasión tuve la oportunidad de explotarlas a nivel profesional, como lo fue en el modelado en UML y diseño de sistemas modulares.

Este proyecto culmina con el funcionamiento de CIDE INTRANET, obteniendo una opinión favorable por parte de la mayoría de usuarios, también se cumplió con el objetivo de facilitar sus labores y tener un sistema independiente de la plataforma, adecuado a la infraestructura y con un diseño agradable para el usuario, todo ello, considerado como los objetivos planteados en un inicio, y cumplidos mediante el análisis, diseño e implementación de la metodología RUP.

19. Glosario

Abstracción: Operación mental destinada a aislar conceptualmente una propiedad concreta de un objeto, ignorando otras propiedades del objeto.

Actor: Representación gráfica de la interacción entre usuarios y funcionalidad del sistema.

Artefacto: Documento producido durante las etapas de desarrollo de software; contiene información referente al modelado de casos de uso, definición de arquitectura de software, etcétera.

Auditabilidad: Examen, verificación o demostración de un sistema computacional.

Componente: Elemento del sistema encargado de ejecutar alguna tarea dentro del proceso.

Contingencia: Acto o hecho no presente en el funcionamiento habitual del sistema.

Control de versiones: Se denomina control de versiones a la gestión de los cambios realizados en el código fuente de un sistema.

Caso de uso: Secuencia de actividades desarrolladas entre un sistema y usuarios.

Debian GNU/LINUX: Sistema operativo de libre distribución.

Disciplina: Conjunto de actividades que tienen como objetivo desglosar uno o más procesos de relativa importancia.

Disparador (trigger): Procedimiento almacenado que se ejecuta al cumplirse cierta condición establecida.

Disponibilidad: Continuidad operacional de un software (tiempo de funcionamiento sin interrupciones).

Escalabilidad: Propiedad que permite extender la funcionalidad del sistema sin afectar el desempeño.

Excepción: Conjunto de eventos disparados por un funcionamiento alterno.

Extensión: Esta relación significa que un caso de uso puede estar basado en otro con menor funcionalidad.

Framework: Estructura de software mediante la cual otro proyecto de software es organizado y desarrollado, incluye soporte de librerías y otros componentes con la finalidad de obtener una integración estándar.

Hito: Hecho clave y fundamental dentro de un ámbito o contexto.

HTML: Siglas de HyperText Markup Language (*Lenguaje de Marcas de Hipertexto*) lenguaje de marcado para la construcción de páginas Web.

HTTP: *HyperText Transfer Protocol* (Protocolo de Transferencia de Hipertexto), por sus siglas en inglés.

Implementación: En esta etapa el sistema es puesto en marcha para uso común.

Inclusión: Descomposición funcional de un caso de uso.

Interfaz: Medio de comunicación entre el usuario y un sistema computacional.

Intranet: Red de computadoras conectadas entre sí, y de acceso controlado.

Ítem: Elemento de configuración perteneciente al proyecto.

Iteración: Sucesión de actividades dentro de un sistema.

Java: Lenguaje de programación multiplataforma desarrollado por la empresa Sun Micro Systems.

JEE: Abreviatura de Java Enterprise Edition, plataforma de programación para desarrollar arquitecturas de N niveles, empleando el lenguaje de programación Java.

JSP: Tecnología Java que permite generar contenido dinámico para web en forma de documentos HTML, XML u otro tipo.

Librerías: Conjunto de subprogramas utilizados para desarrollar software, normalmente forman parte del compilador.

Mantenibilidad: Propiedad en la cual se representa el esfuerzo requerido para conservar el funcionamiento, o restablecerlo en caso de ser necesario.

Multiplataforma: Sistema capaz de ejecutarse en diferentes plataformas.

MVC: *Modelo Vista Controlador*, patrón de arquitectura de software que separa los datos, interfaz de usuario y lógica de control.

Patrón de diseño: Esquema utilizado para definir estructuras de diseño computacional.

Performance: Rendimiento obtenido al ejecutar un sistema de cómputo.

Plataforma: Conjunto de hardware y software que puede ejecutar o desarrollar software.

POO: Abreviación de *Programación Orientada a Objetos*.

Portabilidad: Capacidad de un software para ejecutarse en diferentes plataformas.

Procedimiento almacenado: Programa almacenado físicamente en la base de datos, el cual al ser ejecutado actúa directamente sobre los datos almacenados.

Prototipo: Representación no funcional de un sistema computacional.

Restricciones: Limitaciones derivadas de las tecnologías disponibles para implementar sistemas de información.

Rol: Funciones del usuario dentro de un sistema computacional.

RUP: Rational Unified Process (*Proceso Unificado Racional*), es la metodología estándar utilizada para realizar análisis, desarrollo y documentación de sistemas informáticos.

Sistema de Información Gerencial (SIGCIDE): Sistema con el que actualmente se crean los requerimientos en el CIDE. Se encuentra desarrollado en Visual Basic.

Sistema de Gestión de Base de Datos (SGBD): En inglés: (*DataBase Managment System abreviado DBMS*) es una interfaz de software entre la base de datos, el conjunto de aplicaciones y el usuario. Para este desarrollo se empleará SQL Server en su versión 2008.

Subversión: Software de sistema de control de versiones.

Servidor: Equipo encargado de ofrecer servicios a un conjunto de usuarios.

Struts framework: Herramienta de desarrollo de aplicaciones Web bajo el patrón MVC y JEE.

Tags: Conjunto de etiquetas utilizadas para realizar operaciones comunes dentro de los archivos JSP.

TCP/IP: Protocolo de red que permite la transmisión de datos entre computadoras conectadas entre sí.

Tomcat: Contenedor de servlets y Java Server Pages (JSP).

Transacción: Conjunto de ordenes ejecutadas en un Sistema de Gestión de Base de Datos (SGBD).

UML: Unified Modeling Language (*Lenguaje Unificado de Modelado*), es el lenguaje gráfico utilizado para especificar o describir procesos dentro de un sistema.

Versión: En el ámbito del desarrollo de software una versión es una representación numérica que indica el grado de desarrollo de un programa. La versión es comúnmente representada por dos dígitos separados por un punto, el cambio significativo de una versión ocurre cuando existe un cambio en el dígito de la izquierda.

WAR: Web Archive Resource

XML: siglas de Extensible Markup Language (*Lenguaje de Marcas Extensible*). Es un lenguaje estándar para el intercambio de información estructurada entre diferentes plataformas.

20. Bibliografía

- ✓ Cekvenich Vic; *Struts Fast Track: J2EE/JSP Framework*, BaseBeans Engineering, 2001, pp. 320
- ✓ Eeles, P., Houston, K., Kozaczynski, W., *Building J2EE Applications with the Rational Unified Process*, Addison Wesley, 2002, pp. 288
- ✓ Falgueras, C., Benet, *Ingeniería del software*, UOC, 2003, pp.320
- ✓ Favre, Liliana; *UML and the unified process*, IRM, 2003, pp.402
- ✓ Forta, Ben., *JavaServer Pages application development*, Sams, 2000, pp.398
- ✓ Husted T., Dumoulin C., Franciscus G, *Struts in Action*, Manning Publications, 2002, pp. 672
- ✓ Jacobson, I., Booch, G., Rumbaugh J., *El proceso Unificado de desarrollo de Software*, Addison Wesley, 2000.
- ✓ Kruchten, Philippe; *The Rational Unified Process: An Introduction*, Addison Wesley, 2000, segunda edición, pp.320
- ✓ Moodie, Matthew; *Pro Apache Tomcat 6*, Apress, 2007, pp. 325
- ✓ Schch, Stephen R., *Análisis y diseño orientado a objetos con UML y el proceso unificado*, 2005, pp.458

21. Fuentes de Información Digital (Internet)

- ✓ Jean Louis Maréchaux, Developing a J2EE Architecture with Rational Software Architect using the Rational Unified Process®, [En línea], EE.UU., IBM, 16 de agosto de 2005, Dirección URL: http://www-128.ibm.com/developerworks/rational/library/05/0816_Louis/, [Consulta: 7 de abril de 2009].

- ✓ s/Autor, IBM Rational Software Architect, [En línea], EE.UU., IBM, s/Fecha, Dirección URL: <http://www-306.ibm.com/software/awdtools/architect/swarchitect/index.html>, [Consulta: 26 de noviembre de 2008].

- ✓ s/Autor, The IBM Rational Unified Process, [En línea], EE.UU., s/Fecha, IBM, Dirección URL: <http://www-306.ibm.com/software/awdtools/rup/index.html>, [Consulta: 10 de enero de 2009]

- ✓ Philippe Kruchten, The “4+1” view model of software architecture, [En línea], EE.UU., IBM, Noviembre 1995, Dirección URL: <http://www3.software.ibm.com/ibmdl/pub/software/rational/web/whitepapers/2003/Pbk4p1.pdf>, [Consulta: 12 de septiembre de 2009]