

UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

FACULTAD 9



Título: *“Subsistema Docente de la Facultad 9”.*

**“Trabajo de Diploma para optar por el título de
Ingeniero en Ciencias Informáticas”**

Autores:

Yudelis Delgado Sanabria

Lidiris Vega Gómez

Tutor: Ing. Aniuska Grotestan Columbié.

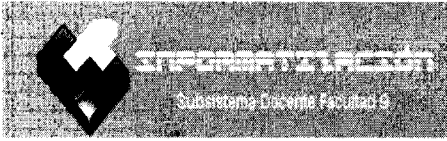
Co-Tutor: Ing. Yoandris Silverio Pacheco.

“Año 50 de la Revolución”

Ciudad de La Habana 4 de Julio 2008

...“Hay una fuerza motriz más poderosa que el vapor, la electricidad y la energía atómica: la voluntad...”

“Albert Einstein”



DEDICATORIA

A mis padres por tanto amor y dedicación.

A mi hermana por su confianza y cariño.

A mi novio por estar a mi lado en todo momento.

Yudelis

A mi abuelo Vicente por haber sido siempre el pilar que sostenía nuestra familia.

A mi mamá por tantos años de dedicación a mi hermano y a mí.

A mi novio por todo su amor y comprensión.

A mi hermano Víctor Lidier.

Lidiris



AGRADECIMIENTOS

De Yudelis

A mi padres, por ser mis guías, por educarme tan bien, por el cariño, la dedicación, por tanto amor, y porque a ellos le debo lo que soy.

A mi hermana, por estar siempre junto a mí, por ser la hermana más buena del mundo, y brindarme su apoyo y ayuda.

A mi sobrino, por existir y brindarme su cariño en los momentos más difíciles.

A quien es parte de mi en todo momento, mi novio, por su espera, amor y compañía.

A mi cuñado Julito por estar siempre dispuesto.

A toda mi familia en general, que aunque algunos estuvieron lejos de una forma u otra siempre me dieron su apoyo.

A toda la familia de mi novio por haberme dado la oportunidad de ser parte de ella.

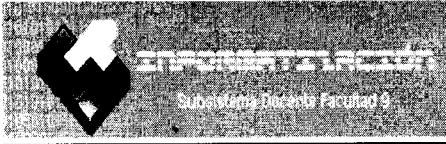
A mi tutora y cotutor por estar siempre dispuestos, sin importar la hora, el momento y el lugar.

A mis compañeros de aula, por permitirme compartir tantos momentos lindos a lo largo de estos cinco años de carrera, en especial a Lidiris, Dayamí y Dalila.

A mis otros compañeros de la UCI, en especial a Karelys, Yaniris y Maybel por su compañía, ayuda y apoyo en todo momento.

A todos aquellos que de una u otra forma me ayudaron en la realización de este trabajo, en especial a los trabajadores de la Secretaría Docente y el Vicedecanato de Formación de la facultad.

Y muy especialmente a la Revolución Cubana por haberme dado la oportunidad de estudiar en una Universidad de excelencia como lo es esta y formarme como Ingeniera en Ciencias Informáticas.



AGRADECIMIENTOS

De Lidiris

Quiero agradecer muy especialmente a:

Mi mamá por haberme dado siempre todo, amor, comprensión, apoyo incondicional.

Mi novio por todo su amor y comprensión y por haber estado siempre a mi lado, en los buenos y en los malos momentos.

Mi hermano Víctor Lidier, deseando que cumplas tu sueño como yo estoy cumpliendo el mío.

La Revolución Cubana por haberme dado la posibilidad de estudiar y de cumplir mi sueño de ser Ingeniera en Ciencias Informáticas.

Agradecer además a:

Mis hermanos Mailyn y Víctor Manuel.

Mi familia toda, por su cariño y preocupación.

La familia de mi novio, especialmente a mi suegra por quererme y preocuparse por mí como si fuera su hija.

Mis profesores por enseñarme todo lo que sé.

Mis amigos de ayer, de hoy y los amigos de toda la vida, especialmente a Yudelis, Dalila y a Dayamí por su apoyo en los días más difíciles de estos años.

Mis compañeros de tantas aulas, especialmente a los del Grupo 11 del IPVCE.

Los trabajadores de la brigada de construcción, especialmente a los trabajadores de Mataguá y muy especialmente a Juan Aguada por haberme tratado y ayudado como a una hija.

Mi tutora y co-tutor por su guía y ayuda en la realización de este trabajo.

Las trabajadoras de la Secretaría Docente y el Vicedecanato de Formación por brindarnos su ayuda.

En fin, a todas las personas que me ayudaron en la realización de este trabajo así como a lo largo de toda mi vida.



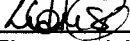
DECLARACIÓN DE AUTORÍA

DECLARACIÓN DE AUTORÍA


Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste se firma la presente a los 7 días del mes de julio del año 2008.

Lidiris Vega Gómez


Firma del Autor

Yudelis Delgado Sanabria


Firma del Autor

Ing. Aniuska Grotestan


Firma del Tutor

Ing. Yoandris Silverio


Firma del Co-Tutor



OPINIÓN DEL USUARIO

OPINIÓN DEL USUARIO DEL TRABAJO DE DIPLOMA

El Trabajo de Diploma titulado "Subsistema Docente de la Facultad 9", fue desarrollado en la Universidad de las Ciencias Informáticas. Esta entidad considera que, en correspondencia con los objetivos trazados, el trabajo realizado le satisface:

Totalmente ____

Parcialmente en un ____ %

Los resultados de este Trabajo de Diploma le reportan a esta entidad los beneficios siguientes:

Y para que así conste, se firma la presente a los ____ días del mes de _____ del año _____

Representante de la Entidad

Cargo

Firma

Cuño



OPINIÓN DEL TUTOR

OPINIÓN DEL TUTOR

Título: Subsistema Docente de la Facultad 9.

Autores: Yudelis Delgado Sanabria.

Lidiris Vega Gómez.

Por todo lo anteriormente expresado considero que los estudiantes están aptos para ejercer como Ingenieros en Ciencias Informáticas; y propongo que se le otorgue al Trabajo de Diploma la calificación de _____.

Aniuska Grotestan Columbié

Firma

Fecha



RESUMEN

La gestión de la información se ha convertido en un proceso importante en las instituciones y empresas, debido a que esta permite organizar y poner en uso los recursos de información de la organización, esto ha permitido un progresivo desarrollo tecnológico, que posibilita que las instituciones desarrollen sistemas que le permitan gestionar grandes volúmenes de información. En la Facultad 9 de la Universidad de las Ciencias Informáticas (UCI), los procesos que se realizan en el Vicedecanato de Formación, la Planificación Docente y en la Secretaría Docente no están automatizados, el gran volumen de información que se maneja se hace manualmente, los datos son recogidos y archivados en copia dura. Entiéndase como información que se maneja todo lo referido a las Afectaciones de los Profesores, a los Cambios de Turnos que estos solicitan y a la Solicitud de Locales para la realización de una actividad fuera del horario docente y además a todo lo relacionado con la gestión de la información sobre los estudiantes. Hasta el momento tramitar toda la información de estos procesos se hace lenta y engorrosa para el personal del Vicedecanato de Formación, la Planificación Docente y la Secretaría Docente. El objetivo de este trabajo es desarrollar un sistema informático que permita automatizar la gestión de la información en el Vicedecanato de Formación, la Planificación Docente y en la Secretaría Docente de la Facultad 9. Para esto se realiza un análisis de las tecnologías y lenguajes de programación más utilizados actualmente en Cuba y el mundo para el desarrollo de aplicaciones Web, se estudian otros sistemas existentes de Gestión de la Información Docente en Cuba y el Mundo, se hace una documentación de cómo se lleva a cabo el flujo de la información en el Vicedecanato de Formación, la Planificación Docente y la Secretaría Docente de la Facultad 9 y finalmente se diseña e implementa el sistema a desarrollar.



PALABRAS CLAVES

PALABRAS CLAVES

Gestión, Docencia, Secretaría Docente, Planificación Docente, Vicedecanato de Formación.



TABLAS Y FIGURAS

TABLA 1: DESCRIPCIÓN DE LOS ACTORES DEL NEGOCIO 45

TABLA 2: DESCRIPCIÓN DE LOS TRABAJADORES DEL NEGOCIO. 46

TABLA 3: DESCRIPCIÓN TEXTUAL DEL CU GESTIONAR AFECTACIÓN. 50

TABLA 4: DESCRIPCIÓN TEXTUAL DEL CU SOLICITAR LICENCIA DE MATRÍCULA. 51

TABLA 5: DESCRIPCIÓN TEXTUAL DEL CU SOLICITAR TRASLADOS EXTERNOS..... 52

TABLA 6: ACTORES DEL SISTEMA..... 56

TABLA 7: DESCRIPCIÓN TEXTUAL DEL CUS GESTIONAR AFECTACIÓN..... 68

TABLA 8: DESCRIPCIÓN TEXTUAL DEL CUS MOSTRAR LISTADOS PLANIFICACIÓN DOCENTE. 70

FIGURA 1: ARQUITECTURA CLIENTE/ SERVIDOR. 19

FIGURA 2: EL VOCABULARIO DE UML. 30

FIGURA 3: INTERFACES Y PROCESOS PRINCIPALES DE MÉTRICA 3. 32

FIGURA 4: FASES DE LA METODOLOGÍA XP. 33

FIGURA 5: LOS FLUJOS DE TRABAJO TIENEN LUGAR SOBRE LAS CUATRO FASES: INICIO, ELABORACIÓN, CONSTRUCCIÓN Y TRANSICIÓN..... 35

FIGURA 6: DIAGRAMA DE CASOS DE USO DEL NEGOCIO. 48

FIGURA 7: DIAGRAMA DE ACTIVIDADES DEL CU GESTIONAR AFECTACIÓN: SECCIÓN INFORMAR AFECTACIÓN..... 53

FIGURA 8: DIAGRAMA DE ACTIVIDADES DEL CU GESTIONAR AFECTACIÓN: SECCIÓN ELIMINAR AFECTACIÓN. 53

FIGURA 9: DIAGRAMA DE ACTIVIDADES DEL CU SOLICITAR LICENCIA DE MATRICULA. 54

FIGURA 10: DIAGRAMA DE ACTIVIDADES DEL CU SOLICITAR TRASLADO EXTERNO. 54

FIGURA 11: MODELO DE OBJETOS DEL NEGOCIO. 55

FIGURA 12: DIAGRAMA DE CASOS DE USO DEL SISTEMA. 64

FIGURA 13: DIAGRAMAS DE CLASES DEL ANÁLISIS DEL CUS GESTIONAR AFECTACIÓN..... 72

FIGURA 14: DIAGRAMAS DE CLASES DEL ANÁLISIS DEL CUS MOSTRAR LISTADOS PLANIFICACIÓN DOCENTE. 72

FIGURA 15: DIAGRAMA DE COLABORACIÓN DEL CUS GESTIONAR AFECTACIÓN: SECCIÓN INSERTAR AFECTACIÓN. 73

FIGURA 16: DIAGRAMA DE COLABORACIÓN DEL CUS GESTIONAR AFECTACIÓN: SECCIÓN MODIFICAR AFECTACIÓN. 73

FIGURA 17: DIAGRAMA DE COLABORACIÓN DEL CUS GESTIONAR AFECTACIÓN: SECCIÓN ELIMINAR AFECTACIÓN. 74

FIGURA 18: DIAGRAMA DE COLABORACIÓN DEL CUS MOSTAR LISTADOS PLANIFICACIÓN DOCENTE: SECCIÓN LISTADO AFECTACIONES..... 74

FIGURA 19: DIAGRAMA DE COLABORACIÓN DEL CUS MOSTAR LISTADOS PLANIFICACIÓN DOCENTE: SECCIÓN LISTADO DE AFECTACIONES POR DEPARTAMENTO..... 74

FIGURA 20: DIAGRAMA DE COLABORACIÓN DEL CUS MOSTAR LISTADOS PLANIFICACIÓN DOCENTE: SECCIÓN LISTADO DE SOLICITUDES DE CAMBIO..... 75

FIGURA 21: DIAGRAMA DE COLABORACIÓN DEL CUS MOSTAR LISTADOS PLANIFICACIÓN DOCENTE: SECCIÓN LISTADO DE SOLICITUDES DE CAMBIO POR DEPARTAMENTO..... 75

FIGURA 22: DIAGRAMA DE COLABORACIÓN DEL CUS MOSTAR LISTADOS PLANIFICACIÓN DOCENTE: SECCIÓN LISTADO DE SOLICITUD DE LOCALES..... 75

FIGURA 23: DIAGRAMA DE COLABORACIÓN DEL CUS MOSTAR LISTADOS PLANIFICACIÓN DOCENTE: SECCIÓN LISTADO DE SOLICITUD DE LOCALES POR DEPARTAMENTO..... 76

FIGURA 24: EL FLUJO DE TRABAJO DE SYMPHONY..... 77

FIGURA 25: DIAGRAMA DE CLASES DEL DISEÑO DEL CUS GESTIONAR AFECTACIÓN..... 78

FIGURA 26: DIAGRAMA DE CLASES DEL DISEÑO DEL CUS MOSTRAR LISTADOS PLANIFICACIÓN DOCENTE. 79

FIGURA 27: PLANTILLA DECORADA CON UN LAYOUT..... 82



TABLAS Y FIGURAS

FIGURA 28: DIAGRAMA DE CLASES PERSISTENTES.....	83
FIGURA 29: MODELO DE DATOS.....	85
FIGURA 30: DIAGRAMA DE DESPLIEGUE.....	86
FIGURA 31: DIAGRAMA DE COMPONENTES DEL CU GESTIONAR AFECTACIÓN.	88
FIGURA 32: DIAGRAMA DE COMPONENTES DEL CU MOSTRAR LISTADOS PLANIFICACIÓN DOCENTE.....	89



ÍNDICE

Resumen	IX
Introducción	1
Capítulo 1: Fundamentación Teórica del Subsistema Docente de la Facultad 9.	5
1.1 Introducción.....	5
1.2 Conceptos Asociados a la Gestión de la Información Docente en el Vicedecanato de Formación, la Planificación Docente y la Secretaría Docente de la Facultad 9.	5
1.2.1 Concepto de Gestión.	5
1.2.2 Concepto de Gestión de la Información.	5
1.2.3 Concepto de Sistema.	5
1.2.4 Concepto de Sistema Informático.	6
1.2.5 Concepto de Subsistema.....	6
1.2.6 Procesos Sustantivos.	6
1.2.7 Planificación Docente.	6
1.2.8 Secretaría Docente.....	7
1.2.9 Vicedecanato de Formación.	7
1.3 Objeto de Estudio.....	7
1.3.1 Descripción General.	7
1.3.2 Vicedecanato de Formación y Secretaría Docente de la Facultad 9.	8
1.3.3 Procesos que se realizan actualmente en el Vicedecanato de Formación y la Secretaría Docente de la Facultad 9.	9
1.4 Sistemas existentes de Gestión de la Información Docente en Cuba y el Mundo.	13
1.4.1 Ámbito Internacional.	13
1.4.2 Ámbito Nacional.....	14
1.5 Conclusiones.....	16



Capítulo 2: Tendencias y Tecnologías Actuales a Desarrollar para el Subsistema Docente de la Facultad 9.....	17
2.1 Introducción.....	17
2.2 Las Tecnologías de la Información y las Comunicaciones (TIC).....	17
2.3 Aplicaciones Web.....	17
2.4 Lenguajes de Programación Web.....	19
2.4.1 Lenguajes de Programación Web del lado del Cliente.....	19
2.4.1.1 JavaScript.....	19
2.4.1.2 Hypertext Markup Language (HTML).....	20
2.4.2 Lenguajes de Programación Web del lado del Servidor.....	20
2.4.2.1 "PHP Hypertext Pre-processor" (PHP).....	20
2.4.2.2 "Active Server Pages" (ASP).....	20
2.4.2.3 Java Server Pages (JSP).....	21
2.5 ¿Por qué se utilizó JavaScript como Lenguaje para Validar?	22
2.6 ¿Por qué se utilizó HTML?	22
2.7 ¿Por qué se utilizó PHP como Lenguaje de Programación Web?	22
2.8 Gestores de Base Datos.....	23
2.8.1 Oracle.....	24
2.8.2 MySQL.....	25
2.8.3 PostgreSQL.....	26
2.9 ¿Por qué se utilizó PostgreSQL como Gestor de Base Datos?	27
2.10 El Lenguaje Unificado de Modelado (UML) como soporte de la modelación de la solución propuesta.....	27
2.11 Metodologías de Desarrollo de Software.....	30
2.11.1 Métrica 3: Metodología de Planificación, Desarrollo y Mantenimiento de Sistemas de Información.....	30
2.11.2 Metodología Extreme Programming (XP).....	32
2.11.3 El Proceso Unificado de Desarrollo de Software (RUP).....	33
2.12 ¿Por qué se utilizó El Proceso Unificado de Desarrollo de Software (RUP) como base en el desarrollo de la solución?.....	35



2.13	Frameworks para PHP.	36
2.13.1	Kumbia.	36
2.13.2	CakePHP.	37
2.13.3	Symfony.	37
2.14	¿Por qué se utiliza Symfony como framework ?	38
2.15	Herramientas de Modelado.	39
2.15.1	Rational Rose.	39
2.15.2	Visual Paradigm for UML.	40
2.16	¿Por qué se utiliza Rational Rose como herramienta de modelado?	40
2.17	Otras Herramientas.	40
2.17.1	Servidor Apache.	40
2.17.2	Eclipse.	42
2.18	Conclusiones.	42
Capítulo 3: Presentación de la solución propuesta para el Subsistema Docente de la Facultad 9.		43
3.1	Introducción.	43
3.2	Reglas del Negocio.	43
3.3	Modelo de Negocio.	44
3.3.1	Actores del Negocio.	45
3.3.2	Trabajadores del Negocio.	45
3.3.3	Procesos de Negocio.	46
3.3.4	Diagrama de Casos de Uso del Negocio.	47
3.3.5	Descripción Textual de los Casos de Uso de Negocio.	49
3.3.6	Diagramas de Actividades de los Casos de Uso del Negocio.	53
3.3.7	Modelo de Objetos del Negocio.	55
3.4	Modelo del Sistema Propuesto.	55



ÍNDICE

3.4.1	Actores del Sistema	56
3.4.2	Requisitos Funcionales.....	56
3.4.3	Requisitos No Funcionales.....	61
3.4.4	Diagrama de Casos de Uso del Sistema.....	63
3.4.5	Descripción de los Casos de Uso del Sistema.....	65
3.5	Conclusiones.....	70
Capítulo 4: Construcción de la solución propuesta para el Subsistema Docente de la Facultad 9.		
	71
4.1	Introducción.....	71
4.2	Modelo de Análisis.....	71
4.2.1	Diagramas de Clases del Análisis.....	71
4.2.2	Diagramas de Interacción: Colaboración.....	73
4.3	Modelo de Diseño.....	76
4.3.1	Diagramas de Clases del Diseño.....	77
4.4	Principios del Diseño.....	80
4.4.1	Tratamiento de excepciones y seguridad del Sistema.....	80
4.4.2	Patrones de Diseño implementados por Symfony y utilizados en el Diseño del Subsistema Docente de la Facultad 9.....	80
4.4.2.1	Patrones GRASP.....	80
4.4.2.2	Patrones GOF.....	81
4.5	Diseño de la Base de Datos.....	82
4.5.1	Diagrama de Clases Persistentes.....	83
4.5.2	Modelo de Datos.....	84
4.6	Modelo de Despliegue.....	86
4.7	Modelo de Implementación.....	87
4.7.1	Diagramas de Componentes.....	87
4.8	Conclusiones.....	90



ÍNDICE

Conclusiones	91
Recomendaciones	92
Referencias Bibliográficas.....	93
Bibliografía.....	96
Glosario de Términos.....	99



Introducción

Al triunfo de la Revolución Cubana el 1º de enero de 1959, el pueblo cubano se encontraba sumido en una crisis social y económica de carácter estructural. Existía una economía deformada, de base agropecuaria atrasada, con un desarrollo industrial escaso y concentrado principalmente en la rama azucarera. Con el paso de los años se ha identificado la conveniencia y necesidad de introducir y dominar en el país las Tecnologías de la Información y las Comunicaciones (TIC) y lograr una cultura digital como una de las características imprescindibles del hombre nuevo.

Las TIC se pueden definir como aquellos medios y servicios que permiten recopilar, almacenar y transmitir información con medios electrónicos.(1)

Aprovechando las nuevas tecnologías de la información y las comunicaciones en apoyo a la infraestructura ya existente, se trabaja en diversos programas para el desarrollo del individuo, la economía y la sociedad, privilegiando el uso social y colectivo de las TIC. Se ha priorizado la Educación, la Salud, la Ciencia y la Técnica, las Universidades, las instituciones del Gobierno, la Administración Central del Estado, y los servicios al ciudadano.(2)

La Informatización de la Sociedad es el proceso de utilización ordenada y masiva de las Tecnologías de la Información y las Comunicaciones en la vida cotidiana, para satisfacer las necesidades de todas las esferas de la sociedad, en su esfuerzo por lograr cada vez más eficacia y eficiencia en todos los procesos y por consiguiente mayor generación de riqueza y aumento en la calidad de vida de los ciudadanos.(3)

Una sociedad que aplique la informatización en todas sus esferas y procesos será más eficaz, eficiente y competitiva. La preparación de las nuevas generaciones en la utilización de las TIC y el empleo de éstas para aumentar la calidad del proceso docente educativo son elementos que buscan asegurar el futuro de la revolución. La Industria Cubana del Software (ICSW) está llamada a convertirse en una significativa fuente de ingresos para el país, la dirección de la revolución ha brindado especial interés en la preparación de los especialistas de las TIC ejemplo de ello es la Universidad de las Ciencias Informáticas (UCI) con la cual en nuestro país se han logrado cambios tanto en el proceso educativo como en la industria del software cubano.

La Universidad de las Ciencias Informáticas trabaja con el objetivo de convertirse en una ciudad digital, es decir, que todos los procesos posibles, que dentro de ella se desarrollan, estén informatizados, la



INTRODUCCIÓN

institución cuenta con una infraestructura que soporta el desarrollo de sistemas informáticos capaces de gestionar gran cantidad de información con un costo mínimo de tiempo de respuesta y con la posibilidad de integración entre estos.(4)

En la Facultad 9 los procesos que se realizan en el Vicedecanato de Formación, la Planificación Docente y en la Secretaría Docente no están automatizados, el gran volumen de información que se maneja se hace manualmente, los datos son recogidos y archivados en copia dura. Esto conlleva a que puedan ocurrir errores, inconsistencia y ambigüedades en los datos así como en algunos casos pérdida de información. Esto favorece el surgimiento del siguiente **problema científico**: No existencia de un sistema informático que permita el control de los procesos sustantivos que se llevan a cabo en el Vicedecanato de Formación, la Planificación Docente y la Secretaría Docente de la Facultad 9, así como el trabajo constante con la información que sobre estos se maneja. De aquí que el **objeto de estudio** de esta investigación sea la Gestión de la Información en el Vicedecanato de Formación, la Planificación Docente y la Secretaría Docente de la Facultad 9. Se ha planteado como **objetivo general**: desarrollar un sistema informático que permita el control de los procesos sustantivos que se llevan a cabo en el Vicedecanato de Formación, la Planificación Docente y la Secretaría Docente de la Facultad 9, y así darle una solución factible al problema existente, planteándose los siguientes **objetivos específicos**:

- Documentar el flujo de información entre el Vicedecanato de Formación, la Planificación Docente y la Secretaría Docente de la Facultad 9 que trabajan y operan en conjunto.
- Establecer las modificaciones necesarias al flujo de información logrando su optimización máxima.
- Diseñar una Base de Datos necesaria para sustentar el sistema a desarrollar.
- Realizar las validaciones necesarias para comprobar la validez del producto.

Se hace necesario profundizar en el estudio del **campo de acción** de esta investigación: la Informatización de la Gestión de la Información en el Vicedecanato de Formación, la Planificación Docente y la Secretaría Docente de la Facultad 9. La **Hipótesis que se plantea** es: Si se estudian los procesos que se llevan a cabo en el Vicedecanato de Formación, la Planificación Docente y la Secretaría Docente de la Facultad 9 y las tecnologías necesarias para el desarrollo de aplicaciones Web entonces se podrá crear una aplicación que permita automatizar dichos procesos.

Las **tareas** planteadas para llevar a cabo la realización de la investigación son:



INTRODUCCIÓN

- Análisis del estado del arte de las principales y más actuales tecnologías utilizadas para el diseño y desarrollo de aplicaciones Web en Cuba y el mundo.
- Análisis de estándares nacionales e internacionales para el desarrollo de aplicaciones Web.
- Análisis de las condiciones actuales en las cuales se desarrolla el proceso de Gestión Docente de la Facultad.
- Planteamiento del flujo de información, sus nodos centrales, así como los datos de entrada y reportes para la actividad de gestión Docente y el flujo de la Información que se maneja.
- Diseño del subsistema para la Gestión Docente de la Facultad.
- Desarrollo de un prototipo funcional para el subsistema de Gestión Docente de la Facultad.
- Validación del producto.

Los métodos de investigación científica que se utilizan en esta investigación son:

Métodos Teóricos

- Análisis Histórico Lógico: Se realizó el análisis histórico lógico de los procesos sustantivos que se llevan a cabo en el Vicedecanato de Formación, la Planificación Docente y la Secretaría Docente de la Facultad 9.
- Modelación: Se utiliza este método para la modelación de los diagramas necesarios para llevar a cabo la construcción del sistema funcional.

Métodos Empíricos

- Observación: Se realizó una observación detallada de cómo se llevan a cabo actualmente los procesos que se realizan en el Vicedecanato de Formación, la Planificación Docente y la Secretaría Docente de la Facultad 9.
- Entrevista: Se realizaron entrevistas a todo el personal del Vicedecanato de Formación y de la Secretaría Docente de la Facultad 9 para conocer las condiciones actuales en las cuales se desarrolla el proceso de Gestión de la Información Docente de la Facultad.

Con el desarrollo de esta investigación se espera obtener varios resultados como:

- Documentación de los procesos que sostienen el Vicedecanato de Formación, la Planificación Docente y la Secretaría Docente en la Facultad 9 y sus mejoras potenciales para un funcionamiento óptimo.
- Diseño de la Base de Datos que contendrá la información de los procesos mencionados.
- Sistema Web para la informatización de la actividad de control de dichas áreas.



Este documento cuenta con 4 partes esenciales:

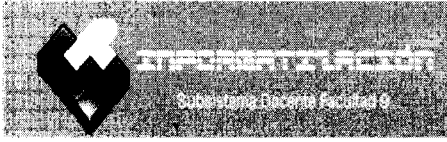
Capítulo 1: Fundamentación Teórica del Subsistema Docente de la Facultad 9.

En este capítulo se tratan los conceptos fundamentales asociados al dominio del problema. Además se hace un estudio más profundo del problema a resolver.

Capítulo 2: Tendencias y tecnologías actuales a desarrollar para el Subsistema Docente de la Facultad 9. En este capítulo se tratan las tecnologías utilizadas en el desarrollo de aplicaciones Web haciendo énfasis en las utilizadas para el desarrollo del sistema a implementar. También se explica la metodología y el lenguaje de modelado utilizado.

Capítulo 3: Presentación de la solución propuesta para el Subsistema Docente de la Facultad 9. En este capítulo se detallan las características del sistema mediante el modelamiento del negocio, y se especifican los requerimientos del software. Además se describe el sistema propuesto mediante los casos de uso del sistema.

Capítulo 4: Construcción de la solución propuesta para el Subsistema Docente de la Facultad 9. En este capítulo se realiza el diagrama de clases del análisis, se presentan el diseño del sistema, el modelo de implementación, así como las pruebas realizadas al sistema propuesto.



CAPÍTULO 1

Fundamentación Teórica del Subsistema Docente de la Facultad 9.

1.1 Introducción.

El presente capítulo aborda los conceptos asociados al dominio del problema, describe los procesos actuales que se llevan a cabo en el Vicedecanato de Formación, la Planificación Docente y la Secretaría Docente de la Facultad 9, identificándose las principales causas que motivan a la realización de este sistema y explica más detalladamente el objeto de estudio de la investigación. En este capítulo se realiza un análisis más exhaustivo de la situación problemática existente y además se analizan otras soluciones existentes que parcialmente resuelven el problema científico planteado.

1.2 Conceptos Asociados a la Gestión de la Información Docente en el Vicedecanato de Formación, la Planificación Docente y la Secretaría Docente de la Facultad 9.

1.2.1 Concepto de Gestión.

Es el proceso mediante el cual se obtiene, despliega o utiliza una variedad de recursos básicos para apoyar los objetivos de la organización.(5)

1.2.2 Concepto de Gestión de la Información.

Es todo lo relacionado con la obtención de la información adecuada, en la forma correcta, para la persona indicada, al costo adecuado, en el tiempo oportuno, en el lugar apropiado, para tomar la acción correcta.(5)

1.2.3 Concepto de Sistema.

Conjunto de entes independientes entre sí mismos que se encuentran en interrelación con ellos mismos y con el ambiente que los rodea.(5)



1.2.4 Concepto de Sistema Informático.

Cualquier conjunto de dispositivos que colaboran en la realización de una tarea. Sistema se refiere también a cualquier colección o combinación de programas, procedimientos, datos y equipamiento utilizado en el procesamiento de información. (6)

1.2.5 Concepto de Subsistema.

Un subsistema es un entorno operativo único y predefinido a través del cual el sistema coordina el flujo de trabajo y la utilización de recursos. Los subsistemas gestionan recursos. (7)

1.2.6 Procesos Sustantivos.

Son los procesos no automatizados que se realizan en el Vicedecanato de Formación, la Planificación Docente y la Secretaría Docente de la Facultad 9.

1.2.7 Planificación Docente.

Se conoce como Planificación Docente a los procesos que incluyen la planificación del horario docente. Algunos de estos procesos y los que nos conciernen son:

- Afectaciones de los profesores.
- Solicitud de cambio.
- Solicitud de locales para docencia.

Afectaciones de los profesores:

Constituyen afectaciones para los profesores aquellos espacios que tienen comprometidos con actividades no docentes importantes, ejemplos: preparaciones metodológicas, maestrías, cursos de postgrados, entre otras.

Solicitud de cambio de turno:

Constituye una solicitud de cambio de turno cuando un profesor se ve imposibilitado de impartir un turno de clases y solicita un cambio de ese turno para otro espacio en el horario.



Solicitud de locales para docencia:

Constituye una solicitud de locales para docencia cuando los profesores necesitan realizar alguna actividad que no está planificada en el horario y solicitan un local que no esté ocupado en el momento de desarrollar su actividad.

1.2.8 Secretaría Docente.

Es donde se gestiona toda la información docente de los estudiantes de la facultad.

1.2.9 Vicedecanato de Formación.

- El Vicedecanato de Formación es el encargado directo de hacer cumplir el Plan de Estudio del Ingeniero en Ciencias Informáticas en la Universidad de las Ciencias Informáticas (UCI).
- Dirige el proceso de formación del estudiante y para esto es responsable de planificar y velar porque se realice la Planificación Docente.
- Es el responsable del Desarrollo del Trabajo Educativo, del Trabajo Docente Metodológico, de todo el proceso docente en general.

1.3 Objeto de Estudio.

1.3.1 Descripción General.

El objeto de estudio de esta investigación es la gestión de la información en el Vicedecanato de Formación, la Planificación Docente y la Secretaría Docente de la Facultad 9. Entiéndase por la información que se va a gestionar:

Planificación Docente

1. Afectaciones de los profesores.
2. Solicitud de cambio de turno.
3. Solicitud de locales para docencia.

Secretaría Docente

1. Cartas de Advertencia.
2. Ratificación de Matrícula.



3. Bajas:
 - Bajas por Deserción.
 - Bajas Académicas.
 - Bajas Voluntarias.
 - Bajas por Sanción.
4. Control de Cadetes:
 - Control de Cadetes del MINFAR.
 - Control de Cadetes del MININT.
5. Control de Estudiantes CDI.
6. Traslados Externos.
7. Licencias de Matrículas.
8. Resolución Decanal:
 - Resolución de Alumnos Ayudantes.
 - Resolución de Instructores No Graduados.
 - Resolución de Licencias Especiales.
 - Resolución de Sanciones.
 - Resolución de Traslados Internos.

1.3.2 Vicedecanato de Formación y Secretaría Docente de la Facultad 9.

El Vicedecanato de Formación está integrado por el Vicedecano de Formación y la Planificadora Docente los cuales son los encargados de la realización del horario docente de la facultad, la Planificadora Docente igualmente controla las afectaciones de los profesores, estas son informadas por los distintos Jefes de Departamentos de la facultad, además se encarga de realizar los cambios de turnos a los profesores que se encuentren imposibilitados para impartirlos, cuyos Jefes de Departamentos los aprueben. Por otra parte facilita locales a los profesores que se lo soliciten para actividades fuera del horario.

La Secretaría Docente está integrada por la Secretaria Docente y la Técnico en Trabajo Docente Investigativo de la Educación Superior (TECADIES). La Secretaria Docente se encarga de las Resoluciones Decanal, las cuales incluyen la Resolución de Alumnos Ayudantes que posee el listado oficial de los Alumnos Ayudantes tanto los nuevos como los que ratifican, la Resolución de Instructores No Graduados que posee el listado oficial de los Instructores No Graduados tanto los nuevos como los que ratifican, la Resolución de Licencias Especiales que posee el listado oficial de los estudiantes que



han solicitado al Decano una licencia especial, la Resolución de Sanciones que posee el listado oficial de los estudiantes que han sido sancionados, Resolución de Traslados Internos que posee el listado oficial de los estudiantes que han solicitado al Decano un traslado para otra facultad de la universidad, así como el listado oficial de los estudiantes que han solicitado al Decano entrada a la facultad. Ambas realizan un trabajo en conjunto en el procesamiento de la información sobre: Bajas, las cuales pueden ser solicitadas por el estudiante o solicitadas por el centro; Cadetes, donde controlan la información necesaria sobre los estudiantes cadetes del MINFAR y del MININT; Estudiantes CDI, donde registran los datos sobre los estudiantes que se encuentran de misión en los CDI de Venezuela; Actas por Indisciplina las cuales se les aplican a estudiantes que han cometido alguna indisciplina; Ratificación de Matrícula la cual se aplica a todos los estudiantes de la facultad, constituye una boleta que es archivada en el expediente de cada uno de ellos; Traslados Externos que posee el listado oficial de los estudiantes que han solicitado traslado para otra universidad del país, Licencias de Matrículas que posee el listado oficial de los estudiantes que han solicitado al Decano una licencia de matrícula.

1.3.3 Procesos que se realizan actualmente en el Vicedecanato de Formación y la Secretaría Docente de la Facultad 9.

En la Facultad 9 la gran mayoría de los procesos que se realizan en el Vicedecanato de Formación, la Planificación Docente y en la Secretaría Docente actualmente no están automatizados, los datos son recogidos y archivados en copia dura y en Hojas de Cálculo de Microsoft Excel. Esto conlleva a que puedan ocurrir errores, inconsistencia y ambigüedades en los datos así como en algunos casos pérdida de información. A continuación se describe con exactitud como se llevan a cabo estos procesos en la actualidad.

Los profesores informan a su Jefe de Departamento las afectaciones que poseen para la ubicación de sus turnos en el horario docente, el Jefe de Departamento las aprueba o no y el listado final lo envía a la Planificadora Docente, la cual elabora el horario teniendo en cuenta la información que le fue enviada, si desaparece alguna de las existentes el Jefe de Departamento actualiza el listado y lo envía a la Planificadora. El listado de afectaciones de los profesores cuenta con el nombre del departamento al que pertenecen, nombre y apellidos del profesor, día, sección, turno y motivo de la afectación.

Finalizando cada semana los profesores solicitan a su Jefe de Departamento cambios de turnos de clases de la semana siguiente, el Jefe de departamento los aprueba o no y le informa a la Planificadora Docente esta información, estos cambios no son reflejados en el horario docente de la semana



CAPÍTULO 1

siguiente pero los estudiantes implicados son informados cuando el profesor impartirá el turno cambiado.

Cuando los profesores de la facultad necesitan realizar una determinada actividad que se encuentra fuera del horario docente, estos solicitan a la Planificadora Docente un local para la realización de la misma, esta consulta el horario y asigna al profesor el local que tenga disponible en el horario especificado por este.

Si un determinado profesor decide ponerle un acta por indisciplina a alguno de sus estudiantes por alguna indisciplina cometida, este llena el modelo del acta y la lleva al Decano para su aprobación y si es aprobada, es localizado al estudiante para que la firme. El profesor entrega el acta en la secretaría para archivarla. La Secretaria Docente archiva la copia dura y en una Hoja de Cálculo de Microsoft Excel inserta los siguientes datos: carné de identidad, sexo, nombre, primer apellido, segundo apellido del estudiante, fecha, grupo, año, causa, sancionado por, vía de ingreso a la universidad (Diferido, Escalafón, MINFAR, MININT, Orden 18, Orden 18 Experimental, Deporte) y observaciones.

La Ratificación de Matrícula se aplica a todos los estudiantes de la facultad, constituye una boleta que es archivada por la TECADIES en el expediente de cada uno de los estudiantes. Los datos de la boleta son: curso, nombre, apellidos, carné de identidad, año matriculado, grupo, tipo de curso (Diurno o Por Encuentro), situación escolar, si es interno o externo, nacionalidad, dirección actual, provincia, municipio, organizaciones políticas a las que pertenece (PCC, UJC), teléfono, otros datos de interés, elaborado por (firma), firma del estudiante y fecha.

Las Bajas pueden ser: solicitadas por los estudiantes que sería una baja voluntaria o solicitadas por el Decano que sería una baja académica, por sanción o por deserción. Las bajas por deserción las decide el Decano por causas como: la no incorporación del estudiante en un período determinado a la universidad (semestre), fallecimiento, entre otras, la secretaria llena un modelo de baja con los datos del estudiante y el Decano lo firma. Toda la información sobre las bajas es controlada por la Secretaria Docente en una Hoja de Cálculo de Microsoft Excel donde almacena los siguientes datos: carné de identidad, sexo, nombre, primer apellido, segundo apellido, fecha, facultad, año, grupo, provincia, municipio, causa (voluntarias, deserción, académica, sanción), vía de ingreso (Diferido, Escalafón, MINFAR, MININT, Orden 18, Orden 18 Experimental, Deporte), procedencia (IPVCE, IPUEC, Técnico Medio en Informática, ESPA), observaciones, organización a la que pertenece y teléfono.



CAPÍTULO 1

El Control de los Cadetes de la facultad tanto del MINFAR como del MININT es llevado por la Secretaria Docente en una Hoja de Cálculo de Microsoft Excel donde almacena los siguientes datos: carné de identidad, sexo, nombre, primer apellido, segundo apellido, facultad, año, grupo, provincia, municipio, vía de ingreso (Diferido, Escalafón, MINFAR, MININT, Orden 18, Orden 18 Experimental, Deporte), procedencia (IPVCE, IPUEC, Técnico Medio en Informática, ESPA), organización a la que pertenece, teléfono y dirección particular.

Para el Control de Estudiantes de CDI es el Decano de la facultad el que le informa a la Secretaria Docente el listado que posee el nombre y apellidos así como el grupo de los estudiantes implicados en esta tarea.

Los Traslados Externos se realizan cuando un estudiante presenta al Decano una carta de solicitud de traslado para otra universidad, en caso de ser aprobado el Decano envía a la Secretaria Docente dicha carta quien la reenvía a la Secretaria General de la Universidad, donde en una comisión que se realiza integrada por ella se aceptan o no los traslados, el listado con los estudiantes a los que se les ha aceptado el traslado le es enviado a la Secretaria Docente de la facultad quien prepara los expedientes y la información necesaria del traslado y cuando la Secretaria General de la Universidad lo solicita se lo envía. La Secretaria Docente guarda en una Hoja de Cálculo de Microsoft Excel los siguientes datos de los estudiante: carné de identidad, nombre, primer apellido, segundo apellido, grupo, fecha, motivo, dirección particular, provincia, municipio, observaciones, procedencia y vía de ingreso.

Las Licencias de Matrícula son un derecho de todos los estudiantes, pueden pedir al Decano una licencia de matrícula todos los estudiantes incluidos los estudiantes que constituyen licencias especiales, el Decano analiza la solicitud y si considera que los elementos planteados por dicho estudiante en el momento en que se realiza la solicitud corresponden a los requisitos para una Licencia de Matrícula la acepta y envía la resolución a la Secretaria Docente que la archiva y almacena en una Hoja de Cálculo de Microsoft Excel los siguientes datos: carné de identidad, sexo, nombre, primer apellido, segundo apellido, fecha, facultad, año, grupo, provincia, municipio, causa (Misiones internacionalistas o Movilización militar, Enfermedad o Accidente, Maternidad o Embarazo, Participación en actividades deportivas como atletas de alto rendimiento, Participación en actividades de grupos nacionales de cultura por ser miembros de los mismos, Necesidad impostergable de la producción o los servicios, Desaprobar una asignatura en la convocatoria extraordinaria de fin de curso si, estando matriculado en el curso regular diurno, cumplen con las disposiciones que al respecto se dictan en el Capítulo V del Reglamento, Problemas Familiares), vía de ingreso(Diferido, Escalafón,



CAPÍTULO 1

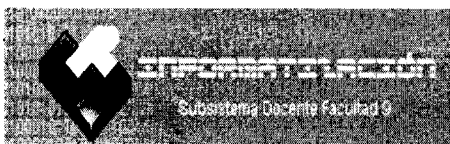
MINFAR, MININT, Orden 18, Orden 18 Experimental, Deporte), procedencia (IPVCE, IPUEC, Técnico Medio en Informática, ESPA), observaciones, organización a la que pertenece, teléfono y si es Licencia especial o no. La Licencia de Matrícula autoriza al estudiante a reincorporarse nuevamente a los estudios que cursaba considerándose reingreso, reconociéndosele las asignaturas aprobadas anteriormente; aunque debe renovarse al inicio de cada curso en tanto no cese la causa que la motiva.

Las Resoluciones Decanal se archivan en copia dura, estas poseen un número consecutivo y el curso al cual pertenecen. Existen diferentes categorías de Resoluciones Decanal, estas son:

Resolución de Alumnos Ayudantes, antes de emitir esta resolución el Decano de la facultad informa a los estudiantes que fueron Alumnos Ayudantes el año anterior que necesitan ratificar su ayudantía, e informa al resto de los estudiantes sobre la convocatoria de Alumnos Ayudantes, los estudiantes interesados lo informan al Decano, este envía a la Secretaria Docente los listados de estudiantes para que esta calcule el promedio y obtenga la información sobre si llevan mundial o no, o si ha sido sancionado o no, de cada uno de los estudiantes. Esta envía al Decano la información que le solicitó, el cual analiza la información y le envía los listados oficiales de la resolución. Una resolución se queda en secretaría y otra se le envía a nómina para que le sea realizado el pago a los estudiantes. Estos listados constan de nombre, apellidos y carné de identidad del estudiante, además la Secretaria Docente en una Hoja de Cálculo de Microsoft Excel almacena los siguientes datos: nombre, primer apellido, segundo apellido, asignatura que imparte, departamento al que pertenece, grupo donde recibe clases, grupo al que imparte clases, promedio histórico, si lleva mundial y si ha sido sancionado.

Resolución de Instructores No Graduados, un estudiante se convierte en Instructor No Graduado si lleva más de 2 años siendo Alumno Ayudante, el Decano informa a los Instructores No Graduados y a los que se convierten en nuevos Instructores No Graduados que necesitan ratificar su ayudantía, en adelante ocurren lo mismo que en la Resolución de Alumnos Ayudantes.

Resolución de Licencias Especiales, cualquier estudiante que suspenda una asignatura en mundial tiene el derecho de pedir al Decano una licencia especial, que consiste en comenzar el próximo curso en el mismo año en el que estaba, repitiendo la asignatura desaprobada y adelantando varias del curso próximo. El Decano analiza las solicitudes y aprueba las licencias cuyos estudiantes merecen esta oportunidad, envía a la Secretaria Docente el listado con los estudiantes que constituirán licencias especiales del curso, que consta de nombre, apellidos y carné de identidad, además la Secretaria Docente en una Hoja de Cálculo de Microsoft Excel almacena los siguientes datos: nombre, primer apellido, segundo apellido, grupo oficial, asignatura que suspendió, asignaturas que va a adelantar.



Resolución de Sancionados, son emitidas cuando un estudiante es sancionado en alguna comisión disciplinaria por alguna indisciplina que haya cometido, una vez que el estudiante haya firmado la sanción, la Secretaria Docente archiva en una en una Hoja de Cálculo de Microsoft Excel los siguientes datos del estudiante: carné de identidad, sexo, primer apellido, segundo apellido, nombre, fecha, facultad, año, grupo, provincia, municipio, sanción, causa, sancionado por, vía de ingreso a la universidad, procedencia, observaciones.

Resolución de Traslados Internos el estudiante envía un correo solicitando el traslado, al Decano de la facultad y al Decano de la facultad hacia donde desea trasladarse, después de ser aceptado por ambos el Decano de la facultad le envía a la Secretaria Docente los correos de los estudiantes que están solicitando traslado, esta lo reenvía a la Secretaria General de la universidad que es la que acepta o no finalmente el traslado, luego envía esta información a la Secretaria Docente y esta emite la resolución. Cuando un estudiante de otra facultad solicita entrada a la facultad se lleva a cabo el mismo proceso que cuando un estudiante solicita salida de la facultad. La Secretaria Docente guarda en una Hoja de Cálculo de Microsoft Excel los siguientes datos de los estudiantes: carné de identidad, nombre, primer apellido, segundo apellido, facultad de la que sale, grupo, facultad a la que entra, grupo; esta información se almacena tanto para los estudiantes que entran como los que salen de la facultad.

1.4 Sistemas existentes de Gestión de la Información Docente en Cuba y el Mundo.

1.4.1 Ámbito Internacional.

En el ámbito internacional fueron muchos los sistemas para la Gestión de la Información Docente encontrados, la gran mayoría de las universidades del mundo tienen implementado su propio sistema que les proporciona ventajas como: seguridad y organización de la información docente, rapidez en el trabajo del personal que gestiona la información docente, entre otras. Prácticamente toda la información que estos manejan está relacionada con la información que se maneja en el Vicedecanato de Formación y la Secretaría Docente pero no se encontró un subsistema específico que gestionara la información.

Sistema de Información Docente (SID) de la Universidad de Chile.

La Universidad de Chile cuenta con un Sistema de Información Docente creado por la Facultad de Ciencias Físicas y Matemáticas referente principalmente a control docente y encuestas docentes. Todo



el proceso, está orientado a prestar servicios a los alumnos, profesores, departamentos y autoridades. Dentro de estos se pueden destacar los servicios prestados a los profesores: el resultado de su encuesta docente, catálogo de cursos, horario personalizado profesores y coordinador de cursos.

La implementación de este sistema ha simplificado enormemente toda la gestión, ahorrándose, tiempo, más de 100.000 papeles impresos por semestre y lo que es más importante proporciona una mejor calidad de vida y atención para los alumnos y profesores, quienes pueden realizar la tramitación de inscripciones de cursos, modificar la inscripción académica, eliminar cursos y la participación en encuestas docentes a través de INTERNET desde cualquier lugar del país o del mundo.

1.4.2 **Ámbito Nacional.**

En el ámbito nacional se encontraron algunos sistemas para la Gestión de la Información Docente que aunque no en su totalidad cubren en alguna medida el trabajo de Gestión de la Información del Vicedecanato de Formación y la Secretaría Docente.

Sistema Automatizado de Información Docente (SAIDO).

Fue uno de los primeros sistemas diseñados e implantados por el Centro de Cibernética Aplicada a la Medicina (CECAM). A partir de su explotación se han incorporado nuevos módulos y subsistemas y se han elaborado otros sistemas relacionados con él, de acuerdo a las necesidades de los usuarios y al avance de las tecnologías computacionales. Actualmente está implantado al nivel de Facultad e Instituto en todo el país.

SAIDO cuenta con paquetes de sistemas de Información Docente específicamente el paquete **SAIDO-F**: (Sistema automatizado de información docente, facultades) Es un sistema de base de datos que contiene y procesa información sobre los estudiantes. Incluye módulos de matrícula, actualización y notas.

Sistema para la Gestión Académica: GestAcad.

GestAcad fue creado en la Facultad de Informática de la Universidad Matanzas "Camilo Cienfuegos". El sistema desde su creación ha dado lugar a la aparición de dos versiones. La primera donde la concepción fue la creación de módulos Stand Alone para la actualización de los datos y un sitio Web destinado a mostrar reportes de la información almacenada. La segunda, ya en explotación en la Universidad de Matanzas, esta concebida para desprenderse de aquellas dependencias de los módulos programados en un lenguaje no Open Source y la migración total hacia plataformas de



CAPÍTULO 1

Software Libre. Esta versión está completamente basada en plataforma Web y está totalmente programada en PHP. Ya incluye los módulos correspondientes de administración, de secretaría y de matrícula.

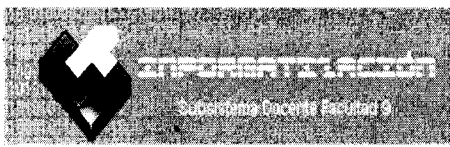
En la actualidad el sistema es capaz de realizar diversas acciones y brindar numerosos reportes como son:

- Listado de estudiantes por grupo.
- Reportes dinámicos de la información existente.
- Reporte de notas por asignatura y grupo.
- Tabla con los resultados docentes de un grupo en un semestre.
- Reporte de los resultados académicos de un estudiante en toda su carrera.
- Actas de exámenes de las diferentes asignaturas.
- Dar baja a un estudiante.(8)

AKADEMOS.

Es el sistema de gestión académica de la Universidad de Ciencias Informáticas (UCI), pone a disposición del alumnado un nuevo módulo desarrollado a partir de la necesidad de conocer las calificaciones obtenidas por un estudiante durante la carrera.

Es un sistema de gestión académica que permite gestionar la información de los estudiantes en la Universidad de las Ciencias Informáticas, está dividido por módulos y define niveles de acceso de tal manera que cada cual pueda visualizar sólo aquella información relacionada con su función. Este cuenta con numerosas funcionalidades que permiten la gestión de la información, se almacenan los datos por facultad de los resultados obtenidos por los estudiantes en las pruebas realizadas, la matrícula con que se cuenta, asistencia a clases, las causas que describen la situación del ausente, los resultados obtenidos en los cursos optativos, la evaluación de los estudiantes en los diferentes cortes, entre otras. Permitiendo así mantener informado, a los estudiantes sobre su desempeño académico y a los profesores sobre los resultados de sus estudiantes.



1.5 Conclusiones.

En el presente capítulo se especificaron los conceptos fundamentales que se abordan en la situación problémica. Este abordó el objeto de estudio ejemplificando el principal elemento que lo integra. Describió el Vicedecanato de Formación y la Secretaría Docente especificando el personal que participa en ellos así como una descripción general de los procesos que allí se realizan.

Además detalló la situación problémica, especificando como se producen actualmente los procesos que se llevan a cabo en el Vicedecanato de Formación, la Planificación Docente y la Secretaría Docente de la Facultad 9, demostrando así por qué se necesita el sistema. Analizó las principales soluciones existentes encontradas las cuales servirán de guía para el sistema a desarrollar.



CAPÍTULO 2

Tendencias y Tecnologías Actuales a Desarrollar para el Subsistema Docente de la Facultad 9.

2.1 Introducción.

Para el desarrollo del sistema funcional del Subsistema Docente de la Facultad 9 se realiza un estudio profundo sobre las Tecnologías de la Información y las Comunicaciones (TIC) y las aplicaciones Web, en Cuba y el mundo. Se realizó un análisis de las principales y más actuales tecnologías utilizadas para el diseño y desarrollo de aplicaciones Web. Se trata además los lenguajes de programación Web y los gestores de Base Datos más utilizados actualmente en el mundo para el desarrollo de este tipo de aplicaciones, finalmente según las condiciones de trabajo existentes se seleccionan la tecnología y la metodología de desarrollo de software a usar.

2.2 Las Tecnologías de la Información y las Comunicaciones (TIC).

Las Tecnologías de la Información y las Comunicaciones (TIC), son el conjunto de tecnologías que permiten la adquisición, producción, almacenamiento, tratamiento, comunicación, registro y presentación de informaciones, en forma de voz, imágenes y datos contenidos en señales de naturaleza acústica, óptica o electromagnética. Las mismas incluyen la electrónica como tecnología base que soporta el desarrollo de las telecomunicaciones, la informática y el audiovisual.(9)

Las aplicaciones de las TIC se encuentran presentes en multitud de ámbitos de la vida cotidiana, siendo muy difícil encontrar alguna actividad en la que no aparezca de una u otra forma algún aspecto relacionado con estas tecnologías. El desarrollo alcanzado por las TIC en los últimos años es espectacular, tanto a nivel de desarrollo tecnológico como de oferta de nuevos servicios.

La importancia de las TIC no es la tecnología en sí, sino el hecho de que permita el acceso al conocimiento, la información, y las comunicaciones: elementos cada vez más importantes en la interacción económica y social de los tiempos actuales.(10)

2.3 Aplicaciones Web.

Conocidas además como aplicaciones de servidor debido a que el procesamiento de la información se realiza en otra computadora llamada servidor. Las computadoras clientes se comunican con el servidor



a través de un navegador Web. El servidor puede interactuar con gran cantidad de computadoras clientes al mismo tiempo, estas pueden utilizar los mismos datos y una misma aplicación, desde diferentes lugares sin más requerimientos que un navegador Web y conexión a Internet u otra red compartida.

Las aplicaciones Web ofrecen ventajas a quienes las utilizan:

- No requieren instalación.
- Son fáciles de usar.
- Alta disponibilidad.

La Aplicación Web a realizar utilizará la misma arquitectura que la gran mayoría de las aplicaciones Web del mundo, Cliente – Servidor. Este tipo de arquitectura es la más utilizada en la actualidad, debido a que es la más avanzada y la que mejor ha evolucionado en estos últimos años.

La arquitectura Cliente – Servidor es un modelo para el desarrollo de sistemas de información en el que las transacciones se dividen en procesos independientes que cooperan entre sí para intercambiar información, servicios o recursos.

Se denomina:

Cliente al proceso que inicia el diálogo o solicita los recursos, los clientes interactúan con el usuario, usualmente en forma gráfica, frecuentemente se comunican con procesos auxiliares que se encargan de establecer conexión con el servidor, enviar el pedido, recibir la respuesta, manejar las fallas y realizar actividades de sincronización y de seguridad.

Servidor al proceso que responde a las solicitudes, los servidores proporcionan un servicio al cliente y devuelven los resultados, deben manejar servicios como administración de la red, mensajes, control y administración de la entrada al sistema, auditoría, recuperación y contabilidad. Usualmente en los servidores existe algún tipo de servicio de bases de datos.

Entre las principales características de la arquitectura Cliente – Servidor están:

- El cliente no necesita conocer la lógica del servidor, sólo su interfaz externa.
- El cliente no depende de la ubicación física del servidor, ni del tipo de equipo físico en el que se encuentra, ni de su sistema operativo.

- El servidor presenta a todos sus clientes una interfaz única y bien definida.
- Los cambios en el servidor implican pocos o ningún cambio en el cliente.

Una representación gráfica de este tipo de arquitectura sería:

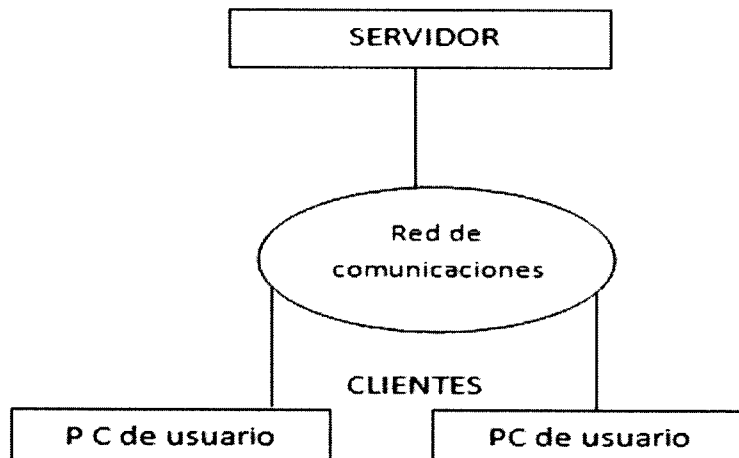


Figura 1: Arquitectura Cliente/ Servidor.

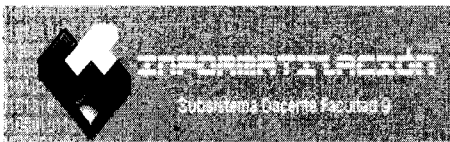
2.4 *Lenguajes de Programación Web.*

Lenguaje de Programación es aquel lenguaje con el que se escriben los programas y que las computadoras entienden.

2.4.1 *Lenguajes de Programación Web del lado del Cliente.*

2.4.1.1 *JavaScript.*

Es un lenguaje interpretado, es decir, que no requiere compilación, utilizado principalmente en páginas Web, con una sintaxis semejante a la del lenguaje Java y el lenguaje C. No es un lenguaje orientado a objetos, ya que no dispone de herencia, es más bien un lenguaje basado en prototipos, ya que las nuevas clases se generan clonando las clases base y extendiendo su funcionalidad. JavaScript se venía utilizando en páginas Web HTML, para realizar tareas y operaciones en el marco de una aplicación cliente, sin acceso a funciones del servidor. En junio de 1997 fue adoptado como un estándar ECMA, con el nombre de ECMAScript. Poco después también lo fue como un estándar ISO.



2.4.1.2 Hypertext Markup Language (HTML).

Es un lenguaje de composición de documentos y especificación de ligas de hipertexto que define la sintaxis y coloca instrucciones especiales que no muestra el navegador, aunque si le indica como desplegar el contenido del documento, incluyendo texto, imágenes y otros medios soportados.(11)

Es un lenguaje de marcado, muy estructurado, que se utiliza para crear páginas Web. Un lenguaje de marcado es, sencillamente, un conjunto de códigos, llamados elementos, que se utilizan para indicar la estructura y, frecuentemente, el formato de un documento. Un explorador Web que representa el documento interpreta el significado de esos códigos para averiguar como estructurar o mostrar el documento.(12)

HTML está diseñado para estructurar documentos y hacerlos más accesibles, no para darles formato con el propósito de obtener diseños espectaculares.

2.4.2 Lenguajes de Programación Web del lado del Servidor.

2.4.2.1 "PHP Hypertext Pre-processor" (PHP).

Es un lenguaje de programación orientado a objetos usado normalmente para la creación de páginas Web dinámicas del lado del servidor, habitualmente en combinación con el motor de base datos MySQL, aunque cuenta con soporte nativo para otros motores. PHP se trata de un lenguaje interpretado y de código abierto. Puede ser utilizado para la creación de otro tipo de programas incluyendo aplicaciones con interfaz gráfica. Tiene la capacidad de ser ejecutado en la mayoría de los sistemas operativos tales como UNIX y Windows, y puede interactuar con los servidores de Web más populares.

Es un lenguaje de secuencia de comandos de servidor diseñado específicamente para la Web. Dentro de una página Web se puede incrustar código PHP que se ejecutará cada vez que se visite esa página. El código PHP es interpretado en el servidor Web y genera código HTML y otro contenido que el visitante verá.(13)

2.4.2.2 "Active Server Pages" (ASP).

Es una tecnología del lado de servidor desarrollada por Microsoft para el desarrollo de sitios Web dinámicos. ASP no necesita ser compilado para ejecutarse. Existen varios lenguajes que se pueden



utilizar para crear páginas ASP, el más utilizado es VBScript, nativo de Microsoft. ASP. El código ASP puede ser insertado junto con el código HTML.

Ventajas:

- Comunicación óptima con SQL Server.
- Soporta el lenguaje Javascript.

Desventajas:

- Código desorganizado.
- Se necesita escribir mucho código para realizar funciones sencillas. (14)

2.4.2.3 Java Server Pages (JSP).

Es un lenguaje para la creación de sitios Web dinámicos. Está orientado a desarrollar páginas Web en Java. JSP es un lenguaje multiplataforma, creado para ejecutarse del lado del servidor y desarrollado para la creación de aplicaciones Web potentes. Para su funcionamiento se necesita tener instalado un servidor Tomcat.

JSP posee características como:

- Código separado de la lógica del programa.
- Las páginas son compiladas en la primera petición.
- Permite separar la parte dinámica de la estática en las páginas Web.
- Los archivos se encuentran con la extensión (jsp).
- El código JSP puede ser incrustado en código HTML.

Ventajas:

- Código bien estructurado.
- La parte dinámica está escrita en Java.

Desventajas:

- Complejidad de aprendizaje. (14)

2.5 ¿Por qué se utilizó JavaScript como Lenguaje para Validar?

La utilización del lenguaje JavaScript brinda diferentes ventajas como:

- **Fácil de aprender, rápido y potente.**

JavaScript es muy sencillo de aprender. Es ideal para agregar ciertas funciones rápidas a una página Web. Es un lenguaje muy potente de alto nivel. No necesita una fase de compilación como Java o C, sólo hay que crear el código y cargarlo.

- **Usabilidad**

JavaScript es uno de los lenguajes de programación que más se utiliza en la Web. Hay publicadas millones de páginas Web que incorporan elementos que lo usan. Una de las principales razones por la que los desarrolladores Web han adoptado este lenguaje es porque reduce la carga del servidor.

- **Reducción de la Carga del Servidor**

JavaScript se puede hacer cargo de gran parte de las funciones del cliente de las cuales se encargaba el servidor. Uno de los mejores ejemplos es la validación.

Con JavaScript es posible validar los elementos antes de que el usuario los envíe al servidor. De esta forma se reduce la cantidad de transacciones que se efectúan a través de http y las posibilidades de que se genere un error durante la inserción de datos.(15)

2.6 ¿Por qué se utilizó HTML?

Se utiliza HTML porque ofrece diferentes maneras de definir la apariencia de los documentos: especificaciones de tipografía, saltos de línea y texto con formato previo son, todos, funciones del lenguaje.

Además, tiene muchas funciones para estructurar el contenido de los documentos sin importar la apariencia final: encabezados de sección, listas estructuradas, párrafos, líneas horizontales, títulos e imágenes, todos estos elementos quedan definidos por HTML sin importar como los debe presentar el navegador. (11)

Fue diseñado para indicar la estructura de un documento para que el navegador pueda presentarlo de forma apropiada, contiene gran cantidad de etiquetas que le permiten indicar la semántica del contenido de su documento.

2.7 ¿Por qué se utilizó PHP como Lenguaje de Programación Web?

Para desarrollar el sistema funcional se decidió utilizar PHP como lenguaje de programación ya que debido a su uso fácil permite a los programadores crear aplicaciones complejas y de contenido dinámico con mucha facilidad. Posee una biblioteca nativa de funciones sumamente amplia e incluida y



no requiere definición de tipos de variables. Permite la conexión a diferentes tipos de servidores de bases de datos tales como MySQL, Postgres, Oracle, Microsoft SQL Server; lo cual permite la creación de Aplicaciones Web muy robustas.

Además PHP cuenta con las siguientes ventajas:

- Alto rendimiento.

Es muy eficiente. Mediante el uso de un único servidor, puede servir millones de acceso al día.

- Interfaces para una gran cantidad de sistemas de base de datos diferentes.

Dispone de una conexión propia a todos los sistemas de base de datos. Además de MySQL, puede conectarse directamente a las bases de datos de PostgreSQL, Oracle, filepro, Hyperwave, Informix, InterBase y Sybase, entre otras. El uso de ODBC (Open Database Connectivity Standard, estándar de conectividad abierta de base de datos) permite establecer una conexión a cualquier base de datos que suministre un controlador ODBC.

- Bibliotecas incorporadas para muchas tareas Web habituales.

Incorpora una gran cantidad de funciones integradas para realizar útiles tareas relacionadas con la Web, como por ejemplo generar documentos PDF.

- Bajo coste.

Es gratuito.

- Facilidad de aprendizaje y uso.

La sintaxis de PHP se basa en otros lenguajes de programación, principalmente en C o algún lenguaje de tipo C como C++ o Java y Perl.

- Portabilidad.

Está disponible para una gran cantidad de sistemas operativos diferentes. Puede escribir código PHP en todos los sistemas operativos gratuitos del tipo Unix, como Linux y FreeBSD, versiones comerciales de Unix, como Solaris e IRIX o en las diferentes versiones de Microsoft Windows. Su código funcionará sin necesidad de aplicar ninguna modificación a los diferentes sistemas que ejecute PHP.(13)

2.8 Gestores de Base Datos.

El Sistema Gestor de Base de Datos (DBMS) es un conjunto de programas que se encargan de manejar la creación y todos los accesos a las bases de datos. El objetivo principal del sistema gestor de base de datos es proporcionar una forma de almacenar y recuperar la información de una base de datos de manera que sea tanto práctica como eficiente. Se compone de un lenguaje de definición de



datos (DDL: Data Definition Language), de un lenguaje de manipulación de datos (DML: Data Manipulation Language) y de un lenguaje de consulta (SQL : Structured Query Language).(16)

2.8.1 Oracle.

Es un sistema de administración de base de datos (RDBMS Relational Data Base Management System), fabricado por Oracle Corporation, básicamente una herramienta cliente/servidor para la gestión de Bases de Datos. Es un producto vendido a nivel mundial, aunque la gran potencia que tiene y su elevado precio hacen que sólo se vea en empresas muy grandes y multinacionales, por norma general.

Es sin duda una de las mejores bases de datos que tiene el mercado, es un sistema gestor de base de datos robusto, tiene muchas características que garantizan la seguridad e integridad de los datos; que las transacciones se ejecuten de forma correcta, sin causar inconsistencias; ayuda a administrar y almacenar grandes volúmenes de datos, es estable y multiplataforma.(16)

Algunos aspectos de la Arquitectura de un servidor Oracle:

- Por cada instancia de Oracle se tiene una sola base de datos.
- La Base de Datos: dentro de los procesos que forman parte de la base de datos tenemos:
 1. Control File, que contiene la información para controlar y mantener la integridad de la base de datos.
 2. Data files, son los archivos en los cuales se almacenan los datos de las aplicaciones.
 3. Redo Log Files, almacena los cambios hechos en la base de datos con propósito de recuperarlos en caso de falla.
- Estructuras Adicionales:
 1. Parameter File, contiene parámetros y valores que definen las características de la instancia y de la base de datos.
 2. Password File, se utiliza para validar al usuario que puede bajar y subir la instancia Oracle.
 3. Achived Log Files, son copias fuera de línea de los archivos Redo Log Files que son necesarios para el proceso de recuperación en caso de falla del medio de almacenamiento.(16)



2.8.2 MySQL.

Es un sistema para la administración de bases de datos relacional (RDBMS) rápido y sólido. Las bases de datos permiten almacenar, buscar, ordenar y recuperar datos de forma eficiente. El servidor de MySQL controla el acceso a los datos para garantizar el uso simultáneo de varios usuarios, para proporcionar acceso a dichos datos y para asegurarse de que sólo obtienen acceso a ellos los usuarios con autorización. Por lo tanto, es un servidor multiusuario y de subprocesamiento múltiple.(13)

Es muy utilizado en aplicaciones Web, en plataformas (Linux/Windows-Apache-MySQL-PHP/Perl/Python), y por herramientas de seguimiento de errores como Bugzilla. Es propiedad y está patrocinado por una empresa privada, que posee el copyright de la mayor parte del código. Está escrito en una mezcla de C y C++. En aplicaciones Web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones. Dispone de una gran cantidad de plataformas y sistemas, además brinda una conectividad segura.

Posee características que son implementadas únicamente por él como:

- Múltiples motores de almacenamiento lo cual permite al usuario escoger el que sea más adecuado para cada tabla de la base de datos.
- Agrupación de transacciones, es decir reúne múltiples transacciones de varias conexiones para así incrementar el número de transacciones realizadas por segundo.

MySQL cuenta con las siguientes ventajas:

- Alto Rendimiento.

Es muy rápido.

- Bajo Coste.

Está disponible de manera gratuita, bajo una licencia de código abierto, o por un precio reducido en forma de licencia comercial si resultara necesario para su aplicación

- Facilidad de Configuración y Aprendizaje.

Las bases de datos más modernas utilizan SQL. MySQL resulta además más sencillo de configurar que otros productos similares.

- Portabilidad



Se puede utilizar en una gran cantidad de sistemas Unix diferentes, así como bajo Microsoft Windows.(13)

2.8.3 PostgreSQL.

Es un Sistema de Gestión de Bases de Datos Objeto-Relacionales (ORDBMS) ha sido desarrollado de varias formas desde 1977. Comenzó como un proyecto denominado Ingres en la Universidad Berkeley de California. En 1986 otro equipo continuó el desarrollo del código de Ingres para crear un sistema de bases de datos objeto-relacionales llamado Postgres. En 1994 se añadió un intérprete de lenguaje SQL a Postgres. En 1996, debido a un nuevo esfuerzo de código abierto y a la incrementada funcionalidad del software, Postgres fue renombrado a PostgreSQL.

Las principales mejoras en PostgreSQL incluyen:

- Los bloqueos de tabla han sido sustituidos por el control de concurrencia multi-versión, el cual permite a los accesos de sólo lectura continuar leyendo datos consistentes durante la actualización de registro.
 - Se han implementado importantes características del motor de datos, incluyendo subconsultas y valores por defecto.
 - Se han añadido funcionalidades en línea con el estándar SQL92, incluyendo claves primarias, identificadores entrecomillados, forzado de tipos cadena literal, conversión de tipos y entrada de enteros binarios y hexadecimales.
 - Los tipos internos han sido mejorados, incluyendo nuevos tipos de fecha/hora de rango amplio.
- (17)

El proyecto sigue actualmente un activo proceso de desarrollo a nivel mundial gracias a un equipo de desarrolladores y contribuidores de código abierto.

Algunas características básicas de PostgreSQL son:

1. Alta concurrencia: Mediante un sistema denominado MVCC (Acceso concurrente multiversión) Permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Esta estrategia es superior al uso de bloqueos por tabla o por filas común en otras bases de datos, eliminando la necesidad del uso de bloqueos explícitos.
2. Amplia variedad de tipos nativos: PostgreSQL provee nativamente soporte para:
 - Números de precisión arbitraria.



- Texto de largo ilimitado.
- Arrays.

2.9 ¿Por qué se utilizó PostgreSQL como Gestor de Base Datos?

Para el desarrollo de este sistema se decidió utilizar PostgreSQL como Gestor de Base Datos porque está ampliamente considerado como el sistema de bases de datos de código abierto más avanzado del mundo. Posee muchas características que tradicionalmente sólo se podían ver en productos comerciales de alto calibre tales como:

- Consultas SQL declarativas, soporte multi-usuario, optimización de consultas, herencia, y arrays.
- Soporta operadores y tipos de datos definidos por el usuario lo que lo hace altamente extensible.
- Soporta la especificación SQL99 e incluye características avanzadas tales como las uniones (joins) SQL92.
- Soporta integridad referencial, la cual es utilizada para garantizar la validez de los datos de la base de datos.
- Usa una arquitectura cliente/servidor. (18)

2.10 El Lenguaje Unificado de Modelado (UML) como soporte de la modelación de la solución propuesta.

UML es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra una gran cantidad de software. (19)

Tiene una gran cantidad de propiedades que han sido las que han contribuido a hacer de este lenguaje de modelado un estándar de la industria del software. Algunas de estas propiedades son:

- Modela estructuras complejas.
- Las estructuras más importantes que soportan tienen su fundamento en las tecnologías orientadas a objetos, tales como objetos, clase, componentes y nodos.
- Comportamiento del sistema: casos de uso, diagramas de secuencia y de colaboraciones, que sirven para evaluar el estado de las máquinas. (19)



El Modelo Gráfico de UML tiene un vocabulario en el que se identifican:

- Elementos: Abstracciones que constituyen los bloques básicos de construcción.
 1. Estructurales: Partes que representan cosas.
 - Clase: Conjunto de objetos que comparten atributos, operaciones, relaciones y semántica.
 - Colaboración: Colección de operaciones que especifican un servicio de una clase o un componente.
 - Colaboración: Define la interacción entre los elementos que proporcionan un comportamiento cooperativo mayor que la suma de los comportamientos de sus elementos.
 - Caso de uso: Conjunto de secuencia de acciones que un sistema ejecuta y que produce un resultado observable para un actor.
 - Clase activa: Clase cuyos objetos tienen uno o más procesos o hilos de ejecución.
 - Componente: Es una parte física y reemplazable de un sistema que conforma un conjunto de interfaces y proporciona la implementación de dicho conjunto.
 - Nodo: Elemento físico que dispone de memoria y con frecuencia capacidad de almacenamiento.
 2. Comportamiento: Partes del modelo que representan el comportamiento en el tiempo y el espacio.
 - Interacción: Conjunto de mensajes intercambiados entre un conjunto de objetos para alcanzar un propósito específico.
 - Máquina de estado: Especifica las secuencias de estados por las que pasa un objeto o una interacción durante su vida.
 3. Agrupamiento: Cajas en las cuales puede descomponerse un modelo.
 - Paquete: Mecanismo de propósito general para organizar elementos en grupos.
 - Modelo.
 - Subsistema.
 - Marco de Trabajo.
 4. Anotación: Comentarios que se pueden aplicar para describir, clarificar y hacer observaciones sobre cualquier elemento de un modelo.
 - Nota

- Relaciones: Ligan los elementos.
 1. Dependencia: Relación semántica que indica que un cambio en un elemento afecta a la semántica de otro elemento.
 2. Asociación: Relación estructural que describe las conexiones entre objetos.
 3. Generalización/Especialización: Relación en la que el hijo comparte la estructura y el comportamiento del padre.
 4. Realización: Relación semántica entre clasificadores, en donde un clasificador especifica un contrato que otro clasificador garantiza que cumplirá.
- Diagramas: Es la representación gráfica de un conjunto de elementos. Visualizan un sistema desde diferentes perspectivas.
 1. Diagramas de estructura estática: Describen las propiedades estructurales del sistema.
 - Diagrama de Clases: Conjunto de clases, interfaces y colaboraciones; así como sus colaboraciones.
 - Diagrama de Objetos: Conjunto de objetos y sus relaciones.
 - Diagrama de Casos de Uso: Conjunto de casos de uso y actores y sus relaciones.
 2. Diagramas de comportamiento:
 - Diagramas de Interacción (secuencia y colaboración): Objetos y sus relaciones, incluyendo los mensajes que pueden ser enviados entre ellos.
 - Diagrama de Estados: Muestra una máquina de estado que consta de estados, transiciones, eventos y actividades.
 - Diagrama de Actividad: Es un tipo especial de diagrama de estados que muestra el flujo de actividades dentro de un sistema.
 3. Diagramas de implementación:
 - Diagrama de Componentes: Organización y las dependencias entre un conjunto de componentes.
 - Diagrama de Despliegue: Configuración de nodos de procesamiento en tiempo de ejecución y los componentes que residen en ellos. (20)

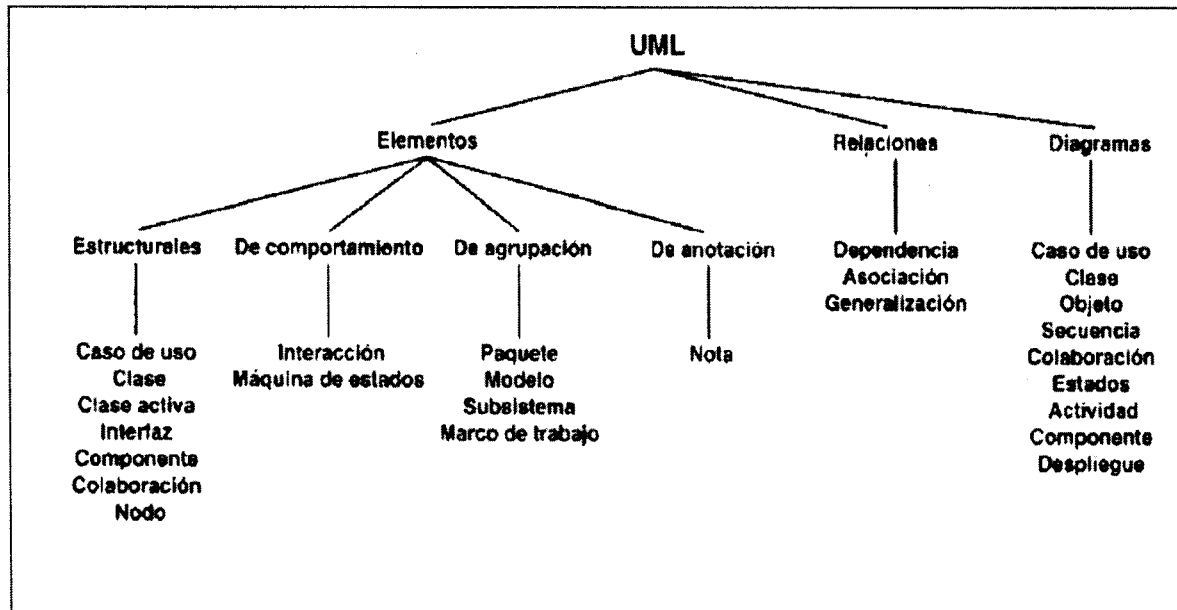


Figura 2: El Vocabulario de UML.

2.11 Metodologías de Desarrollo de Software.

Metodología:

En un proyecto de desarrollo de software la metodología define Quién debe hacer Qué, Cuándo y Cómo debe hacerlo. Una metodología es un proceso. (20)

Conjunto de pasos y procedimientos que deben seguirse para el desarrollo de software.(21)

Conjunto de procedimientos, técnicas, herramientas y soporte documental que ayuda a los desarrolladores a realizar nuevo software.(21)

2.11.1 Métrica 3: Metodología de Planificación, Desarrollo y Mantenimiento de Sistemas de Información.

La metodología MÉTRICA Versión 3 ofrece a las organizaciones un instrumento útil para la sistematización de las actividades que dan soporte al ciclo de vida del software dentro del marco que permite alcanzar los siguientes objetivos:

- Proporcionar o definir sistemas de información que ayuden a conseguir los fines de la organización mediante la definición de un marco estratégico para el desarrollo de los mismos.
- Dotar a la organización de productos software que satisfagan las necesidades de los usuarios dando una mayor importancia al análisis de requisitos.
- Facilitar la operación, mantenimiento y uso de los productos software obtenido. (22)

La Metodología Métrica 3:

- Contempla el desarrollo de Sistemas de Información para las distintas tecnologías y los aspectos de gestión que aseguran que un proyecto cumple sus objetivos en términos de calidad, coste y plazos.
- Se conserva la adaptabilidad, flexibilidad y sencillez, así como la estructura de actividades y tareas.
- Se facilita a través de interfaces la realización de los procesos de apoyo u organizativos: Gestión de Proyectos, Gestión de Configuración, Aseguramiento de Calidad y Seguridad.
- Procesos Principales:
 1. Planificación de Sistemas de Información.

Obtención de un marco de referencia para el desarrollo de Sistemas de Información que responda a los objetivos estratégicos de la organización.

2. Desarrollo de Sistemas de Información.

El Proceso de Desarrollo de Sistemas de Información, se ha subdividido en cinco procesos para facilitar la comprensión, dada su amplitud y complejidad:

- Estudio de Viabilidad del Sistema (EVS).

Análisis de un conjunto concreto de necesidades para proponer una solución a corto plazo, que tenga en cuenta restricciones económicas, técnicas, legales y operativas.

- Análisis del Sistema de Información (ASI).

Obtención de una especificación detallada del Sistema de Información que satisfaga las necesidades de información de los usuarios y sirva de base para el posterior diseño del sistema.

- Diseño del Sistema de Información (DSI).

Definición de la arquitectura del sistema y del entorno tecnológico que le va a dar soporte, junto con la especificación detallada de los componentes del sistema de información.

- Construcción del Sistema de Información (CSI).

Se genera el código de los componentes del Sistema de Información, se desarrollan todos los procedimientos de operación y seguridad y se elaboran todos los manuales de usuario final y de explotación con el objetivo de asegurar el correcto funcionamiento del sistema para su posterior implantación.

- Implantación y Aceptación del Sistema (IAS).

Entrega y aceptación del sistema en su totalidad, y la realización de todas las actividades necesarias para dar paso a la producción del mismo.

3. Mantenimiento de Sistemas de Información.

Obtención de una nueva versión de un Sistema de Información desarrollado con MÉTRICA v.3, a partir de las peticiones de mantenimiento que los usuarios realizan con motivo de un problema detectado en el sistema, o por la necesidad de una mejora del mismo.

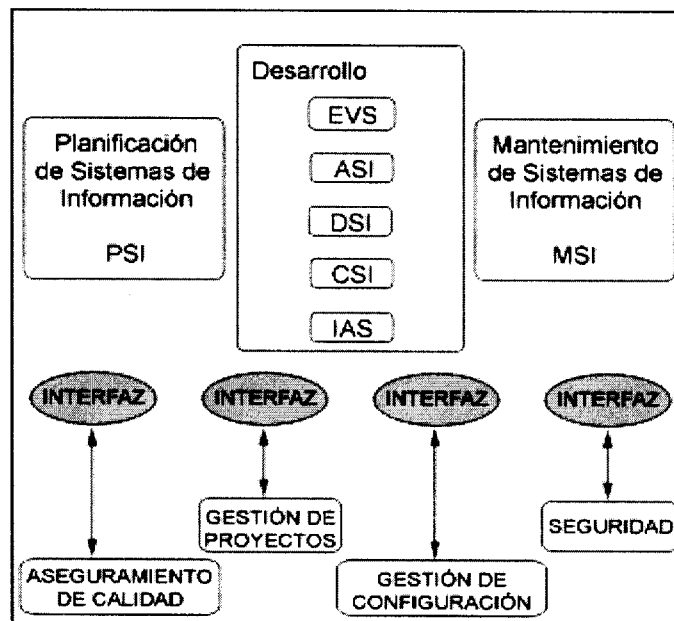


Figura 3: Interfaces y Procesos Principales de Métrica 3.

2.11.2 Metodología Extreme Programming (XP).

La Programación Extrema es una metodología ligera de desarrollo de software que se basa en la simplicidad al desarrollar y codificar los módulos del sistema, la comunicación entre los usuarios y los desarrolladores y la realimentación o reutilización del código desarrollado.(23)

Es una de las metodologías de desarrollo de software más exitosas en la actualidad utilizadas para proyectos de corto plazo, corto equipo y cuyo plazo de entrega era ayer. La metodología consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo, al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto. XP se plantea como una metodología a emplear en proyectos de riesgo, aumenta la productividad.

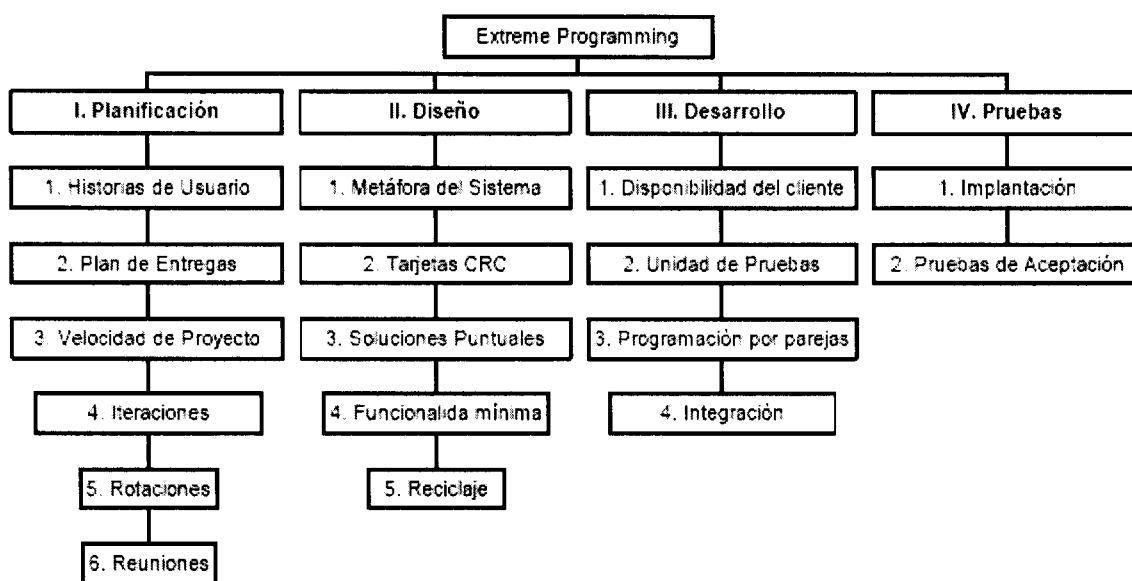


Figura 4: Fases de la Metodología XP.

2.11.3 El Proceso Unificado de Desarrollo de Software (RUP).

El Proceso Unificado de Desarrollo de Software es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas software, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitud y diferentes tamaños de proyecto. (24)

Las piedras angulares del proceso de desarrollo del software son: el proyecto, las personas, el producto y el proceso; existe una estrecha relación entre ellas. Se les conoce como las cuatro P en el desarrollo del software:

- **Personas:** Los principales autores de un proyecto software son los arquitectos, desarrolladores, ingenieros de prueba, y el personal de gestión que les da soporte, además de los usuarios,



clientes, y otros interesados. Las personas son realmente seres humanos, a diferencia del término abstracto trabajadores, que será introducido más adelante.

- Proyecto: Elemento organizativo a través del cual se gestiona el desarrollo de software. El resultado de un proyecto es una versión de un producto.
- Producto: Artefacto que se crea durante la vida del proyecto, como código fuente, código ejecutable, modelos y documentación.
- Proceso: Un proceso de ingeniería de software es una definición del conjunto completo de actividades necesarias para transformar los requisitos de usuario en un producto. Un proceso es una plantilla para crear proyectos.

Principales Elementos:

- Trabajadores (“quién”): Define el comportamiento y responsabilidades (rol) de un individuo, grupo de individuos, sistema automatizado o máquina, que trabajan en conjunto como un equipo. Ellos realizan las actividades y son propietarios de elementos.
- Actividades (“cómo”): Es una tarea que tiene un propósito claro, es realizada por un trabajador y manipula elementos.
- Artefactos (“qué”): Productos tangibles del proyecto que son producidos, modificados y usados por las actividades. Pueden ser modelos, elementos dentro del modelo, código fuente y ejecutables.
- Flujo de actividades (“Cuándo”): Secuencia de actividades realizadas por trabajadores y que produce un resultado de valor observable.

RUP se divide en 4 fases, cada una cuenta con un objetivo principal:

- Inicio: El objetivo es determinar la visión del proyecto.
- Elaboración: El objetivo es determinar la arquitectura óptima.
- Construcción: El objetivo es obtener la capacidad operacional inicial.
- Transición: El objetivo es llegar a obtener el release del proyecto.

Flujos de Trabajo de Desarrollo

- Modelamiento del Negocio: En este flujo se entienden las necesidades del negocio.
- Requerimientos: En este flujo se trasladan las necesidades del negocio a un sistema automatizado.

- Análisis y Diseño: En este flujo se trasladan los requerimientos dentro de la arquitectura de software.
- Implementación: En este flujo se crea el software ajustado a la arquitectura y que tenga el comportamiento deseado.
- Pruebas: En este flujo se asegura que el comportamiento requerido es el correcto y que todo lo solicitado esta presente.

Flujos de Trabajo de Soporte

- Instalación: Se realizan actividades como: empaque, instalación, asistencia a usuarios, etc., para entregar el software a los usuarios finales.
- Administración de Configuración y Cambio: Guardando todas las versiones del proyecto.
- Administración del Proyecto: Administrando horarios y recursos.
- Ambiente: Administrando el ambiente de desarrollo.

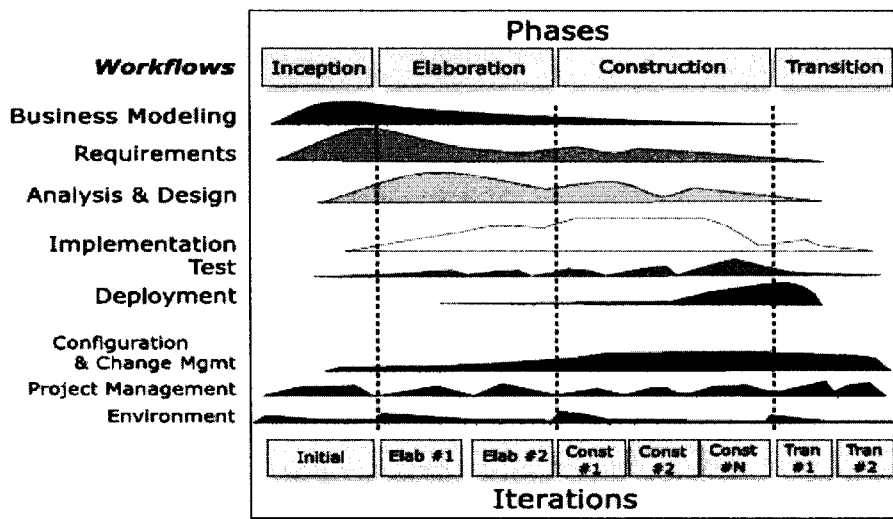


Figura 5: Los Flujos de Trabajo tienen lugar sobre las cuatro fases: Inicio, Elaboración, Construcción y Transición.

2.12 ¿Por qué se utilizó El Proceso Unificado de Desarrollo de Software (RUP) como base en el desarrollo de la solución?

El Proceso Unificado está basado en componentes. Utiliza el nuevo estándar de modelación visual, el Lenguaje Unificado de Modelado (UML), y se sostiene sobre tres ideas básicas -casos de uso,



arquitectura y desarrollo iterativo e incremental. Para hacer que estas ideas funcionen, se necesita un proceso polifacético, que tenga en cuenta ciclos, fases, flujos de trabajo, gestión de riesgos, control de calidad, gestión de proyecto y control de configuración. El Proceso Unificado ha establecido un marco de trabajo que integra todas esas diferentes facetas. (24)

Analizando lo citado anteriormente se puede deducir porque se escoge RUP como metodología de desarrollo y no otra. Ejemplificando se puede decir que RUP unifica los mejores elementos de metodologías anteriores lo que garantiza la seguridad de que es una metodología de punta, beneficia en ser una metodología orientada a objetos y que esta preparada para desarrollar grandes y complejos proyectos. Al estar basada en componentes da la posibilidad de construir un sistema software formado por componentes software interconectados a través de interfaces bien definidas. Además utiliza el Lenguaje Unificado del Modelado para preparar todos los esquemas de un sistema software y han sido explicadas en un epígrafe anterior las ventajas de utilizar este lenguaje de modelado.

2.13 Frameworks para PHP.

Un framework es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado. Un framework puede incluir soporte de programas, bibliotecas, entre otros software para ayudar a desarrollar y unir los diferentes componentes de un proyecto.

Un framework representa una arquitectura de software que modela las relaciones generales de las entidades del dominio. Provee una estructura y una metodología de trabajo la cual extiende o utiliza las aplicaciones del dominio.

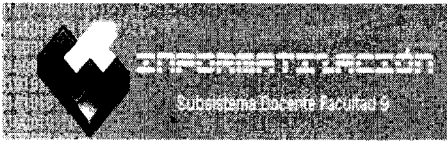
2.13.1 Kumbia.

Kumbia es un Web framework libre escrito en PHP5. Basado en las mejores prácticas de desarrollo Web, usado en software comercial y educativo, fomenta la velocidad y eficiencia en la creación y mantenimiento de aplicaciones Web. (25)

Es compatible con las bases de datos disponibles actuales más usadas:

- MySQL.
- PostgreSQL.
- Oracle. (26)

Principales Características con que cuenta:



- Sistema de Plantillas sencillo.
- Modelo de Objetos y Separación Modelo Vista Controlador.
- Generación de Formularios.
- Componentes Gráficos.
- Seguridad.

2.13.2 CakePHP.

CakePHP es un framework libre y de código abierto para el desarrollo en PHP. Es una estructura de librerías y clases para programadores de aplicaciones Web. Originalmente inspirado en el framework Ruby On Rails. Su meta es permitir el trabajo de manera estructurada, rápida y sin pérdida de flexibilidad. (27)

Características más destacadas:

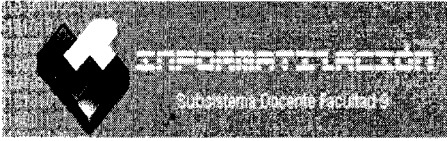
- Arquitectura Modelo Vista Controlador.
- Listas de Control de Acceso.
- Orientado a Objetos.
- Compatibilidad con PHP4 y PHP5.
- Validación Incorporada.
- Plantillas rápidas y flexibles.
- Seguridad, Sesiones y Componentes para Manejo de Peticiones.

2.13.3 Symfony.

Es un framework para construir aplicaciones Web en lenguaje PHP. Es un enorme conjunto de herramientas y utilidades que simplifican el desarrollo de las aplicaciones Web.

Esta basado en las mejores ideas de diferentes framework como: Rails, su primera versión se publicó en octubre del 2005.

Está desarrollado completamente con PHP 5. Es compatible con la mayoría de gestores de bases de datos, como MySQL, PostgreSQL, Oracle y SQL Server de Microsoft. Se puede ejecutar tanto en plataformas *nix (Unix, Linux, etc.) como en plataformas Windows.



Emplea el tradicional patrón de diseño **Modelo-Vista-Controlador** para separar las distintas partes que forman una aplicación Web.

La utilización de este patrón permite separar los datos, la interfaz de usuario y la lógica de negocio en 3 componentes distintos.

- **Modelo:** El modelo representa la información con la que trabaja la aplicación incluye la lógica de datos que asegura la integridad de la información y la derivación de nuevos datos.
- **Vista:** La vista transforma la información obtenida por el modelo en las páginas Web a las que acceden los usuarios. Escucha los cambios en el modelo y se comunica con el modelo usando el controlador.
- **Controlador:** El controlador es el encargado de coordinar todos los demás elementos y transformar las peticiones del usuario en operaciones sobre el modelo y la vista.

Este patrón posee un orden ascendente de reusabilidad: Modelo, Controlador y Vista lo que implica que el modelo debe de ser independiente, las clases del modelo no deben ver ninguna otra clase de otro grupo. El controlador podría y suele ver clases del modelo, pero no de la vista. Esto permite el cambio de interfaz sin afectar la lógica de negocio. Las clases de la Vista podrán ver las clases del modelo y del controlador.

Ventajas:

- Múltiples vistas del mismo modelo.
- Vistas sincronizadas.
- Flexibilidad para el cambio.
- Distintos tipos de interfaz.

2.14 ¿Por qué se utiliza Symfony como framework ?

Se decidió Symfony como framework ha utilizar porque es uno de los frameworks de PHP más populares entre los usuarios y las empresas, ya que permite crear código de más calidad y más fácil de mantener.

Es un completo framework diseñado para optimizar, el desarrollo de las aplicaciones Web. Separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación Web. Proporciona varias



herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación Web compleja.(28)

Se escogió Symfony porque utiliza el patrón arquitectónico escogido para utilizar en la aplicación: Modelo-Vista-Controlador además porque automatiza la mayoría de elementos comunes de los proyectos Web, como por ejemplo:

- La capa de presentación utiliza plantillas y layout que pueden ser creados por diseñadores HTML sin ningún tipo de conocimiento del framework. Los helpers incluidos permiten minimizar el código utilizado en la presentación, ya que encapsulan grandes bloques de código en llamadas simples a funciones.
- Los formularios incluyen validación automatizada y relleno automático de datos lo que asegura la obtención de datos correctos.
- La autenticación y la gestión de credenciales simplifican la creación de secciones restringidas y la gestión de la seguridad de usuario.
- Los listados son más fáciles de utilizar debido a la paginación automatizada, el filtrado y la ordenación de datos. (28)

2.15 Herramientas de Modelado.

2.15.1 Rational Rose.

La Interfaz de Rational Rose está formada por los siguientes elementos principales:

- Browser ó Navegador: Permite navegar rápidamente a través de las distintas vistas del modelo
- Ventana de Documentación: Permite manejar los documentos seleccionados en cualquiera de los diagramas.
- Barra de Herramientas Standard: Permite acceder rápidamente a las acciones comunes a ejecutar para cada uno de los diagramas del modelo.
- Barra de Herramientas Diagrama: Muestra el conjunto de herramientas disponibles para el diagrama activo.
- Ventana de Diagrama: Permite desplegar y editar cualquiera de los diagramas UML.
- Ventana Registro ó Log: Registra todas las órdenes ejecutadas y los errores que se producen durante su ejecución.



- Barra de Estado: Muestra el programa de la carga del modelo, el estado de lectura/escritura del elemento seleccionado, y otros datos de utilidad.

2.15.2 Visual Paradigm.

Visual Paradigm para UML es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue.

Esta herramienta soporta hasta la fecha UML 2.1 completo, permite realizar ingeniería tanto directa como inversa. Además es una herramienta colaborativa, ya que soporta múltiples usuarios trabajando sobre el mismo proyecto, genera la documentación del proyecto automáticamente en varios formatos como Web o .pdf y permite control de versiones.(30)

2.16 ¿Por qué se utiliza Rational Rose como herramienta de modelado?

Se escogió esta herramienta de modelado porque constituye la herramienta líder en el mundo de modelación visual para el proceso de modelación del negocio, análisis de requerimientos y diseño de arquitectura de componentes. Cubre todo el ciclo de vida de un proyecto: concepción y formalización del modelo, construcción de los componentes, transición a los usuarios y certificación de las distintas fases.

Características:

- Mantiene la consistencia de los modelos del sistema software.
- Posee chequeo de la sintaxis UML.
- Genera documentación automáticamente.
- Genera código a partir de los modelos.
- Posee ingeniería inversa (crear modelo a partir código).(29)

2.17 Otras Herramientas.

2.17.1 Servidor Apache.

Apache es la plataforma de servidores Web de código fuente abierto más poderosa del mundo. Apache es un servidor altamente configurable, con diseño modular. Es muy sencillo ampliar sus funcionalidades. Este servidor trabaja con la mayoría de lenguajes de la Web de la actualidad, como



Perl, PHP, y otros lenguajes Script. Existen versiones tanto para sistemas operativos Unix (Linux, Solaris, Mac-OS), así como con Windows. (31)

Se desarrolla dentro del proyecto HTTP Server (http) de la Apache Software Foundation. Apache presenta entre otras características mensajes de error altamente configurables, bases de datos de autenticación y negociado de contenido.

Ventajas:

- Open source
- Multi-plataforma
- Extensible
- Popular (fácil conseguir ayuda/saporte)
- Gratuito

El servidor Apache es un software que esta estructurado en módulos. La configuración de cada módulo se hace mediante la configuración de las directivas que están contenidas dentro del módulo. Los módulos del Apache se pueden clasificar en tres categorías:

Módulos Base: Módulo con las funciones básicas del Apache.

Módulos Multiproceso: son los responsables de la unión con los puertos de la máquina, aceptando las peticiones y enviando a los hijos a atender a las peticiones.

Módulos Adicionales: Cualquier otro módulo que le añada una funcionalidad al servidor.

Las funcionalidades más elementales se encuentran en el módulo base, siendo necesario un módulo multiproceso para manejar las peticiones. Se han diseñado varios módulos multiproceso para cada uno de los sistemas operativos sobre los que se ejecuta el Apache, optimizando el rendimiento y rapidez del código. El resto de funcionalidades del servidor se consiguen por medio de módulos adicionales que se pueden cargar. Para añadir un conjunto de utilidades al servidor, simplemente hay que añadirle un módulo, de forma que no es necesario volver a instalar el software.(32)



2.17.2 Eclipse.

Eclipse es una plataforma de software de código abierto independiente, que es y ha sido muy utilizada para desarrollar entornos de desarrollo (IDE), pero del mismo modo se puede usar para otros tipos de aplicaciones cliente. Lo mejor de Eclipse es que tiene una gran comunidad de usuarios extendiendo constantemente las aplicaciones.(33)

Uno de los proyectos de IDE's con Eclipse es el PHP Development Tools que trabaja para proveer un IDE completamente funcional para PHP en la plataforma Eclipse. Este proyecto ha tenido una gran respuesta entre los desarrolladores de PHP.

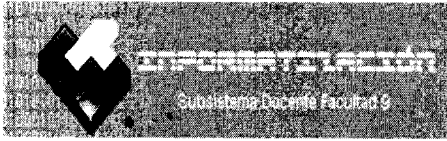
Entre las características en la versión actual (1.0) se encuentran:

- Editor sensible al contexto, el cual provee de resaltamiento de código, asistente de código y autocompletado de código.
- Integración con el modelo del proyecto Eclipse, que permite para inspeccionar el uso de las vistas del contorno del fichero y del proyecto, así como la nueva vista PHP Explorer.

2.18 Conclusiones.

En el presente capítulo se realizó un estudio detallado de los lenguajes de programación Web más utilizados en el mundo actualmente para el desarrollo de aplicaciones Web, incluyendo las principales características y ventajas de los mismos. Como resultado se seleccionó el lenguaje a utilizar para la validación de elementos, para el diseño y para la programación del sistema a desarrollar. Se realizó un análisis sobre los sistemas gestores de bases de datos más utilizados internacionalmente para almacenar y recuperar la información de una base de datos. Este análisis se centró en las principales características y ventajas de los mismos, concluyendo con la selección del SGBD a utilizar, ya que constituye el sistema de bases de datos de código abierto más avanzado del mundo.

Además se escogió el lenguaje de modelación, para visualizar, especificar, construir y documentar los artefactos del sistema a desarrollar y la metodología a utilizar para elaborar una estrategia de desarrollo de software orientada hacia la funcionalidad y la entrega exitosa del sistema. Se presenta además en este capítulo la herramienta de diseño y de modelado a utilizar.



CAPÍTULO 3

Presentación de la solución propuesta para el Subsistema Docente de la Facultad 9.

3.1 Introducción.

Durante el desarrollo de este capítulo se realizará un estudio acerca de cómo se desarrolla actualmente el negocio, con el objetivo de lograr un mejor entendimiento de los problemas existentes. Como resultado de este estudio se esperan encontrar los actores y trabajadores del negocio, así como los casos de uso y entidades del negocio. Se realizará el diagrama de caso de uso del negocio y la descripción textual de cada caso de uso del negocio. De igual forma se identificarán los actores del sistema, los requisitos funcionales y no funcionales, así como las funcionalidades que el sistema ofrecerá para brindar una solución factible al problema existente.

3.2 Reglas del Negocio.

Las reglas de negocio describen políticas que deben cumplirse o condiciones que deben satisfacerse, regulan algún aspecto del negocio.

Gestionar Afectación.

1. Sólo el Jefe de Departamento correspondiente es el encargado de autorizar una afectación a un profesor.

Gestionar Solicitud de Profesores.

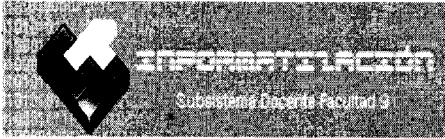
1. Sólo el Jefe de Departamento correspondiente es el encargado de autorizar un cambio de turno a un profesor.

Aplicar Actas Indisciplina.

1. El Decano tiene que aceptar y firmar el acta de advertencia del estudiante para que esta tenga validez.
2. El estudiante implicado tiene que firmar el acta de advertencia para que esta tenga validez.

Ratificar Matrícula.

1. El estudiante tiene que presentarse personalmente en la Secretaría Docente. El estudiante tiene que presentar el carné de identidad.



Solicitar Baja Docente.

1. El Decano tiene que firmar el modelo de baja.

Solicitar Baja Voluntaria.

1. El Decano tiene que aceptar y firmar el modelo de baja.
2. El estudiante tiene que firmar el modelo de baja.

Solicitar Traslado Externo.

1. El Decano tiene que aceptar la carta de solicitud de traslado.
2. La Secretaria General de la Universidad tiene que aprobar el traslado.

Licencia de Matrícula.

1. El Decano tiene que aceptar la Licencia de Matrícula.

Gestionar Resolución Decanal Estudiante.

1. El Decano tiene que aceptar la Licencia Especial.
2. El Decano de la facultad de la que desea trasladarse tiene que aceptar el traslado.
3. El Decano de la facultad a la que desea trasladarse tiene que aceptar el traslado.
4. La Secretaria General de la Universidad tiene que aceptar el traslado.

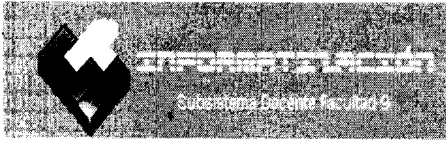
Gestionar Resolución Decanal Docente.

1. Los estudiantes Alumnos Ayudantes tienen que ratificar su ayudantía para estar en la resolución.
2. El Decano es el encargado de escoger los Alumnos Ayudantes que estarán en la resolución.
3. Los estudiantes Instructores No Graduados tienen que ratificar su ayudantía para estar en la resolución.
4. El Decano es el encargado de escoger los Instructores No Graduados que estarán en la resolución.
5. El estudiante implicado en la Resolución de Sanción tiene que firmar la resolución.

3.3 Modelo de Negocio.

El Modelo de Negocio se realiza con el objetivo de comprender la forma en que se está desarrollando el negocio, las características que posee así como las principales actividades que se desarrollan en el mismo. En él se determinan los actores del negocio y la relación que estos tienen con los procesos del negocio, así como los trabajadores del negocio y su relación con los objetos del mismo.

El **modelo de casos de uso del negocio** es un modelo que describe los procesos del negocio (casos de uso del negocio) y su interacción con elementos externos (actores).



Describe las funciones que el negocio pretende realizar y su objetivo básico es describir cómo el negocio es utilizado por sus clientes y socios. Implica la determinación de los actores y casos de uso del negocio. Con esta actividad se pretende: Identificar los procesos en el negocio, definir las fronteras del negocio que van a modelarse, Definir quién y qué interactuarán con el negocio y crear diagramas del modelo de casos de uso del negocio.(34)

El modelo del negocio describe el negocio en términos de casos de usos del negocio, que corresponde a lo que generalmente se le llama procesos.(34)

3.3.1 Actores del Negocio.

Un actor del negocio es cualquier individuo, grupo, entidad, organización, máquina o sistema de información externos; con los que el negocio interactúa. Lo que se modela como actor es el rol que se juega cuando se interactúa con el negocio para beneficiarse de sus resultados.(34)

Actores del Negocio	Justificación
Decano	Máxima dirección de la facultad, inicia la mayoría de las actividades que se realizan en el negocio, este solicita el control de los estudiantes ya sean de CDI o Cadetes, solicita Bajas Docentes e inicia la creación de Resoluciones Decanal Docentes.
Profesor	Es quien inicia la gestión de las afectaciones, la gestión de las solicitudes, así como aplica Actas por Indisciplina a los estudiantes.
Estudiante	Inicia la creación de Resoluciones Decanal de Estudiantes, solicita ya sea baja voluntaria o traslado externo y ratifica matrícula al inicio de cada curso.

Tabla 1: Descripción de los Actores del Negocio

3.3.2 Trabajadores del Negocio.

Representa a personas, o sistemas (software) dentro del negocio que son las que realizan las actividades que están comprendidas dentro de un caso de uso.



CAPÍTULO 3

Los trabajadores están dentro de la frontera del negocio, son los que en un futuro se convertirán en usuarios del sistema que se quiere construir.

Trabajadores del Negocio	Justificación
Jefe de Departamento	Máxima dirección del departamento y es el encargado de informar toda la información sobre la gestión de las afectaciones de sus respectivos profesores, así como aprobar o no los cambio de turnos solicitados por estos.
Planificadora Docente	Almacena las afectaciones de los profesores, realiza los cambios de turnos de estos y facilita locales solicitados por los mismos.
Secretaria Docente	Es la responsable de controlar y registrar toda la información docente relacionada con la facultad.
Técnico en Trabajo Docente Investigativo de la Educación Superior (TECADIES)	Es la encargada de realizar la ratificación de matrícula de cada estudiante al inicio de cada curso y de apoyar a la Secretaria Docente en las actividades que esta realiza.
Decano	Máximo responsable de la facultad. Es el que aprueba las Actas por Indisciplinas, los Traslados Externos, las Bajas Voluntarias, y las RD Estudiantes que incluyen: Licencias Especiales, Traslados Internos y las Licencias de Matrícula
Estudiante	Es responsable de firmar el Acta por Indisciplinas y la RD Docente: Sanción.
Secretaria General	Es la que acepta los Traslados tanto Externo como Internos.

Tabla 2: Descripción de los Trabajadores del Negocio.

3.3.3 Procesos de Negocio.

Un proceso de negocio es un grupo de tareas relacionadas lógicamente que se llevan a cabo en una determinada secuencia y manera y que emplean los recursos de la organización para dar resultados en apoyo a sus objetivos.(34)



Los Procesos de Negocio identificados fueron:

1. En el Vicedecanato de Formación referente a la Planificación Docente:
 - Gestionar Afectación.
 - Gestionar Solicitud de Profesor.
2. En la Secretaría Docente:
 - Aplicar Actas Indisciplina.
 - Ratificar Matrícula.
 - Solicitar Baja Docente.
 - Solicitar Baja Voluntaria.
 - Controlar Estudiante.
 - Solicitar Traslado Externo.
 - Solicitar Licencias de Matrícula.
 - Gestionar Resolución Decanal de Estudiante.
 - Gestionar Resolución Decanal Docente.

3.3.4 Diagrama de Casos de Uso del Negocio.

Un diagrama de casos de uso del negocio representa gráficamente a los procesos del negocio y su interacción con los actores del negocio.

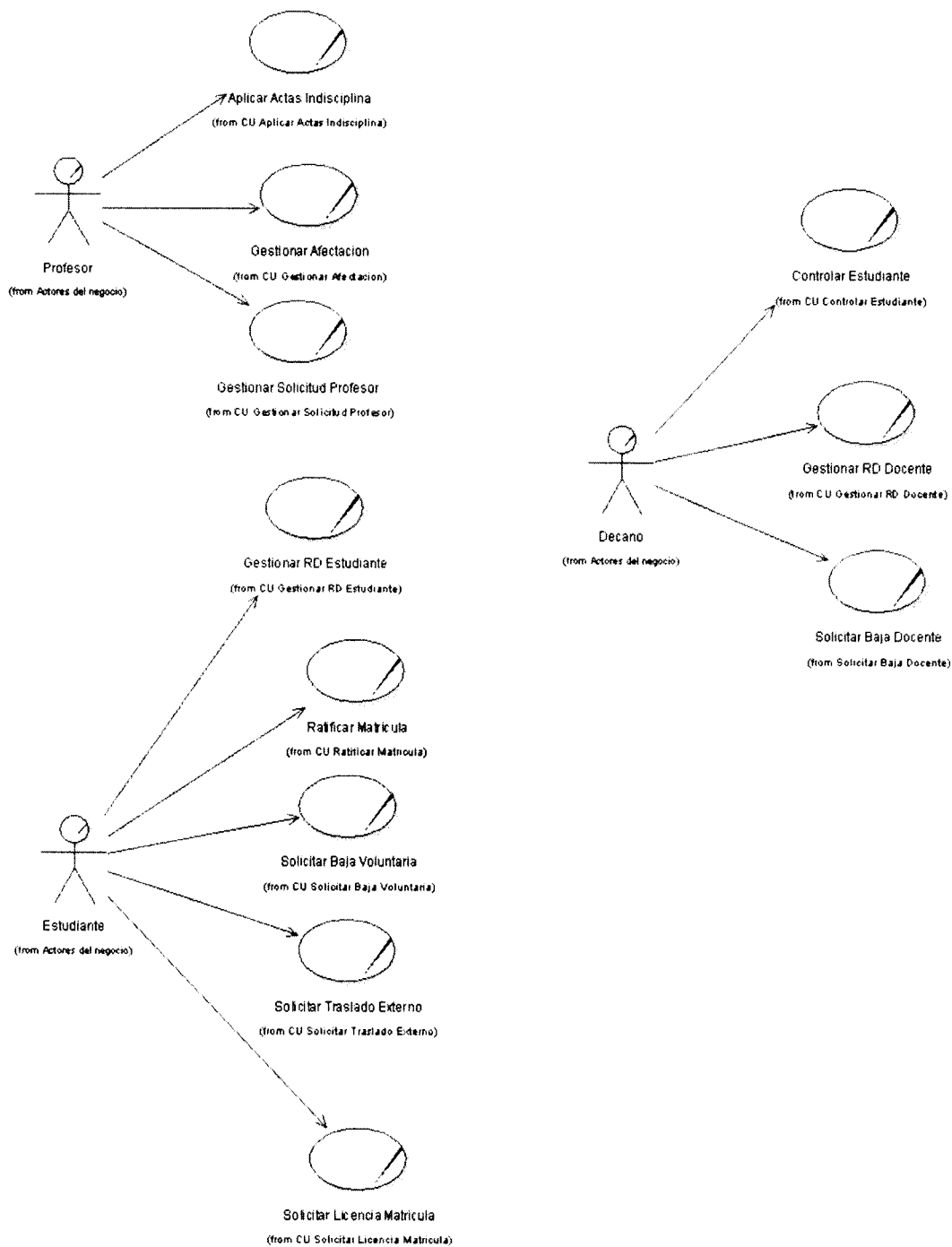


Figura 6: Diagrama de Casos de Uso del Negocio.



3.3.5 Descripción Textual de los Casos de Uso de Negocio.

Caso de Uso del Negocio: Gestionar Afectación	
Actores	Profesor
Propósito	Tiene como propósito que el profesor informe el comienzo de una afectación o cancele una afectación que tenía a sus respectivos Jefes de Departamentos y a partir de estas, se confeccionará el horario docente.
Resumen: El caso de uso se inicia cuando un profesor informa el comienzo de una afectación o el fin de una afectación, a su Jefe de Departamento, el cual añade o elimina esta afectación del listado de afectaciones que envía a la Planificadora Docente para la confección del horario docente	
Curso Normal de los Eventos	
Sección Informar Afectación	
Acción del Actor	Respuesta del Negocio
1- Informa una afectación que posee.	2- Si el Jefe de Departamento aprueba la afectación la adiciona a un listado con las afectaciones de todos los profesores de su departamento y se lo envía a la Planificadora Docente.
	3- La Planificadora Docente elabora el horario docente de acuerdo a la información recibida.
Sección Cancelar Afectación	
Acción del Actor	Respuesta del Negocio
1- Informa a su Jefe de Departamento que la afectación ha concluido.	2- El Jefe de Departamento actualiza el listado eliminando dicha afectación y se lo entrega a la Planificadora Docente.
	3- La Planificadora Docente elabora el horario docente de acuerdo a la información recibida.
Curso Alternativo de los Eventos	
Sección Informar Afectación	
	2.1- Si el Jefe de Departamento no aprueba la afectación se lo informa al profesor.



CAPÍTULO 3

Prioridad	
Mejoras	
Otras Secciones	

Tabla 3: Descripción Textual del CU Gestionar Afectación.

Caso de Uso del Negocio: Solicitar Licencia Matrícula	
Actores	Estudiante
Propósito	Solicitar una Licencia de Matrícula.
Resumen: El caso de uso se inicia cuando el estudiante le solicita al decano una Licencia de Matrícula, si este aprueba la solicitud, la Secretaria Docente crea y archiva el modelo de la Licencia de Matrícula.	
Curso Normal de los Eventos	
Acción del actor	Respuesta del Negocio
1- Solicita Licencia de Matrícula.	2- Si el decano aprueba la solicitud, solicita la creación del modelo de la Licencia de Matrícula a la Secretaria Docente.
	3- La Secretaria Docente crea y archiva el modelo.
	4- La Secretaria Docente actualiza el listado (dicho listado se encuentra en una Hoja de Cálculo de Microsoft Excel) que los siguientes datos de los estudiantes que constituyen Licencias de Matrícula: carné de identidad, sexo, nombre, primer apellido, segundo apellido, fecha, facultad, año, grupo, provincia, municipio, causa (Misiones internacionalistas o Movilización militar, Enfermedad o Accidente, Maternidad o Embarazo, Participación en actividades deportivas como atletas de alto rendimiento, Participación en actividades de grupos nacionales de cultura por ser miembros de los mismos, Necesidad impostergable de la producción o los servicios, Desaprobar una



CAPÍTULO 3

	<p>asignatura en la convocatoria extraordinaria de fin de curso si, estando matriculado en el curso regular diurno, cumplen con las disposiciones que al respecto se dictan en el Capítulo V del Reglamento, Problemas Familiares), vía de ingreso(Diferido, Escalafón, MINFAR, MININT, Orden 18, Orden 18 Experimental, Deporte), procedencia (IPVCE, IPUEC, Técnico Medio en Informática, ESPA), observaciones, organización a la que pertenece, teléfono y si es Licencia especial o no.</p>
Curso Alternativo de los Eventos	
	2.1- Si el decano no aprueba la solicitud se lo informa al estudiante.
Prioridad	
Mejoras	
Otras Secciones	

Tabla 4: Descripción Textual del CU Solicitar Licencia de Matrícula.

Caso de Uso del Negocio: Solicitar Traslados Externos	
Actores	Estudiante
Propósito	Tiene como propósito solicitar un traslado para otra universidad.
<p>Resumen: El caso de uso se inicia cuando un estudiante le presenta al decano una carta de solicitud de traslado para otra universidad y después que este la aprueba le envía a la Secretaria Docente de la facultad la carta presentada por el estudiante con su aprobación y esta a su vez se la reenvía a la Secretaria General de la universidad quien esta presente en la comisión encargada de dar la aprobación final.</p>	
Curso Normal de los Eventos	
Acción del actor	Respuesta del Negocio
1- Presenta carta de solicitud de traslado.	2- Si el Decano aprueba dicha carta, se la envía a la Secretaria Docente.
	3- La Secretaria Docente de la facultad se la envía a la Secretaria General de la Universidad quien esta presente en la comisión encargada de



CAPÍTULO 3

	dar la aprobación final.
	4- La Secretaria General de la Universidad envía a la Secretaria Docente la relación con todos los traslados aceptados.
	5- Si el estudiante se encuentra en dicha relación la Secretaria Docente prepara el expediente e información necesaria para el traslado.
	6- La Secretaria Docente actualiza el listado con los siguientes datos del estudiante al que se le aprobó su solicitud de traslado (dicho listado se encuentra en una Hoja de Cálculo de Microsoft Excel): carné de identidad, nombre, primer apellido, segundo apellido, grupo, fecha, motivo, dirección particular, provincia, municipio, observaciones, procedencia y vía de ingreso.
Curso Alternativo de los Eventos	
	2.1- Si el Decano no aprueba dicha carta se lo informa al estudiante.
	5.1- Si el estudiante no se encuentra en el listado la Secretaria Docente le informa que su solicitud no fue aceptada.
Prioridad	
Mejoras	
Otras Secciones	

Tabla 5: Descripción Textual del CU Solicitar Traslados Externos.

3.3.6 Diagramas de Actividades de los Casos de Uso del Negocio.

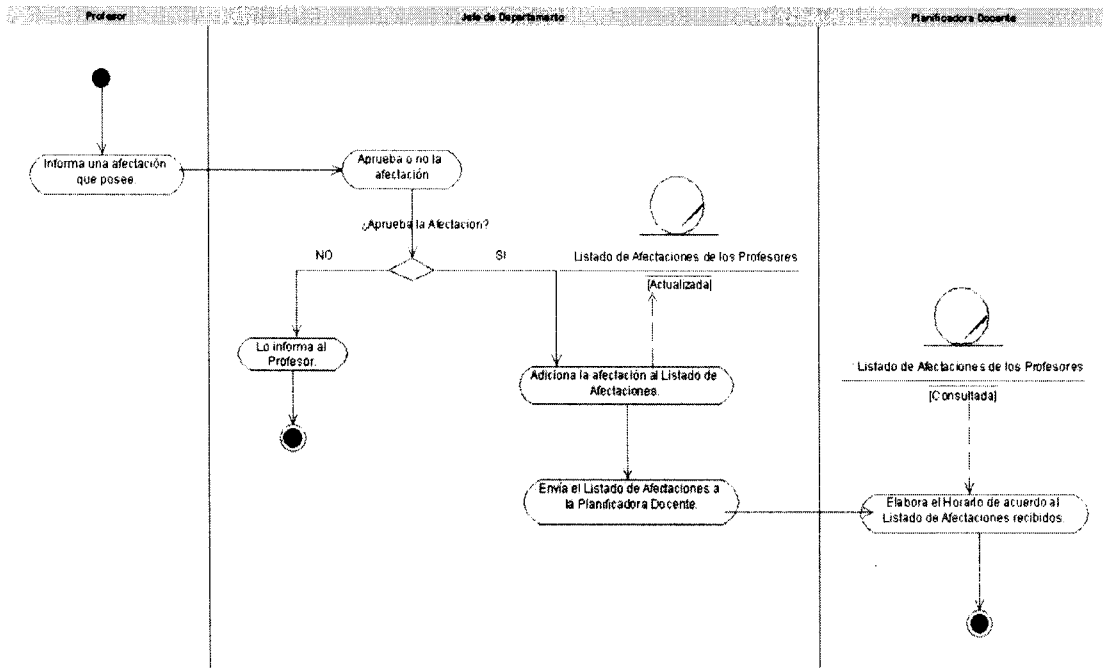


Figura 7: Diagrama de Actividades del CU Gestionar Afectación: Sección Informar Afectación.

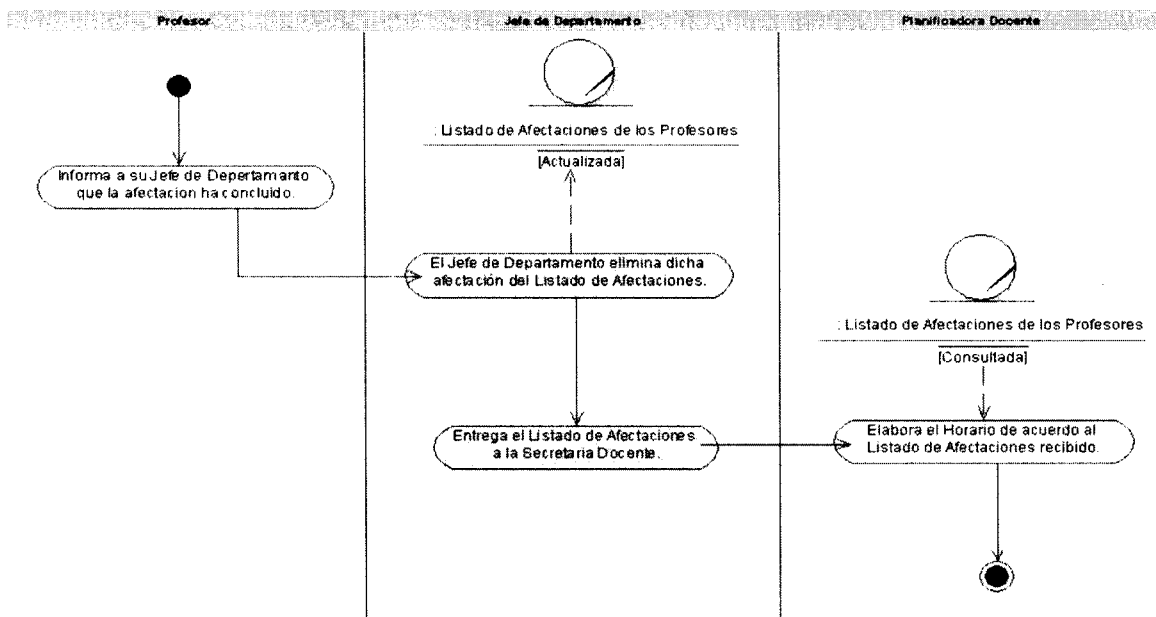


Figura 8: Diagrama de Actividades del CU Gestionar Afectación: Sección Eliminar Afectación.

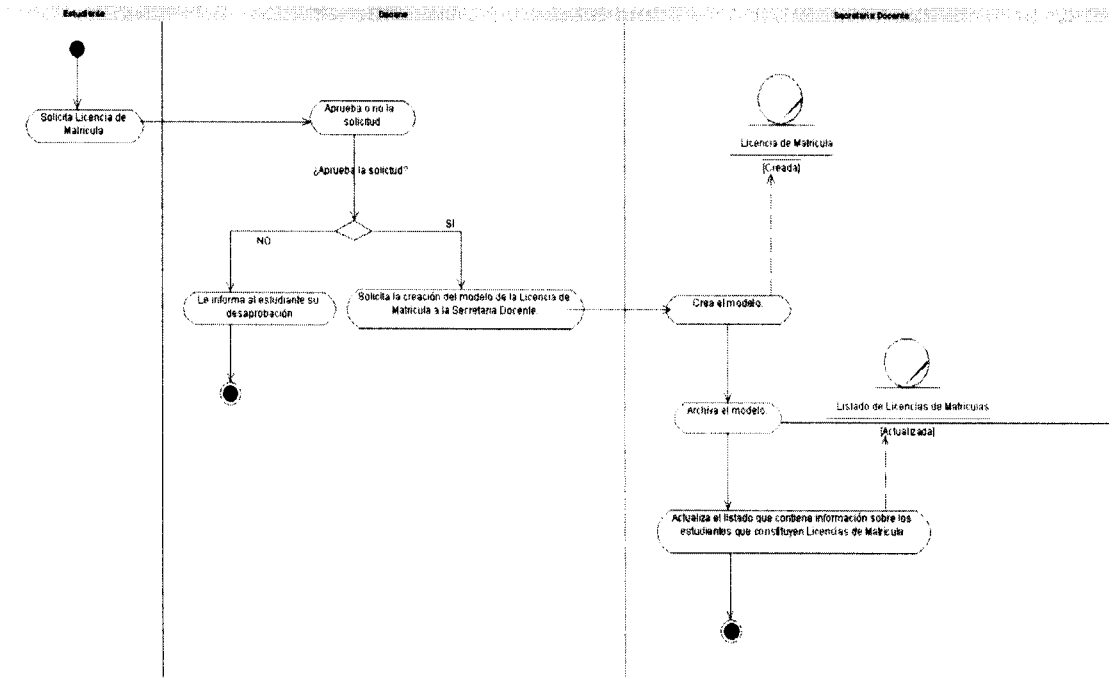


Figura 9: Diagrama de Actividades del CU Solicitar Licencia de Matricula.

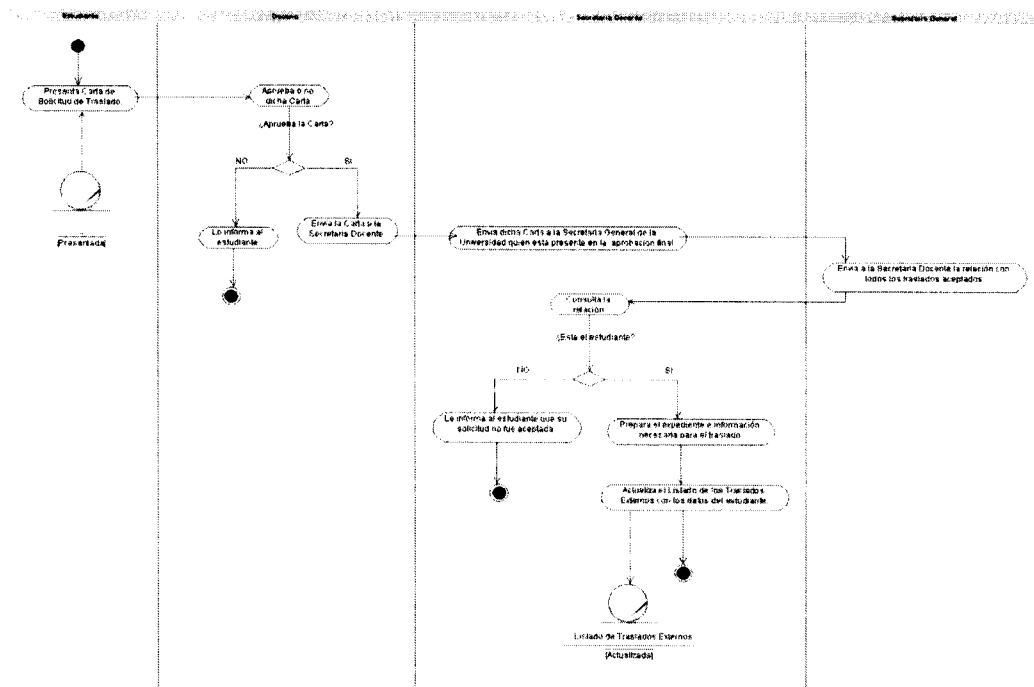


Figura 10: Diagrama de Actividades del CU Solicitar Traslado Externo.

3.3.7 Modelo de Objetos del Negocio.

El Modelo de Objetos del Negocio describe como colaboran los trabajadores y las entidades del negocio dentro del flujo de trabajo del proceso del negocio.

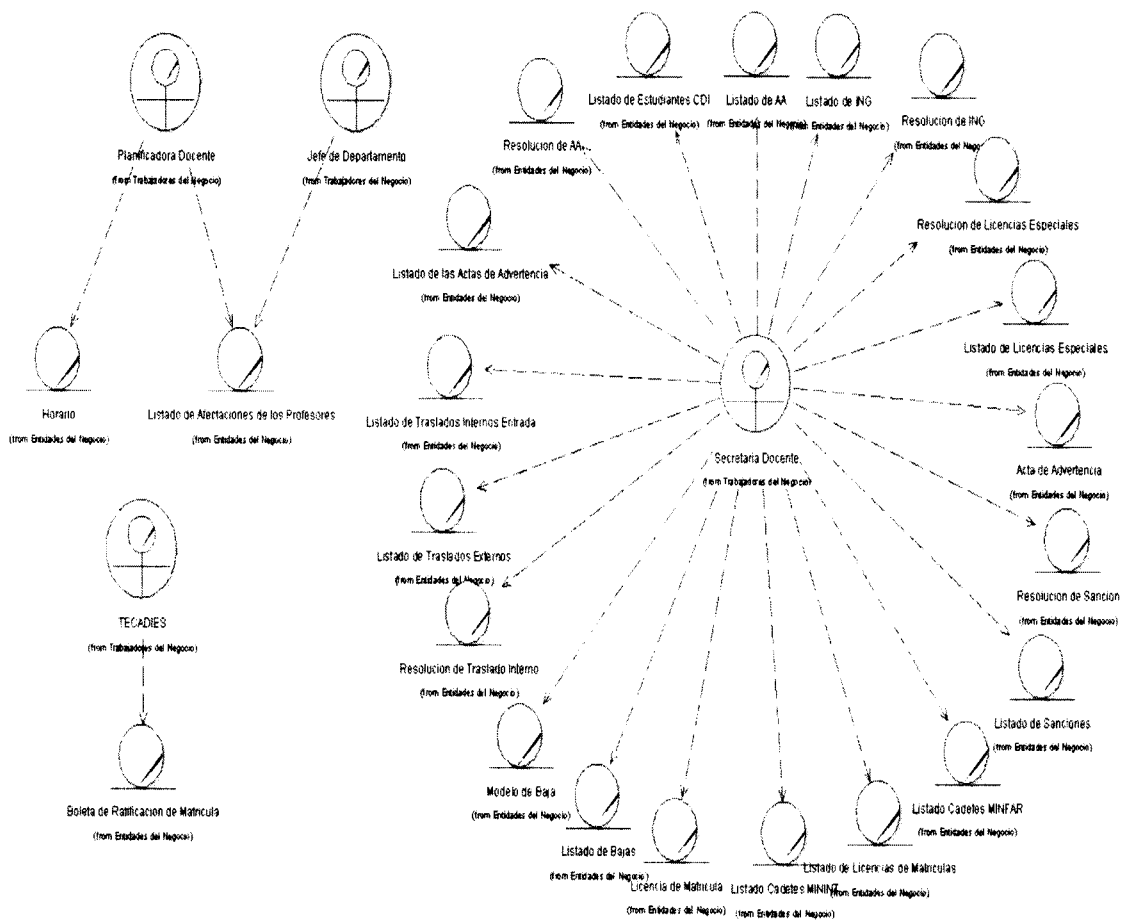


Figura 11: Modelo de Objetos del Negocio.

3.4 Modelo del Sistema Propuesto.

Un modelo de casos de uso del sistema es un modelo del sistema que contiene actores, casos de uso y sus relaciones.(24)



El modelo de casos de uso permite llegar a un acuerdo con los clientes sobre los requisitos funcionales y no funcionales que debe cumplir el sistema, es importante ya constituye la entrada fundamental para el análisis, el diseño y las pruebas.

3.4.1 Actores del Sistema.

Los actores representan terceros fuera del sistema que colaboran con el sistema. (24)

Los actores del sistema suelen corresponderse con los trabajadores del negocio y si algún actor del negocio va a interactuar con el sistema pasa también a ser actor del sistema.

Actores del Sistema	Justificación
Usuario	Generaliza a todos los usuarios que interactúan con el sistema.
Profesor	Realiza las operaciones de gestionar solicitud de locales.
Jefe de Departamento	Realiza las operaciones de gestionar solicitud de cambio de turno, de gestionar afectación y como él es un profesor puede gestionar solicitud de locales.
Planificadora Docente	Realiza todas las operaciones relacionadas con la información de la planificación docente.
TECADIES	Realiza todas las operaciones relacionadas con la ratificación de matrícula.
Secretaria Docente	Es la encargada de toda la gestión de las Resoluciones Decanal, de las bajas, del control de estudiantes, de los traslados externos, de las actas por indisciplina y de las Licencias de Matrícula.

Tabla 6: Actores del Sistema.

3.4.2 Requisitos Funcionales.

Los requerimientos funcionales son capacidades o condiciones que el sistema debe cumplir.

1- Gestionar Afectación.

R1.1: Insertar Afectación.



R1.2: Modificar Afectación.

R1.3: Eliminar Afectación.

2- Gestionar Solicitud de Cambio de Turno.

R2.1: Insertar Solicitud de Cambio Turno.

R2.2: Modificar Solicitud de Cambio Turno.

R2.3: Eliminar Solicitud de Cambio Turno.

3- Gestionar Solicitud de Locales.

R3.1: Insertar Solicitud de Locales.

R3.2: Modificar Solicitud de Locales.

R3.3: Eliminar Solicitud de Locales.

4- Gestionar Actas por Indisciplinas.

R4.1: Insertar Actas por Indisciplina.

R4.2: Modificar Actas por Indisciplina.

R4.3: Eliminar Actas por Indisciplina.

5- Gestionar Ratificación de Matrícula.

R5.1: Insertar Ratificación de Matrícula.

R5.2: Modificar Ratificación de Matrícula.

R5.3: Eliminar Ratificación de Matrícula.

6- Gestionar Baja.

R6.1: Insertar Baja.

R6.2: Modificar Baja.

R6.3: Eliminar Baja.



7- Gestionar Control de Estudiantes.

R7.1: Insertar Estudiante.

- R7.1.1: Insertar Estudiante Cadete.
- R7.1.2: Insertar Estudiante CDI.

R7.2: Modificar Estudiante

- R7.2.1: Modificar Cadete.
- R7.2.2: Modificar Estudiante CDI.

R7.3: Eliminar Estudiante

- R7.3.1: Eliminar Cadete.
- R7.3.2: Eliminar Estudiante CDI.

8- Gestionar Traslado Externo.

R8.1: Insertar Traslado Externo.

R8.2: Modificar Traslado Externo.

R8.3: Eliminar Traslado Externo.

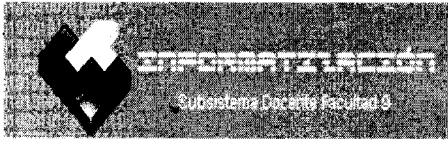
9- Gestionar Resolución Decanal.

R9.1 Insertar Resolución Decanal

- R9.1.1: Insertar Alumno Ayudante.
- R9.1.2: Insertar Instructores No Graduados.
- R9.1.3: Insertar Sanción.
- R9.1.4: Insertar Licencia Especial.
- R9.1.5: Insertar Traslado Interno.

R9.2 Modificar Resolución Decanal

- R9.2.1: Modificar Alumno Ayudante.
- R9.2.2: Modificar Instructores No Graduados.
- R9.2.3: Modificar Sanción.
- R9.2.4: Modificar Traslado Interno.



- R9.2.5: Modificar Licencia Especial.

R9.3 Eliminar Resolución Decanal

- R9.3.1: Eliminar Alumno Ayudante.
- R9.3.2: Eliminar Instructores No Graduados.
- R9.3.3: Eliminar Sanción.
- R9.3.4: Eliminar Licencia Especial.
- R9.3.5: Eliminar Traslado Interno.

10- Gestionar Licencia Matrícula.

- R10.1: Insertar Licencia de Matrícula.
- R10.2: Modificar Licencia de Matrícula.
- R 10.3: Eliminar Licencia de Matrícula.

11- Mostrar Información Estudiante.

R11.1: Mostrar si un Estudiante tiene Acta por Indisciplina o No y si tiene cuales.

R11.2: Mostrar si un Estudiante es Baja o No.

R11.3: Mostrar si un Estudiante es Cadete o No y si lo es, cual es su Vía de Ingreso.

R11.4: Mostrar si un Estudiante es Traslado Externo o No.

R11.5: Mostrar si un Estudiante es Licencia Especial o No y si lo es, que Asignatura Suspendió y que Asignaturas va ha Adelantar.

R11.6: Mostrar si un Estudiante ha sido Sancionado o No y si lo ha sido que Tipo de Sanción le aplicaron y porque Causa.

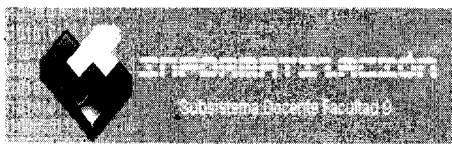
R11.7: Mostrar si un estudiante sancionado es cadete o no

R11.8: Mostrar si un Estudiante es Licencia de Matrícula o No y si lo es cual fue la Causa por la que solicitó la Licencia.

12- Emitir Reportes.

R12.1: Reporte de Estudiantes con Actas por Indisciplina.

R12.2: Reporte de Estudiantes que han Ratificado Matrícula por Grupo.



- R12.3: Reporte de Estudiantes que han sido Bajas.
- R12.4: Reporte de Estudiantes Cadetes que son Baja.
- R12.5: Reporte de Bajas por Año.
- R12.6: Reporte de Bajas por Año y por Causa.
- R12.7: Reporte de Bajas por Grupo.
- R12.8: Reporte de Bajas por Grupo y por Causa.
- R12.9: Reporte de Cadetes por Año.
- R12.10: Reporte de Cadetes por Año y Vía de Ingreso.
- R12.11: Reporte de Estudiantes CDI por Grupo.
- R12.12: Reporte de Traslado Externo.
- R12.13: Reporte de los Estudiantes Cadetes que son Traslado Externo.
- R12.14: Reporte de Alumnos Ayudantes.
- R12.15: Reporte de Instructores No Graduados.
- R12.16: Reporte de Sanciones.
- R12.17: Reporte de Licencias de Matrícula.
- R12.18: Reporte de Licencias Especiales.
- R12.19: Reporte de Traslados Internos que entran a la facultad.
- R12.20: Reporte de Traslados Internos que salen de la facultad.
- R12.21: Reporte de Alumnos Ayudantes por Grupo.
- R12.22: Reporte de Instructores No Graduados por Grupo.
- R12.23: Reporte de Licencias de Matrícula por Semestre y por Año.
- R12.24: Reporte de Licencias Especiales por Semestre y por Año.



13- Mostrar Listados Planificación Docente.

R13.1: Mostrar Listado de Afectaciones.

R13.2: Mostrar Listado de Solicitudes de Cambio Turno.

R13.3: Mostrar Listado de Solicitudes de Locales.

R13.4: Mostrar Listado de Afectaciones por Departamento.

R13.5: Mostrar Listado de Solicitudes de Cambio por Departamento.

R13.6: Mostrar Listado de Solicitudes de Locales por Departamento.

3.4.3 Requisitos No Funcionales.

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener, estas propiedades son las características que hacen al producto atractivo, usable, rápido o confiable.

Requerimientos de Software

1. El sistema será implementado usando una tecnología Web. (Base de Dato en PostgreSQL y programado en PHP).
2. En el lado del Cliente debe existir un navegador compatible con Internet Explorer 4 o superior y que soporte funciones JavaScript.
3. En el lado del servidor debe estar instalado: PHP 5.0 o superior, y gestor de base de datos PostgreSQL 8.2 o superior.
4. Framework Symfony.
5. Servidor Apache 2.0 o superior.

Requerimientos de Hardware

El servidor de aplicaciones y el servidor de base de datos deberá contar con:

1. Procesador Pentium IV o superior.
2. 512 Mb de memoria RAM o superior.

Restricciones en el diseño y la implementación

1. El sistema será implementado en el lenguaje de programación Web PHP.

2. El gestor de base de datos a utilizar será Postgre SQL.
3. El análisis y el diseño del sistema llevarán la metodología RUP, lenguaje de modelación UML y la herramienta de modelado Rational Rose.

Requerimientos de apariencia o interfaz externa

1. El sistema interactuará con el usuario mediante una interfaz sencilla, y en concordancia al estilo de las aplicaciones UCI.

Requerimientos de Seguridad

1. Identificar al usuario antes de que pueda realizar cualquier acción.
2. Garantizar que las funcionalidades del sistema se muestren de acuerdo al nivel de usuario que esté activo.

Requisitos de Disponibilidad

1. El sistema debe permitirle al usuario conectarse desde cualquier PC de la red.

Requerimientos de Usabilidad

1. El sistema podrá ser usado por cualquier persona con acceso al mismo y que posea un conocimiento básico de informática y de trabajo con aplicaciones Web.
2. El sistema podrá ser usado por los Profesores, los Jefes de Departamento, la Planificadora Docente, la Secretaria Docente, la TECADIES y el Vicedecano de Formación de la Facultad 9.

Requerimientos de Soporte

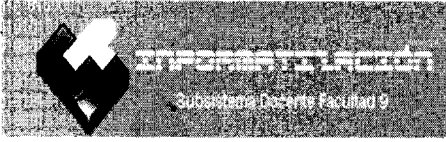
1. El sistema será de fácil instalación y de fácil mantenimiento

Requerimientos Legales

1. Este sistema de Gestión de la Información Docente en el Vicedecanato de Formación, la Planificación Docente y la Secretaría Docente de la Facultad 9 pertenece a la Universidad de las Ciencias Informáticas, específicamente al Vicedecanato de Formación y la Secretaría Docente de la Facultad 9.

Requerimientos de Confiabilidad

1. Garantía de un tratamiento adecuado de las excepciones y validación de las entradas del usuario.



3.4.4 Diagrama de Casos de Uso del Sistema.

Un diagrama de casos de uso del sistema representa gráficamente a los procesos y su interacción con los actores.

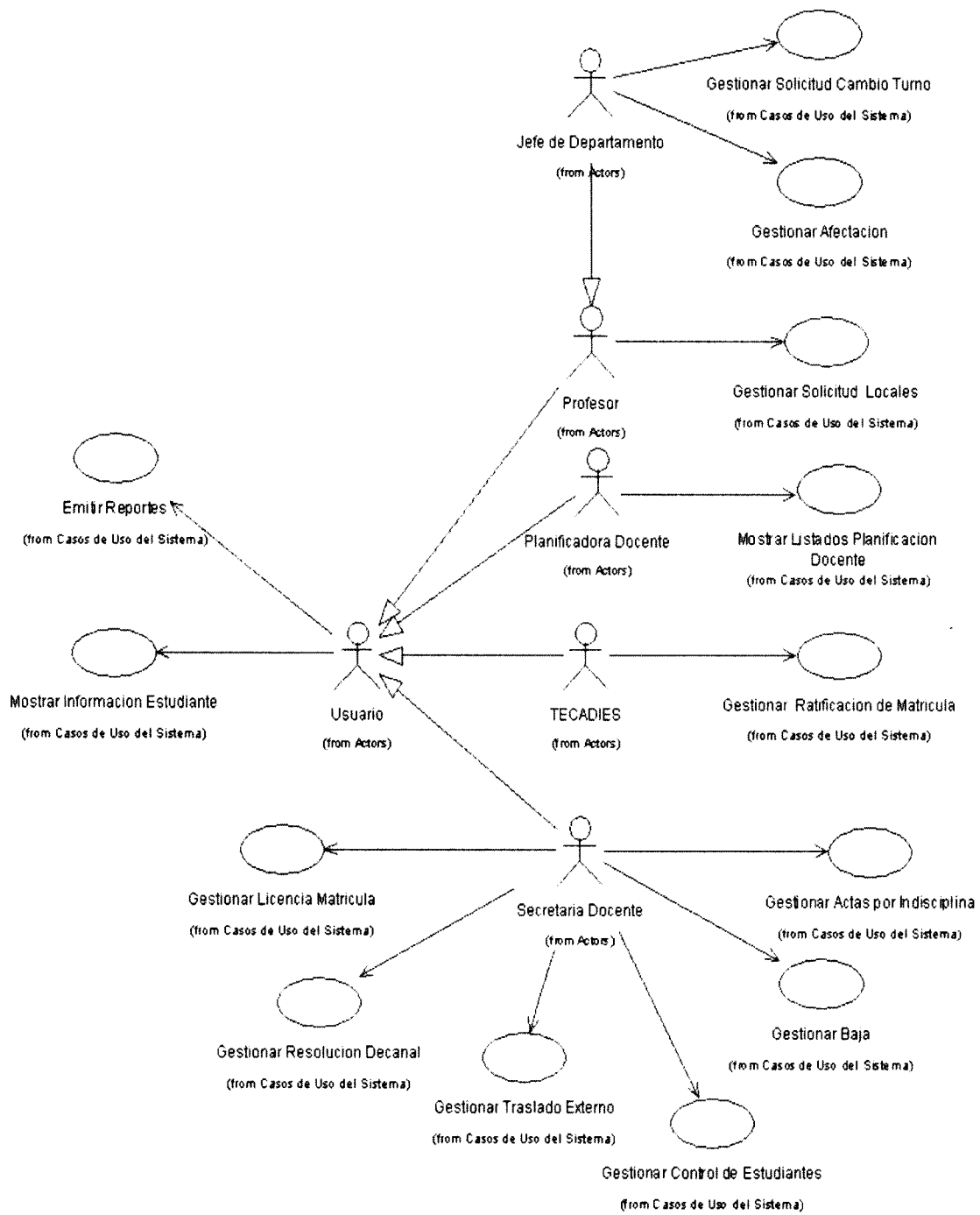


Figura 12: Diagrama de Casos de Uso del Sistema.





3.4.5 Descripción de los Casos de Uso del Sistema.

Nombre del Caso de Uso	Gestionar Afectación.
Actor	Jefe de Departamento (Inicia).
Propósito	Garantizar la gestión de una afectación presentada por un profesor, permitiendo la inserción, modificación y eliminación de la misma en la aplicación.
Precondiciones	El Jefe de Departamento debe estar autenticado.
Resumen	El caso de uso se inicia cuando el Jefe de Departamento solicita realizar la gestión de una afectación. Puede insertar, modificar o eliminar una afectación.
Referencias	R1
Curso Normal de los Eventos	
Acciones del Actor	Respuesta del Sistema
1- El Jefe de Departamento accede a la Interfaz Docencia.	1.1- El sistema muestra la Interfaz con una serie de acciones a realizar.
2- El Jefe de Departamento accede a la Interfaz de Vicedecanato de Formación.	2.1- El sistema muestra la Interfaz con una serie de acciones a realizar.
3- El Jefe de Departamento accede a la interfaz de Gestionar Afectación.	3.1- El sistema muestra la Interfaz, con una serie de acciones a realizar.
4- El Jefe de Departamento elige la acción a realizar.	4.1- Si elige Insertar Afectación, ir a Sección Insertar Afectación.
	4.2- Si elige Modificar Afectación, ir a Sección Modificar Afectación.
	4.3- Si elige Eliminar Afectación, ir a Sección Eliminar Afectación.



CAPÍTULO 3

 INFORMATIZACIÓN Plataforma de Informatización Facultad 9	
Opciones <i>Insertar Afectación</i> <i>Modificar Afectación</i> <i>Eliminar Afectación</i>	Estimado Usuario Usted puede realizar la operación que desee.
Sección: "Insertar Afectación"	
	1- El sistema muestra en la interfaz un formulario con los campos generales que se deben introducir.
2- El Jefe de Departamento introduce los datos de la afectación que desea insertar.	2.1- El sistema valida los datos introducidos. En caso de algún dato inconsistente, ver CA1.
	2.2- El sistema Inserta la nueva afectación y muestra la Interfaz de Insertar Afectación.
Curso Alternativo de los Eventos	
CA1: Si introduce algún dato inconsistente, emitir mensaje de error.	
Poscondiciones	Se ha agregado una nueva afectación al sistema.
 INFORMATIZACIÓN Plataforma de Informatización Facultad 9	
Opciones <i>Modificar Afectación</i> <i>Eliminar Afectación</i>	Insertar Afectación Solapin Nombre de la Afectación Día de la Afectación Hora de la Afectación <input type="button" value="Insertar"/>
Sección: "Modificar Afectación"	
	1- El sistema muestra una interfaz con un listado con las afectaciones que se

	encuentran en la base de datos dando la posibilidad de seleccionar la afectación que se desea modificar.
2- El Jefe de Departamento selecciona la afectación que desea modificar.	2.1- El sistema busca la afectación seleccionada y muestra una interfaz con sus datos, dando la posibilidad de que el Jefe de Departamento realice las modificaciones que desee.
3- El Jefe de Departamento introduce las modificaciones necesarias.	3.1- El sistema verifica que no hayan introducido datos inconsistentes. En caso de que introduzca algún dato inconsistente, ver CA1.
	3.2- El sistema muestra la Interfaz de Modificar Afectación.
Curso Alternativo de los Eventos	
CA1: Si el Jefe de Departamento introduce algún dato inconsistente, emite error.	
Poscondiciones	Se ha modificado una afectación en el sistema.



ID	Solapin_Profesor	Afectacion	Dia	hora	Modificar
1	55349	Maestria	Lunes	8:00am a 12:00m	Modificar



Modificar Afectacion

ID	Solapin_Profesor	Afectacion	Dia	Hora	Modificar
1	55349	Maestria	Lunes	8:00am a 12:00m	Modificar

Sección: "Eliminar Afectación"

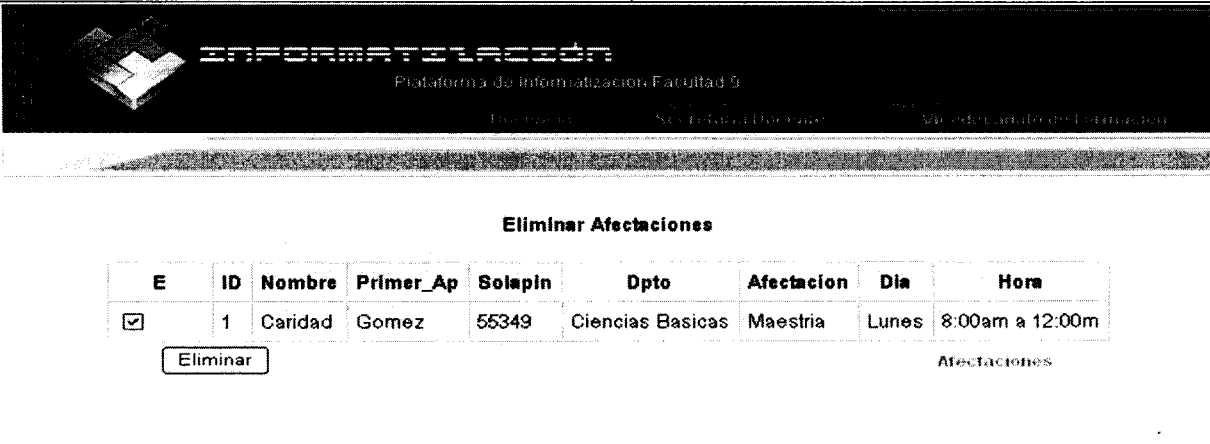
	1- El sistema muestra una interfaz con un listado con las afectaciones que se encuentran en la base de datos dando la posibilidad de seleccionar la afectación que se desea eliminar.
2- El Jefe de Departamento selecciona la afectación que desea eliminar.	2.1- El sistema busca la afectación seleccionada, la elimina y muestra la Interfaz Eliminar Afectación.
Curso Alternativo de los Eventos	
Poscondiciones	Se eliminó una afectación en el sistema.
	

Tabla 7: Descripción Textual del CUS Gestionar Afectación.

Nombre del Caso de Uso	Mostrar Listados Planificación Docente
Actor	Planificadora Docente (Inicia)
Propósito	Mostrar todos los listados relacionados con las solicitudes realizadas por los profesores.
Precondiciones	La planificadora docente tiene que estar autenticada.
Resumen	El caso de uso se inicia cuando la Planificadora Docente solicita mostrar listados Planificación Docente, el sistema le muestra todos los listados relacionados con las solicitudes realizadas por los profesores y ella escoge el que desea consultar.
Referencias	R13
Curso Normal de los Eventos	
Acción del Actor	Respuesta del Sistema



CAPÍTULO 3

1- La Planificadora Docente accede a la Interfaz Docencia.	1.1- El sistema muestra la Interfaz con una serie de acciones a realizar.
2- La Planificadora Docente accede a la Interfaz del Vicedecanato de Formación.	2.1- El sistema muestra la Interfaz con una serie de acciones a realizar.
3- La Planificadora Docente accede a la interfaz de Mostrar Listados de Planificación Docente.	<p>3.1- El sistema le muestra en la interfaz todos los listados que la Planificadora Docente puede consultar:</p> <ul style="list-style-type: none"> - Mostrar Listado de Afectaciones. - Mostrar Listado de Solicitudes de Cambio Turno. - Mostrar Listado de Solicitudes de Locales. - Mostrar Listado de Afectaciones por Departamento. - Mostrar Listado de Solicitudes de Cambio por Departamento. - Mostrar Listado de Solicitudes de Locales por Departamento.

INFORMATIZACIÓN
Planificadora de Informatización Facultad 4

Inicio Inicio de Sesión Inicio de Sesión Inicio de Sesión

Opciones

- Afectaciones
- Solicitudes de Locales
- Solicitudes de Cambio Turno
- Afectaciones por Dpto
- Solicitudes de Locales por Departamento
- Solicitudes de Cambio Turno por Departamento

Estimado Usuario

Usted puede realizar la operacion que desea.



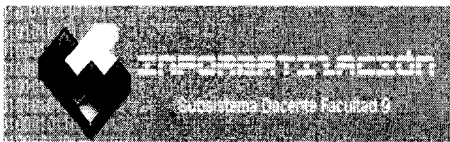
4- La Planificadora Docente escoge el listado a consultar.	4.1- El sistema muestra el listado solicitado.
Poscondiciones	Se muestra un listado solicitado por la Planificadora Docente.

Tabla 8: Descripción Textual del CUS Mostrar Listados Planificación Docente.

3.5 Conclusiones.

Con la realización de este capítulo se ha comprendido mejor el entorno donde coexistirá el sistema a realizar y las restricciones planteadas por los clientes que tendrá dicho sistema, para esto se realizó el modelo de negocio que incluyó el diagrama de casos de uso del negocio, en el cual se representaron los actores del negocio y su relación con los casos de uso del negocio así como el modelo de objetos en el cual se representaron los trabajadores del negocio con las entidades del negocio con las que estos se relacionan. Para cada caso de uso del negocio se realizó su respectiva descripción textual y su diagrama de actividades en el cual se representaron todas las actividades que se llevan a cabo en el proceso destacando aquellas que se van a automatizar.

Se identificaron las características que tendrá el sistema, derivadas estas de los requisitos planteados por los clientes tanto funcionales como no funcionales. Identificadas dichas características se realizó el modelo de sistema en el cual se representaron los actores y su relación con los casos de uso del sistema, se realizó una descripción textual de cada uno de los casos de uso del sistema detallando en ellas las características que va a tener cada proceso automatizado.



CAPÍTULO 4

Construcción de la solución propuesta para el Subsistema Docente de la Facultad 9.

4.1 Introducción

El presente capítulo se centrará en el Análisis y Diseño del sistema a desarrollar. En este flujo de trabajo se representaran las clases que interactúan en el sistema, para ello se modelará para cada Caso de Uso un Diagrama de Clases del Análisis, así como un Diagrama de Colaboración para cada sección que componen a estos Casos de Uso y un Diagrama de Clases del Diseño para dar cumplimiento a los requerimientos funcionales.

En este capítulo también se analizarán cuales son las clases persistentes creando el Diagrama de Clases Persistentes, a partir del cual se generará el Modelo de Datos, el cual refleja las tablas de la Base de Datos con sus relaciones. También se realizará la distribución física del sistema, creando el Diagrama de Despliegue y se construirán los Diagramas de Componentes para cada Caso de Uso.

4.2 Modelo de Análisis.

El Modelo de Análisis se describe utilizando el lenguaje de los desarrolladores, puede ser utilizado para razonar sobre los funcionamientos internos del sistema ya que constituye una vista interna del sistema. Está estructurado por clases que siempre encajan en tres estereotipos básicos: Clase Interfaz, Clase Control, Clase Entidad. Es utilizado fundamentalmente por los desarrolladores para comprender como debería darse forma al sistema, es decir, como debería ser diseñado e implementado.

Se decidió realizar el Modelo de Análisis porque proporciona las siguientes ventajas:

- Suaviza la transición al Diseño.
- Apoya el cambio a otra plataforma de programación. Sirve para tener una visión general de la propuesta del sistema.

4.2.1 Diagramas de Clases del Análisis.

Los Diagramas de Clases del Análisis constituyen diagramas de estructura estática en los que se representan los conceptos en un dominio del problema. Este tipo de diagramas representa los elementos del mundo real, no la implementación automatizada de los mismos.

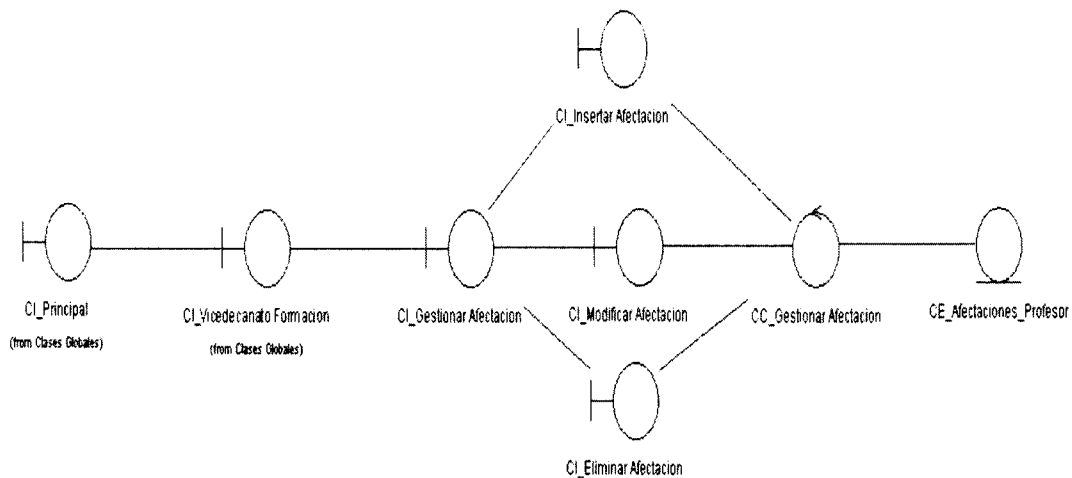


Figura 13: Diagramas de Clases del Análisis del CUS Gestionar Afectación.

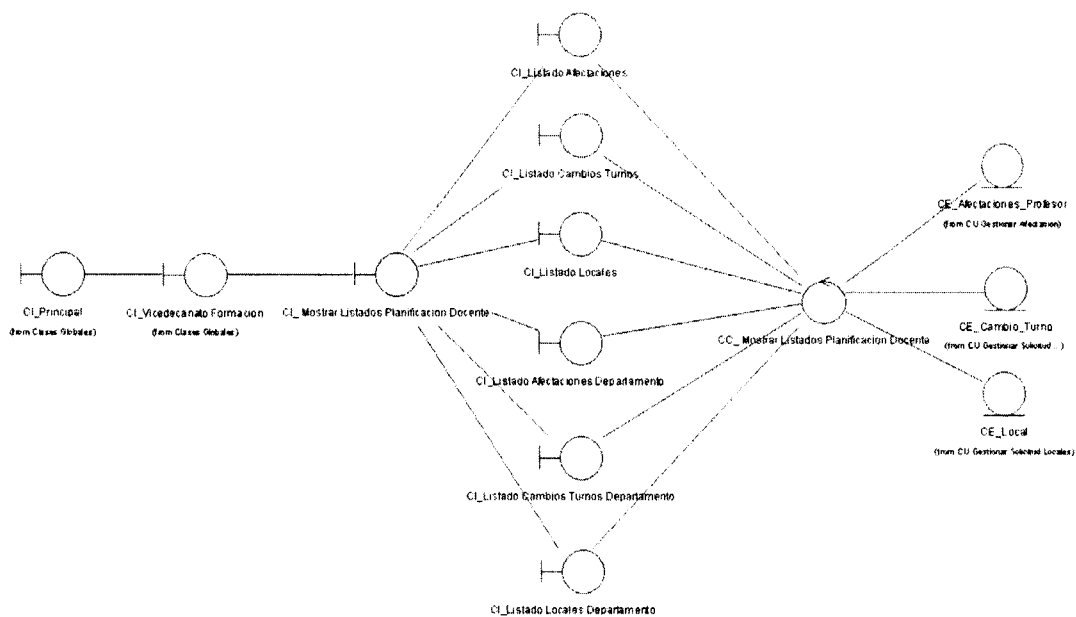


Figura 14: Diagramas de Clases del Análisis del CUS Mostrar Listados Planificación Docente.

4.2.2 Diagramas de Interacción: Colaboración.

Los Diagramas de Interacción muestran las interacciones entre objetos mediante transferencia de mensajes.

Se recomienda en el Análisis realizar Diagramas de Colaboración ya que el objetivo fundamental es identificar requisitos u responsabilidades sobre los objetos.

Los Diagramas de Colaboración destacan la organización de los objetos que participan en una interacción. Se construyen colocando en primer lugar los objetos que participan en la colaboración como nodos del grafo. En este tipo de diagrama se muestran las interacciones entre los objetos creando enlaces entre ellos y añadiendo mensajes a esos enlaces. El nombre del mensaje debería denotar el propósito del objeto invocante en la interacción con el objeto invocado.

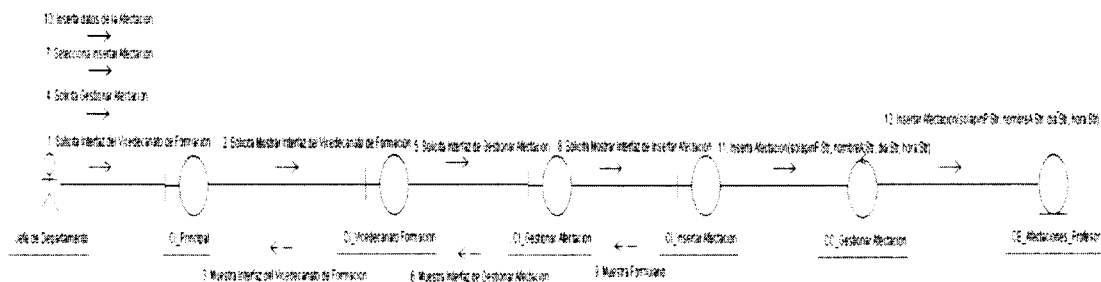


Figura 15: Diagrama de Colaboración del CUS Gestionar Afectación: Sección Insertar Afectación.

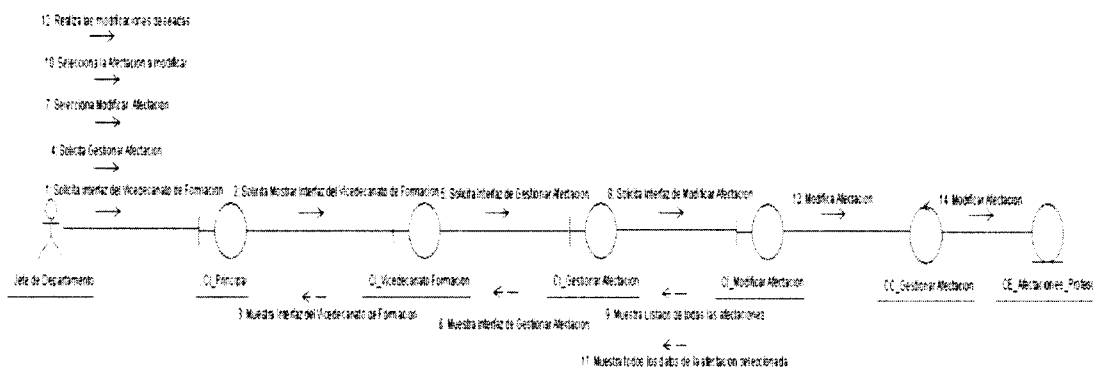


Figura 16: Diagrama de Colaboración del CUS Gestionar Afectación: Sección Modificar Afectación.

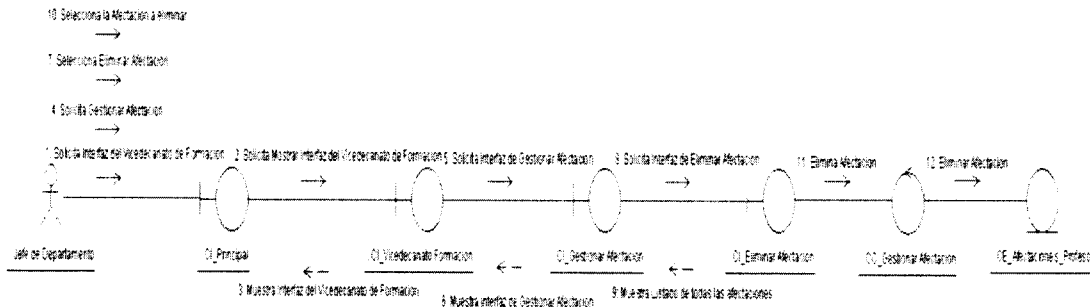


Figura 17: Diagrama de Colaboración del CUS Gestionar Afectación: Sección Eliminar Afectación.

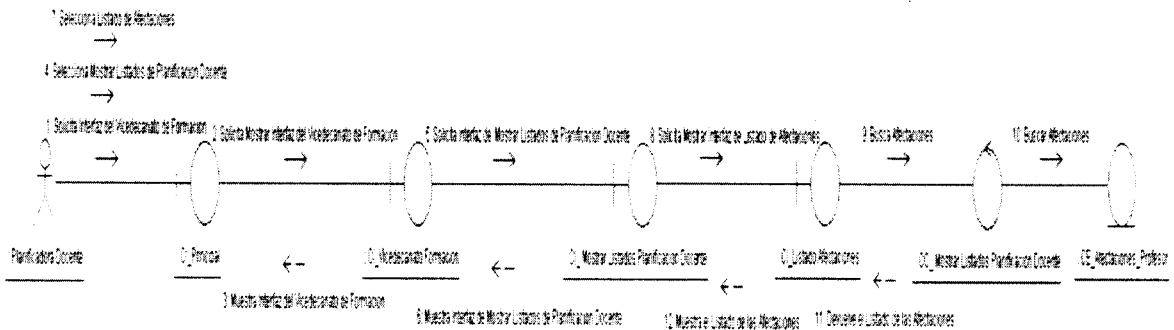


Figura 18: Diagrama de Colaboración del CUS Mostar Listados Planificación Docente: Sección Listado Afectaciones.

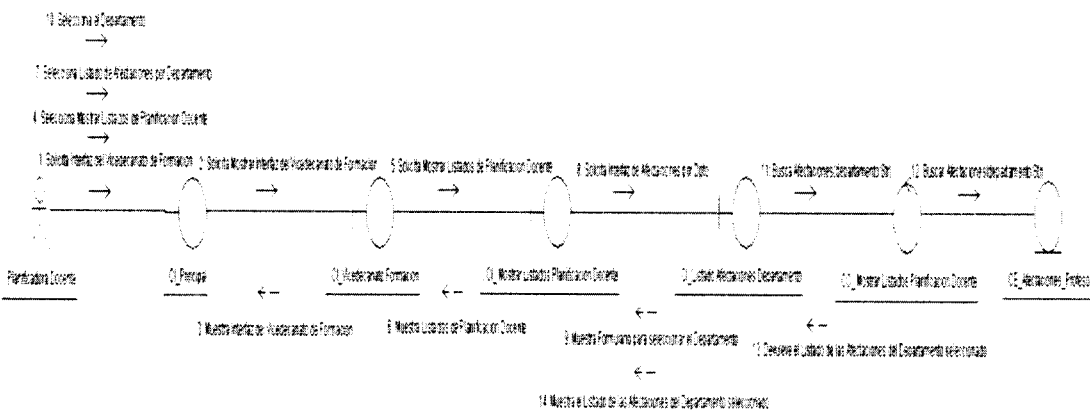


Figura 19: Diagrama de Colaboración del CUS Mostar Listados Planificación Docente: Sección Listado de Afectaciones por Departamento.

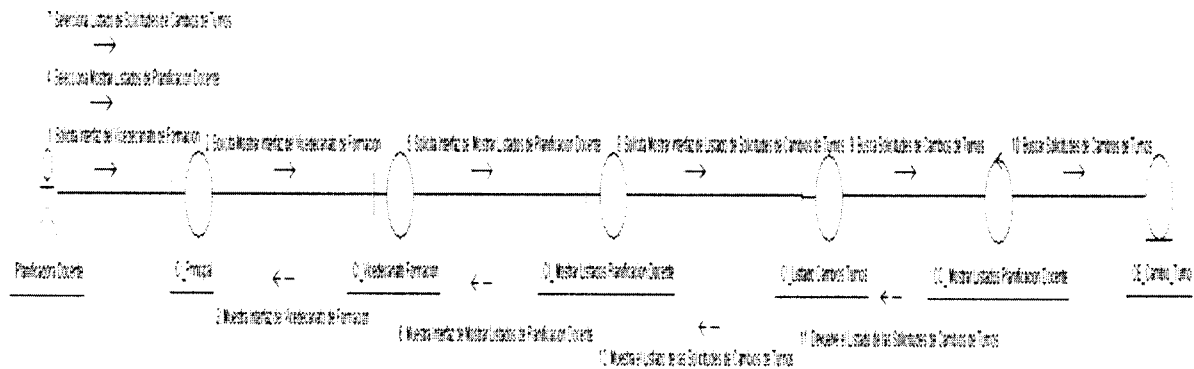


Figura 20: Diagrama de Colaboración del CUS Mostar Listados Planificación Docente: Sección Listado de Solicitudes de Cambio de Turno.

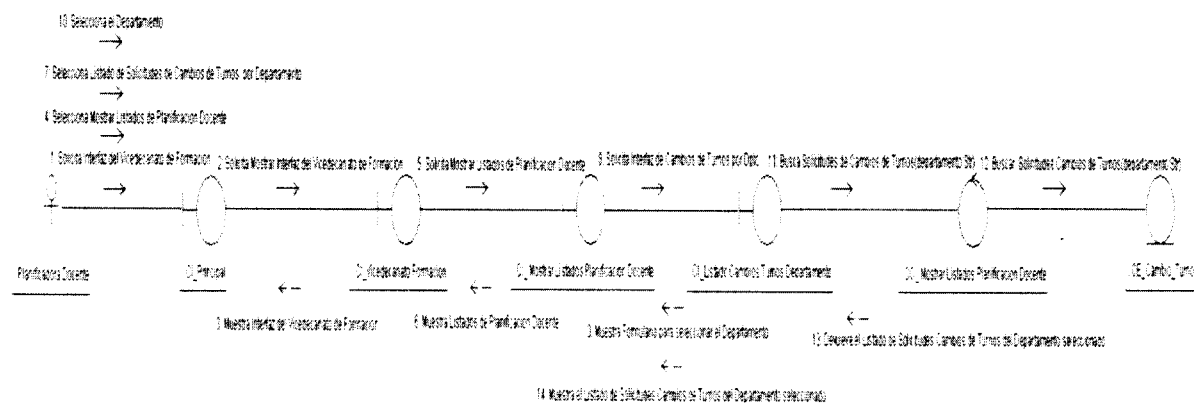


Figura 21: Diagrama de Colaboración del CUS Mostar Listados Planificación Docente: Sección Listado de Solicitudes de Cambio de Turno por Departamento.

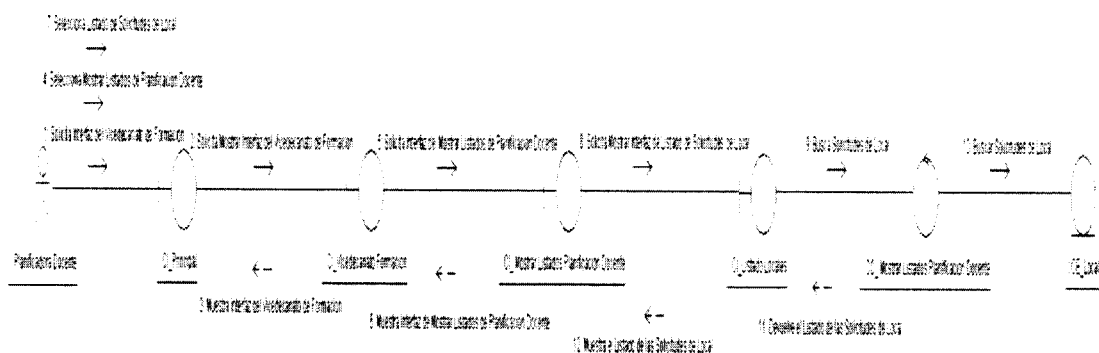


Figura 22: Diagrama de Colaboración del CUS Mostar Listados Planificación Docente: Sección Listado de Solicitud de Locales.

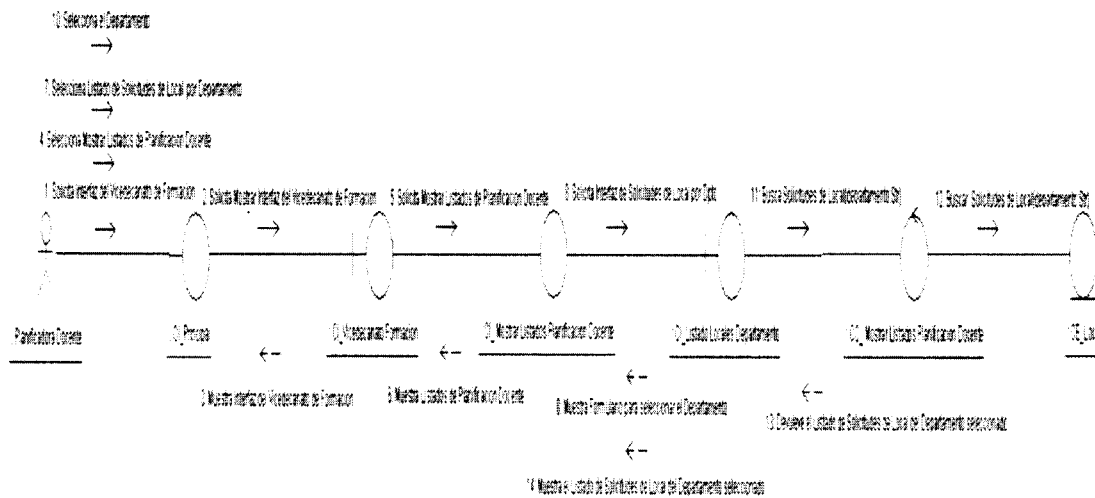


Figura 23: Diagrama de Colaboración del CUS Mostar Listados Planificación Docente: Sección Listado de Solicitud de Locales por Departamento.

4.3 Modelo de Diseño.

El Modelo de Diseño es un modelo de objetos que describe la realización física de los casos de uso centrándose en como los requisitos funcionales y no funcionales, junto con otras restricciones relacionadas con el entorno de implementación, tienen impacto en el sistema a desarrollar. Sirve de abstracción de la implementación, es específico para una implementación determinada y es utilizado como entrada fundamental de las actividades del flujo de trabajo antes mencionado. Está estructurado por clases que poseen un gran número de estereotipos, dependiendo del lenguaje de implementación a utilizar.

Para la confección del Diseño se tuvo en cuenta el framework escogido: Symfony que como ya se ha dicho anteriormente implementa el patrón arquitectónico MVC. La implementación la realiza de la siguiente manera:

La capa del modelo la divide en la capa de acceso a los datos y en la capa de abstracción de la base de datos. Las clases de esta capa se generan automáticamente. La librería Propel se encarga de esta generación automática, creando la estructura básica de las clases de la capa de acceso a los datos y generando automáticamente el código necesario. La abstracción de la base de datos es completamente invisible al programador, ya que la realiza otro componente específico llamado Creole.

La capa de la vista la separa en un layout y en plantillas. Normalmente, el layout es global en toda la aplicación o al menos en un grupo de páginas. Las plantillas sólo se encargan de visualizar las variables definidas en el controlador.

El controlador lo divide en un controlador frontal, que es único para cada aplicación, y las acciones, que incluyen el código específico del controlador de cada página.

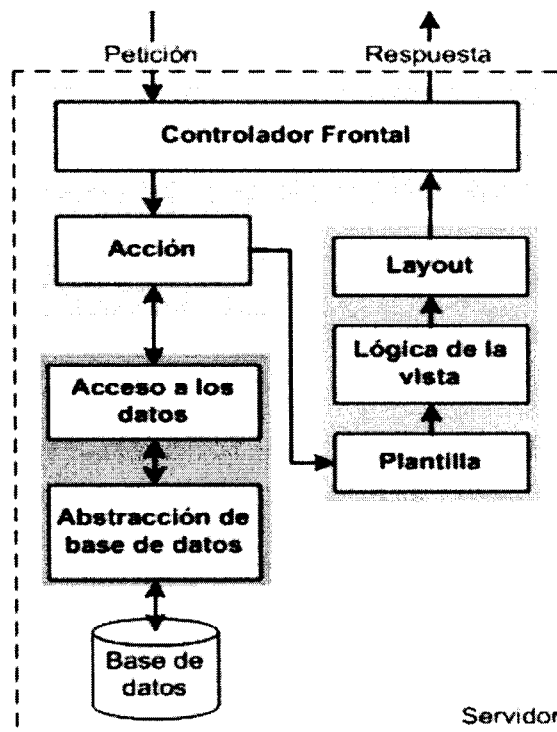


Figura 24: El Flujo de trabajo de Symfony.

Esta separación implementada por Symfony se ve reflejada en los Diagramas de Clases del Diseño, en los cuales además se reflejan componentes de Symfonys que son invisibles a los programadores pero que son utilizados internamente por el framework como son: user, el cual es usado en la autenticación de los usuarios; controller y view, los cuales son usados para el funcionamiento del controlador y el funcionamiento de la vista respectivamente, por mencionar algunos.

4.3.1 Diagramas de Clases del Diseño.

Los Diagramas de Clases del Diseño constituyen diagramas de estructura estática ya que representan la parte estática del sistema, representan las clases del diseño y sus relaciones.

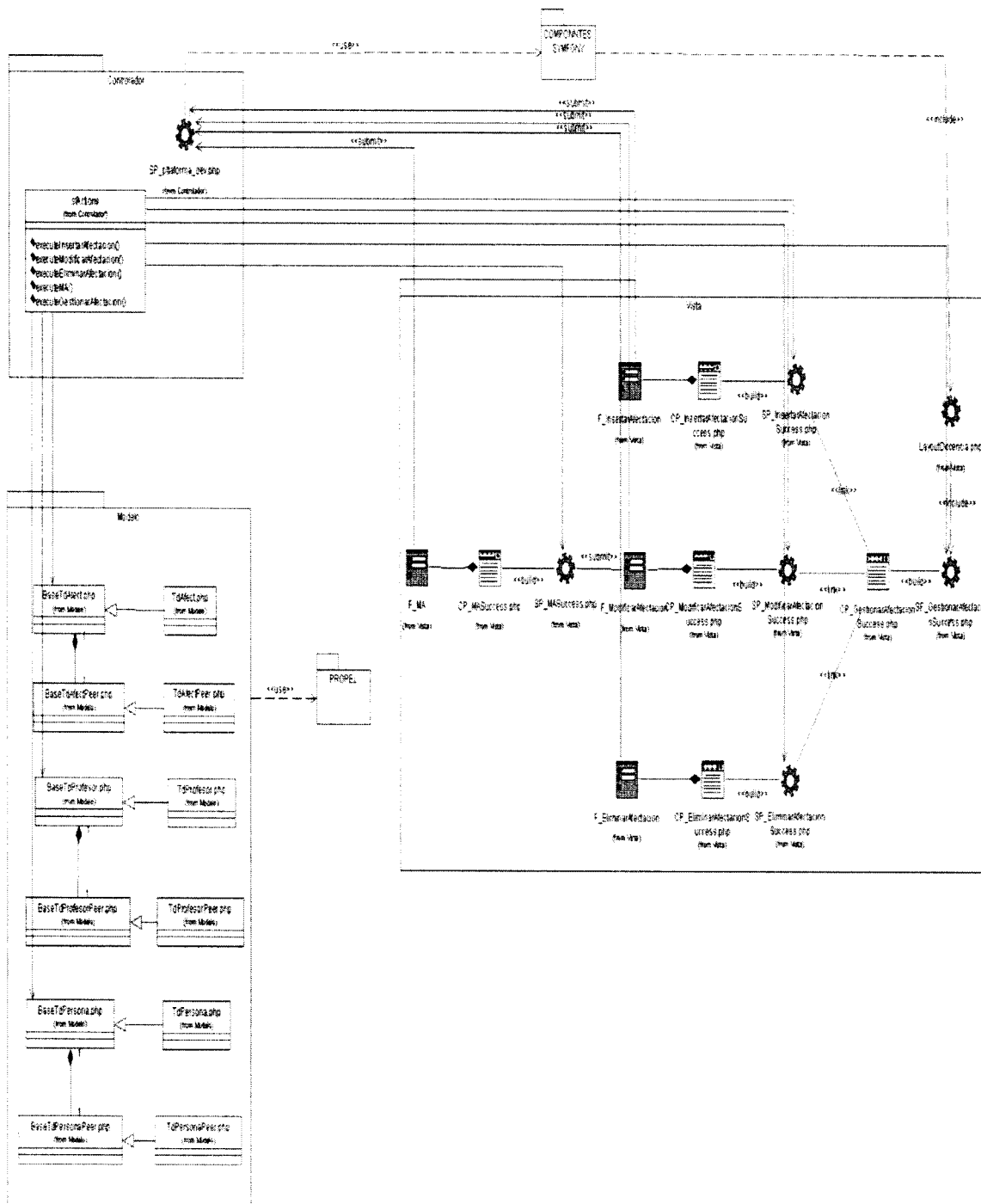


Figura 25: Diagrama de Clases del Diseño del CUS Gestionar Afectación.

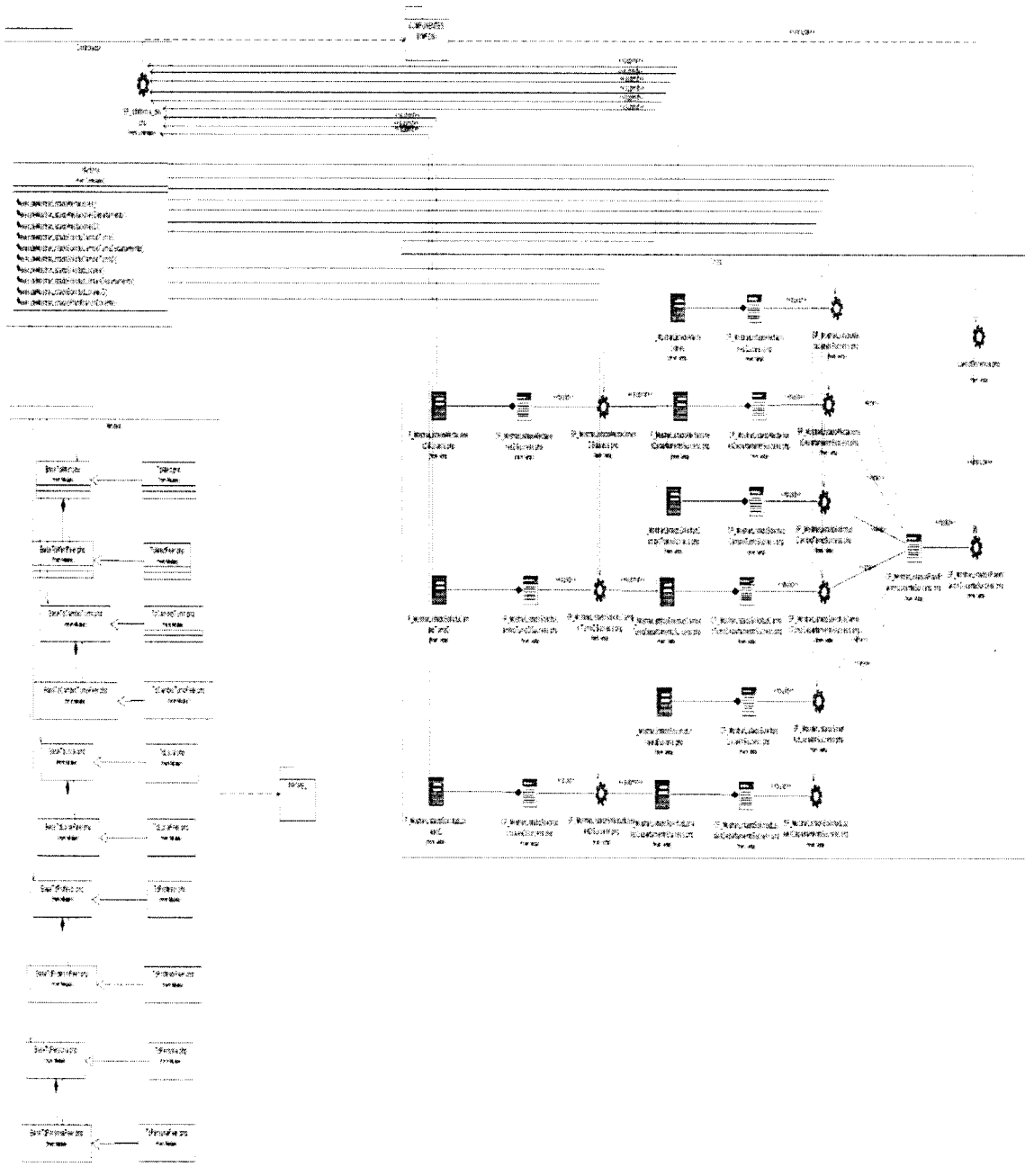


Figura 26: Diagrama de Clases del Diseño del CUS Mostrar Listados Planificación Docente.



4.4 Principios del Diseño.

4.4.1 Tratamiento de excepciones y seguridad del Sistema.

El tratamiento de los errores es fundamental para el buen funcionamiento de cualquier sistema, particularmente en el Subsistema Docente de la Facultad 9, se validarán todos los formularios de entrada de datos, para garantizar así que toda la información que se almacene en la base de datos sea correcta, por ejemplo si se dejan campos en blancos se indicará al usuario lo que está ocurriendo, si se introducen datos del tipo incorrecto, como por ejemplo, introducir en el campo del solapín letras, también se avisará al usuario, cuando sea posible se utilizarán campos de selección de información para evitar así que el usuario entre información errónea, además si el usuario olvida seleccionar la información, recibirá un mensaje indicándole su olvido. Si el usuario intenta introducir información que ya se encuentra en la base de datos, recibirá un mensaje indicándole que dicha información ya está almacenada en la base de datos. Para el tratamiento de los errores se utilizarán funciones JavaScript y funciones definidas por el framework que se utiliza.

La seguridad del Subsistema Docente de la Facultad 9 comienza con la autenticación que tienen que realizar los usuarios en la intranet de la facultad, estos se autentican con su usuario y contraseña del dominio UCI. Específicamente al Subsistema Docente de la Facultad 9 sólo tienen accesos los profesores y trabajadores del Vicedecanato de Formación y la Secretaría Docente de la Facultad, es decir que si un estudiante o trabajador de otra área se autentica y desea entrar al sistema no se le permite el acceso. Una vez dentro del sistema cada usuario tendrá acceso sólo a la información que según sus responsabilidades puede tener. En caso de que desee acceder a información fuera de su alcance recibirá un mensaje informándole que no tiene acceso a esa información. Además si los usuarios conocen la URL de cualquiera de las páginas web del sistema e intenta ponerla directamente en la barra de direcciones de su navegador recibirá un mensaje indicándole que no se ha autenticado y lo remitirá automáticamente a la intranet de la facultad.

4.4.2 Patrones de Diseño implementados por Symfony y utilizados en el Diseño del Subsistema Docente de la Facultad 9.

4.4.2.1 Patrones GRASP.

GRASP es un acrónimo que significa Patrones Generales de Software para Asignar Responsabilidad (General Responsibility Assignment Software Patterns). Los patrones GRASP describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones.



Patrón Creador: En la clase Actions se encuentran las acciones definidas para el Subsistema Docente de la Facultad 9 y se ejecutan cada una de ellas. En las acciones se crean los objetos de las clases que representan las entidades, evidenciando de este modo que la clase Actions es "creador" de dichas entidades.

Patrón Experto: Este es uno de los más utilizados, por ejemplo Propel es la librería externa experta que utiliza Symfony para realizar su capa de abstracción en el modelo, encapsula toda la lógica de los datos y son generadas las clases con todas las funcionalidades comunes de las entidades.

Patrón Alta Cohesión: Symfony permite asignar responsabilidades con una alta cohesión, por ejemplo la clase Actions tiene la responsabilidad de definir las acciones para las plantillas y colabora con otras para realizar diferentes operaciones, instanciar objetos y acceder a las properties, es decir, está formada por diferentes funcionalidades que se encuentran estrechamente relacionadas proporcionando que el software sea flexible frente a grandes cambios.

Patrón Controlador: Todas las peticiones Web son manejadas por un sólo controlador frontal, que es el punto de entrada único de toda la aplicación en un entorno determinado. Cuando el controlador frontal recibe una petición, utiliza el sistema de enrutamiento para asociar el nombre de una acción y el nombre de un módulo con la URL entrada por el usuario.

Patrón Bajo Acoplamiento: La clase Action hereda solamente de sfActions para lograr un bajo acoplamiento de clases.

4.4.2.2 Patrones GOF.

GOF significa Pandilla de los Cuatro (Gang of Four) deben su nombre al número de sus autores. Los patrones GOF de acuerdo a su propósito se clasifican en 3 categorías: Creacionales, Estructurales y de Comportamiento.

En la categoría Creacionales implementa:

Singleton (Instancia única): Garantiza la existencia de una única instancia para una clase y la creación de un mecanismo de acceso global a dicha instancia. En el controlador frontal hay una llamada a sfContext::getInstance(). En una acción, el método getContext(), un objeto muy útil que guarda una referencia a todos los objetos del núcleo de Symfony.

Abstract Factory (Fábrica abstracta): Permite trabajar con objetos de distintas familias de manera que las familias no se mezclen entre sí y haciendo transparente el tipo de familia concreta que se está usando. Cuando el framework necesita por ejemplo crear un nuevo objeto para una petición, busca en la definición de la factoría el nombre de la clase que se debe utilizar para esta tarea.

En la categoría Estructurales implementa:

Decorator (Envoltorio): Añade funcionalidad a una clase, dinámicamente. El archivo `layout.php`, que también se denomina plantilla global, almacena el código HTML que es común a todas las páginas de la aplicación, para no tener que repetirlo en cada página. El contenido de la plantilla se integra en el `layout`, o si se mira desde el otro punto de vista, el `layout` decora la plantilla. Este comportamiento es una implementación del patrón de diseño llamado Decorador.

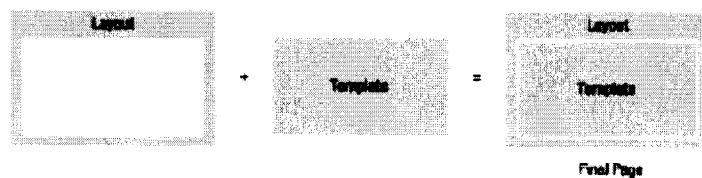


Figura 27: Plantilla decorada con un layout.

Composite (Objeto compuesto): Permite tratar objetos compuestos como si de uno simple se tratase. Sirve para construir objetos complejos a partir de otros más simples y similares entre sí, gracias a la composición recursiva y a una estructura en forma de árbol. Esto simplifica el tratamiento de los objetos creados, ya que al poseer todos ellos una interfaz común, se tratan todos de la misma manera.

4.5 Diseño de la Base de Datos

Proceso por el que se determina la organización de la base de datos, incluidos su estructura y contenido.

Los pasos en el diseño de la BD son:

1. Definir las clases persistentes.
2. Refinar las clases.
3. Clasificar las clases y los atributos.
4. Realizar el diagrama de clases.
5. Realizar el diagrama de transición de estado.

6. Obtener las restricciones estáticas y las fórmulas dinámicas.
7. Convertir las clases al medio de almacenamiento.

4.5.1 Diagrama de Clases Persistentes.

Las clases persistentes son aquellas clases que tienen la capacidad de mantener su valor en el espacio y en el tiempo. Son aquellas cuyos objetos deben ser almacenados en una base de datos.

El Diagrama de Clases Persistentes muestra las clases clasificadas como persistentes y las relaciones entre ellas.

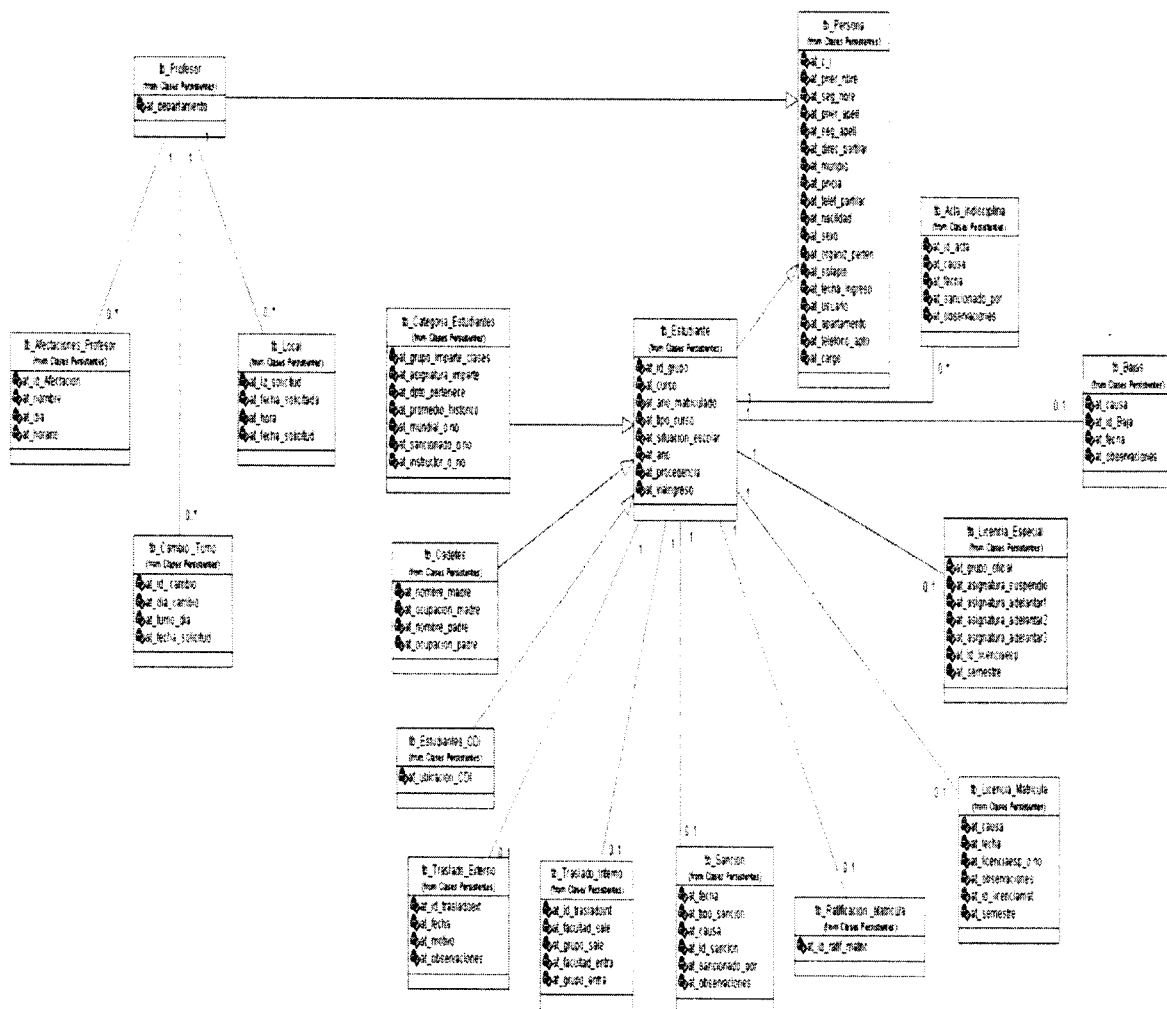


Figura 28: Diagrama de Clases Persistentes.



4.5.2 Modelo de Datos.

Un Modelo de Datos es una colección de conceptos que se emplean para describir la estructura de una base de datos. Esa colección de conceptos incluye entidades, atributos y relaciones. El Modelo de Datos se crea a partir del Diagrama de Clases Persistentes, este modelo se utiliza para diseñar la Base de Datos.

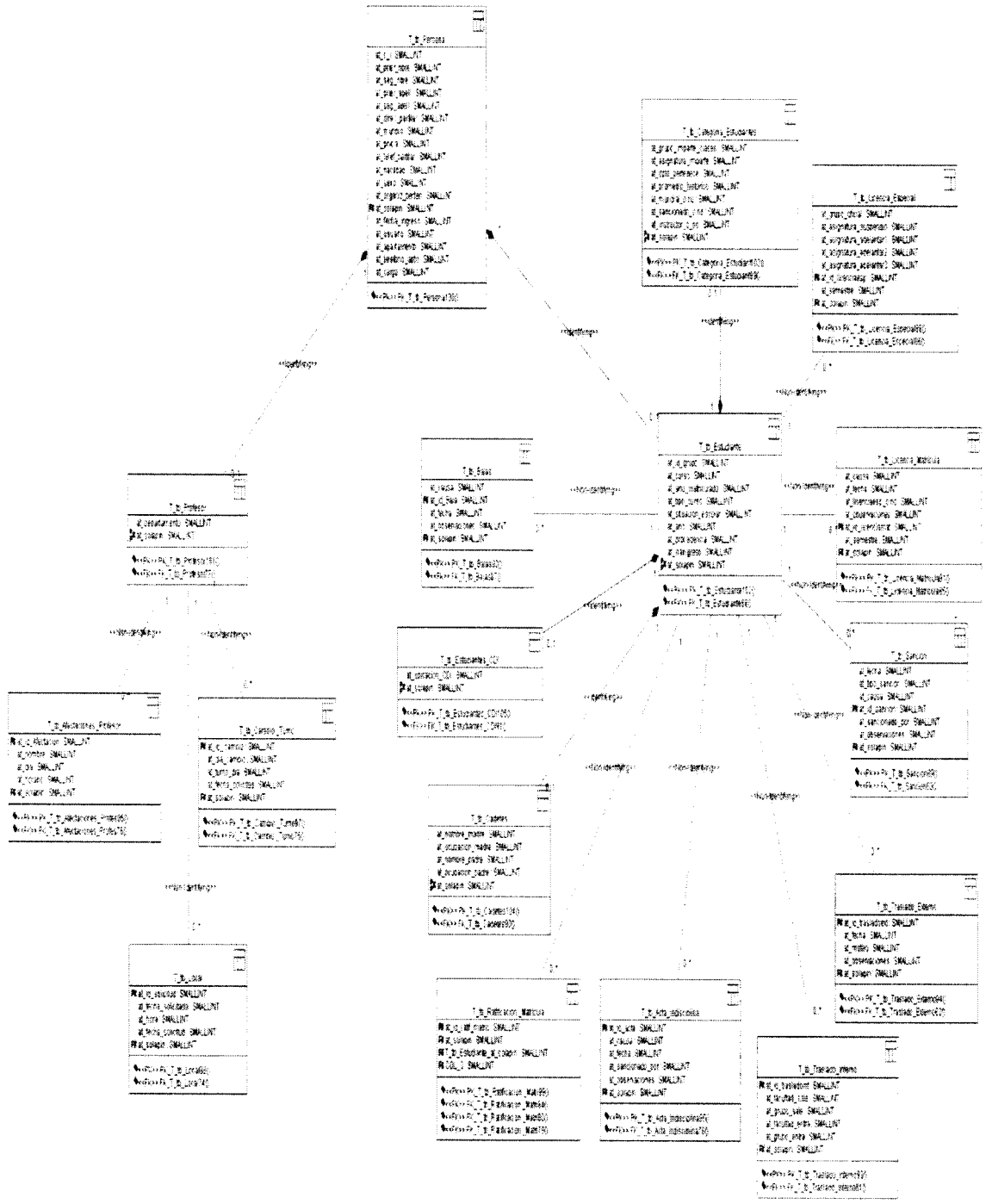


Figura 29: Modelo de Datos.

4.6 Modelo de Despliegue

El Modelo de Despliegue es un modelo de objetos que describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos de cómputo. El Modelo de Despliegue se utiliza como entrada fundamental de las actividades de diseño e implementación debido a que la distribución del sistema tiene una influencia principal en su diseño. En dicho modelo cada nodo representa un recurso de cómputo, normalmente un procesador o un dispositivo de hardware similar. Los nodos poseen relaciones que representan medios de comunicación entre ellos.

La aplicación necesitará tres procesadores: el servidor de la aplicación, el cual se conectará directamente al servidor de base de datos mediante el protocolo de comunicación TCP/IP y las PC clientes se conectarán a dicho servidor mediante el protocolo HTTP. Las PCs clientes utilizarán una impresora la cual será un recurso compartido en la red utilizando un Print Server mediante el protocolo TCP/IP. Además el servidor de aplicación estará conectado a: el servidor UDDI que es el servidor de los web services que existen en la universidad y de los cual la base de datos de la facultad se actualizará, el servidor del dominio UCI donde se verificara que el usuario y la contraseña que se utilicen en la autenticación son los correctos.

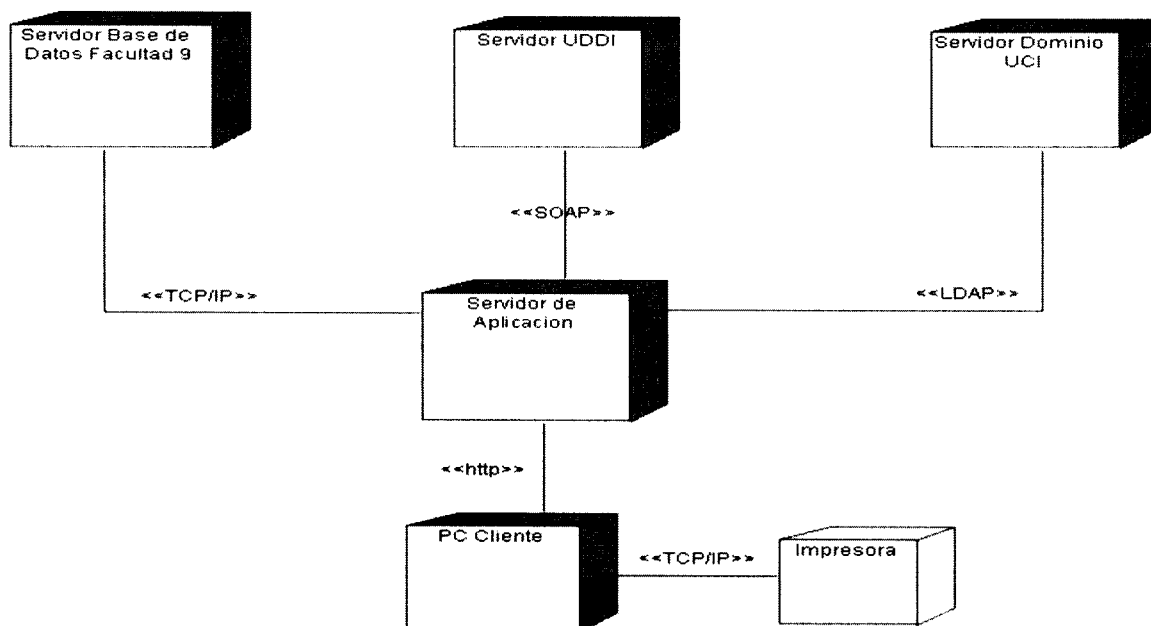


Figura 30: Diagrama de Despliegue.



4.7 Modelo de Implementación

El Modelo de Implementación describe como los elementos del diseño se implementan en términos de componentes: ficheros de código fuente, ejecutables, etc. Este modelo describe como se organizan los componentes de acuerdo a los mecanismos de estructuración, el lenguaje o lenguajes de programación utilizados y como dependen los componentes unos de otros.

4.7.1 Diagramas de Componentes.

Los Diagramas de Componentes son diagramas de estructura estática ya que se utilizan para modelar una vista estática del sistema. Muestran la organización y las dependencias lógicas entre un conjunto de componentes software como son: componentes de código fuente, librerías o componentes de código ejecutable.

No es necesario realizar sólo un diagrama de componentes que incluya todos los componentes del sistema sino que este puede dividirse en varios diagramas. En este trabajo los Diagramas de Componentes se realizarán por Casos de Uso.

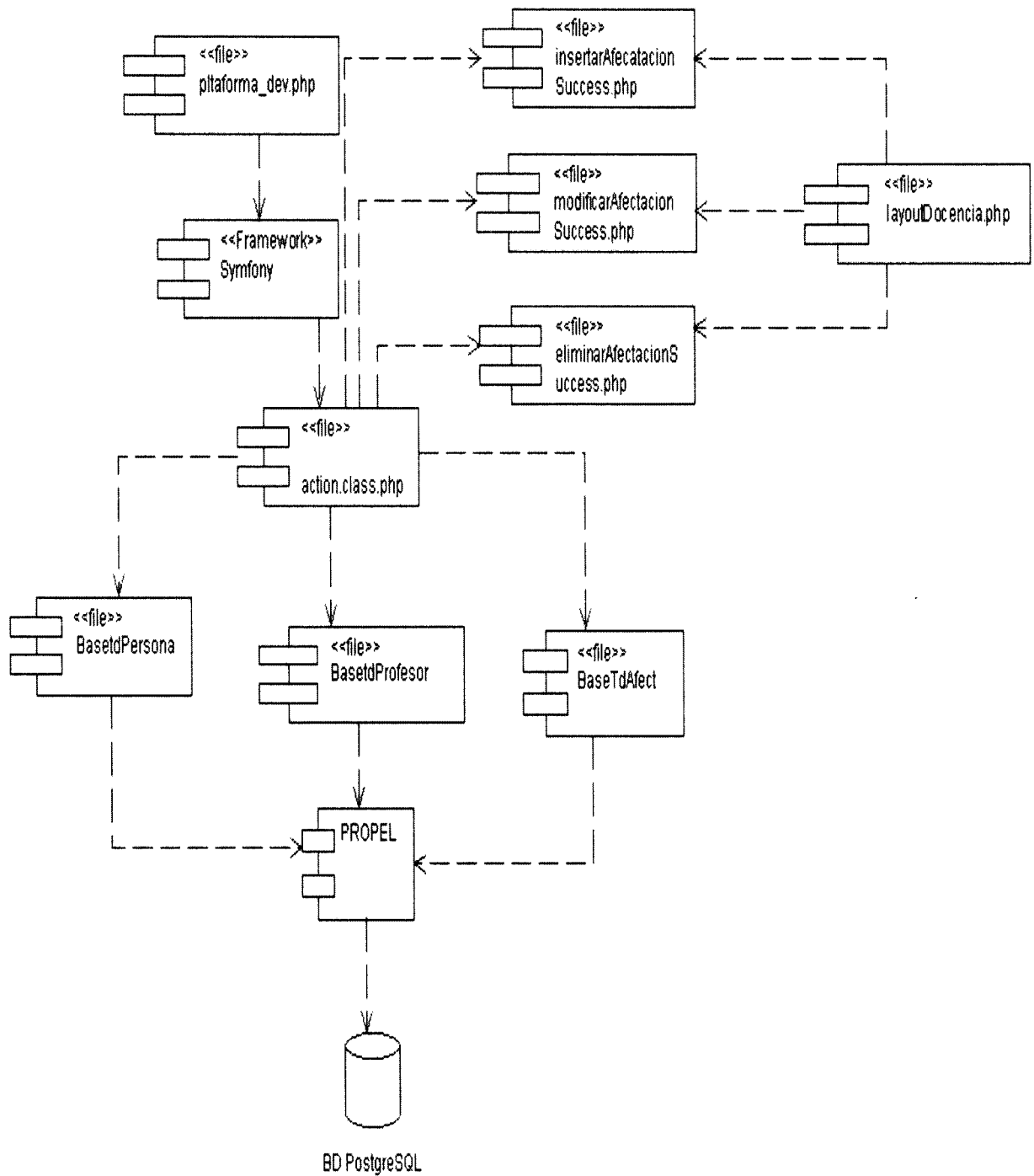


Figura 31: Diagrama de Componentes del CU Gestionar Afectación.

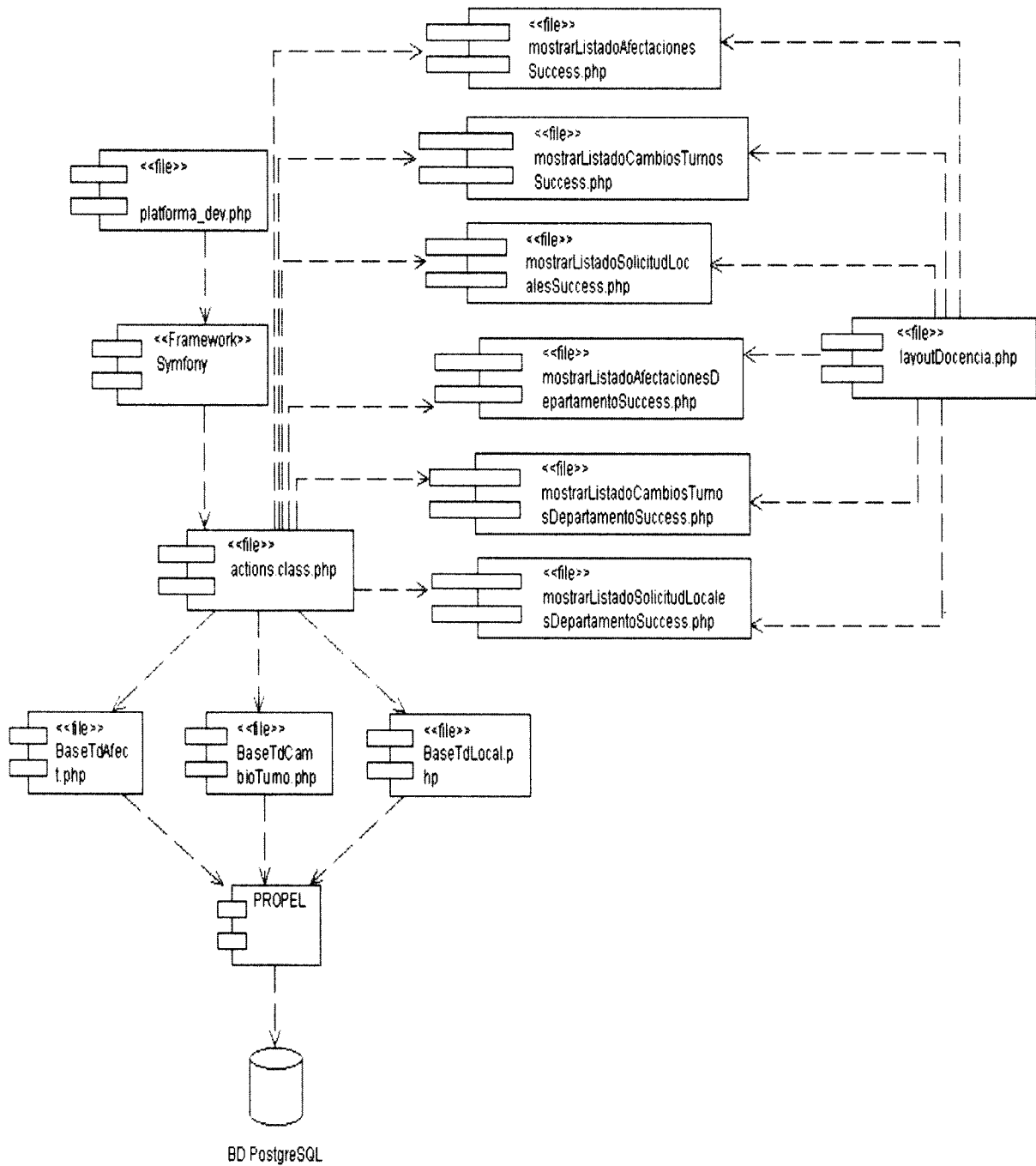


Figura 32: Diagrama de Componentes del CU Mostrar Listados Planificación Docente.

4.8 Conclusiones

En este capítulo fueron expuestos diferentes elementos que ilustran cómo está construido el sistema, en términos de clases del análisis y del diseño. Se elaboraron para cada Caso de Uso un Diagrama de Clases del Análisis, así como un Diagrama de Colaboración para cada sección que componen a estos Casos de Uso y un Diagrama de Clases del Diseño. Se elaboró el diagrama de clases persistentes, a partir de este se obtuvo el Modelo de Datos.

En este capítulo también se logró un refinamiento de la vista de la arquitectura a través del Modelo de Despliegue, así como, se especificaron las dependencias entre los diferentes componentes en los Diagramas de Componentes.

Conclusiones

La realización de este trabajo responde a la necesidad de buscar una alternativa en el Vicedecanato de Formación, la Planificación Docente y la Secretaría Docente de la Facultad 9 de la Universidad de las Ciencias Informáticas para el control de los procesos que allí se realizan, teniendo en cuenta que el control de los mismos se realiza de forma manual y representa una labor compleja, que requiere un lento y cuidadoso tratamiento. Con el objetivo de lograr una solución lo más óptima posible se estudió como se llevan a cabo los procesos, cuales son las necesidades de los clientes. Se analizaron las herramientas, técnicas, tecnologías, metodologías de desarrollo más factibles para el desarrollo de la aplicación Web, además se buscaron referencias en el ámbito internacional, nacional y dentro de la propia universidad de sistemas que manejaran de alguna manera el tema.

Luego de todo este estudio previo se llegó a la conclusión de que el sistema sería implementado en PHP, Gestor de Bases de Datos PostgreSQL, Servidor Web Apache, y la metodología de desarrollo RUP. A partir de este momento se modeló el negocio propuesto, se detectaron actores y trabajadores, además de identificar cuales serían las actividades a automatizar.

Se definieron los requerimientos funcionales y no funcionales, se representó el diagrama de Casos de Uso del Sistema y se describieron los mismos para una mejor comprensión y entendimiento de sus funcionalidades. Se realizaron los Diagramas de Clases del Análisis, del Diseño y el Diagrama de Clases Persistentes, entre otros. Se estructuró el Modelo de Datos, que es la representación física de la base de datos del sistema. Posteriormente se elaboró el Modelo de Despliegue y los Diagramas de Componentes.

Luego de todo este proceso se concluye que el sistema para la gestión de la información docente significa sin duda una mejora para facilitar el trabajo en el Vicedecanato de Formación, la Planificación Docente y la Secretaría Docente de la Facultad 9, además cumple con los objetivos propuestos.



Recomendaciones

Se recomienda:

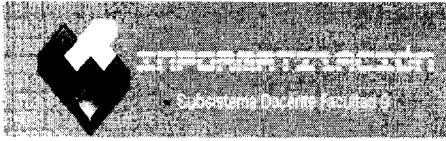
- Poner a prueba el sistema durante el período de tiempo necesario como para comprobar su desempeño y que las funcionalidades del sistema se corresponden con los requisitos del usuario.
- Mejorar el Diseño de la Aplicación.
- Mejorar y ampliar las funcionalidades del sistema.



REFERENCIAS BIBLIOGRÁFICAS

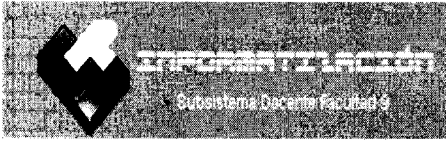
Referencias Bibliográficas

1. ALIAGA, F. y BARTOLOMÉ, A. *El Impacto de las Nuevas Tecnologías en Educación*. 2005, Disponible en: <http://www.uv.es/aliaga/curriculum/Aliaga&Bartolome-2005-borrador.pdf>.
2. PLANAS, I. G. *Intervención Especial de Ignacio González Planas Ministro de la Informática y las Comunicaciones de Cuba en Mesa Redonda de Alto Nivel*. 2003, Disponible en: http://www.cubaminrex.cu/Sociedad_Informacion/Fasel.htm#3.
3. CUBA, M. D. L. I. Y. L. T. D. *¿Qué es Informatización de la Sociedad?* Disponible en: <http://www.mic.gov.cu/hinfosoc.aspx>.
4. VALDÉS, Y. V. y PÉREZ, L. R. *Sistema automatizado para la gestión de información de la secretaría docente: Módulo bajas, traslados y comisiones disciplinarias*. Universidad de las Ciencias Informáticas., 2007.
5. MARTÍN, R. R. y APARICIO, L. E. *Glosario de Términos Bibliotecológicos y de Ciencias de la Información* Disponible en: http://www.uh.cu/facultades/fcom/portal/interes_glosa_terminos.htm.
6. ENCARTA, E. *Sistema (Informática)* Disponible en: [http://es.encarta.msn.com/encyclopedia_761560074/Sistema_\(inform%C3%A1tica\).html](http://es.encarta.msn.com/encyclopedia_761560074/Sistema_(inform%C3%A1tica).html).
7. SKUPIEÑ, D. A. *MANUAL SIMPLIFICADO DE OPERACION Y CONTROL - COMANDOS ESPECIALES - AS/400* Disponible en: <http://ecom.chaco.gov.ar/AS400/Subsistemas.htm>.
8. DELGADO, Y. C. *GESTACAD. SISTEMA PARA LA GESTIÓN ACADÉMICA* Disponible en: <http://www.posgrados.frc.utn.edu.ar/congreso/trabajos/7.doc>.
9. HERNÁNDEZ, A. I. C. y PÉREZ, D. G. *SUBSISTEMA DE GESTIÓN DE LOS DEPARTAMENTOS DOCENTES DE LA FACULTAD 9*. Universidad de las Ciencias Informáticas., 2007.
10. EUROPEO., C. D. L. C. E. C. D. L. C. A. C. Y. A. P. *Definición de las TIC*. Disponible en: http://www.ecomaccess.com/iim/docs/ict_es.doc.
11. MUSCIANO, C. y KEMEDY, B. *HTML La Guía Completa*. 1999, Disponible en: <http://bibliodoc.uci.cu/pdf/reg01313.pdf>.
12. POWELL, T. A. *Diseño de Stios Web Manual de Referencia*. 2000, Disponible en: <http://bibliodoc.uci.cu/pdf/reg01293.pdf>.
13. WELLING, L. y THOMSON, L. *Desarrollo Web con PHP y MySQL*. Disponible en: <http://bibliodoc.uci.cu/pdf/reg02819.pdf>.
14. BRANDENDAUGH, J. *Aplicaciones JavaScript*. 2000, Disponible en: <http://bibliodoc.uci.cu/pdf/reg00009.pdf>.



REFERENCIAS BIBLIOGRÁFICAS

15. PROAÑO, D. J. B. *ANÁLISIS COMPARATIVO DE BASES DE DATOS DE CÓDIGO ABIERTO VS CÓDIGO CERRADO*. 2006, Disponible en: <http://www.mysql-hispano.org/articulos/num43/analisis-comparativo.pdf>.
16. POSTGRESQL, E. E. D. D. D. *Manual del Usuario de PostgreSQL* Disponible en: <http://okulto.net/Postgresql-es/web/navegable/user/user.html>.
17. GONZÁLEZ, C. D. *Base de Datos PostgreSQL, SQL avanzado y PHP* Disponible en: <http://www.usabilidadweb.com.ar/postgre.php>.
18. RUMBAUGH, J.; JACOBSON, I., et al. *El Lenguaje Unificado de Modelado*. 2000, Disponible en: <http://bibliodoc.uci.cu/pdf/reg03050.pdf>.
19. *Conferencia 1: Introducción a la Ingeniería de Software*. Disponible en: http://teleformacion.uci.cu/file.php/42/Clases_Curso_2007-2008/conferencias/Conferencia_1/Profesores/Conferencia_1.pdf.
20. RUBIO, F. Ó. G. y SANTOS, C. B. *Metodologías de Desarrollo de Software*. Disponible en: http://alarcos.inf-cr.uclm.es/per/fgarcia/isoftware/doc/tema3_1xh.pdf.
21. ESPAÑA., M. D. A. P. D. *Métrica Version 3: Introducción*. Disponible en: <http://www.csi.map.es/csi/metrica3/introduccion.pdf>.
22. ESCRIBANO, G. F. *Introducción a Extreme Programming*. Disponible en: <http://www.dsi.uclm.es/asignaturas/42551/trabajosAnteriores/Presentacion-XP.pdf>.
23. RUMBAUGH, J.; JACOBSON, I., et al. *El Proceso Unificado de Desarrollo de Software*. 2000, Disponible en: <http://bibliodoc.uci.cu/pdf/reg00060.pdf>.
24. *Qué es Kumbia Framework?* Disponible en: <http://www.kumbia.org/>.
25. GUTIERREZ, A. F. *Porque Programar debería ser más fácil*. 2008, Disponible en: <http://www.kumbia.org/LibroDeKumbia.pdf>.
26. WINDMÜLLER, J. *Introducción a CakePHP*. Disponible en: http://cakephp-es.org/doku.php?id=introduccion_a_cakephp.
27. LIBROSWEB.ES. *Symfony en pocas palabras*. nº Disponible en: http://www.librosweb.es/symfony/capitulo1/symfony_en_pocas_palabras.html.
28. BLANCO, R. G. y TOBALINA, S. P. *LESE-2 Introducción a Rational Rose* Disponible en: <http://ateam.lsi.upc.es/~es-e/web/documents/lab/0304Q2/lessons/lese-2/LESE-2%20-%20Introduccion%20a%20Rational%20Rose.ppt?PHPSESSID=0ade5aba4d6b566cb794d3b7def7cd08>.



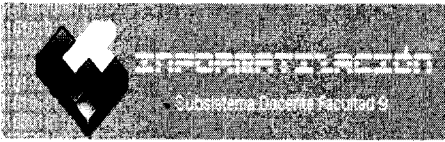
REFERENCIAS BIBLIOGRÁFICAS

29. *SERVIDOR WEB APACHE*. Disponible en: tutatis.albacete.org/trabajos2005/Apache/Apache.ppt.
30. *Arquitectura del servidor Apache*. Disponible en: <http://www.desarrolloweb.com/articulos/1112.php>.
31. HEREDIA, M. G. *Los mejores IDEs para Php*. 2007, Disponible en: <http://www.tufuncion.com/ide-php>.
32. ANAISA HERNÁNDEZ GONZÁLEZ. *IDENTIFICACIÓN DE PROCESOS DE NEGOCIO*. 2004, Disponible en: <http://www.cujae.edu.cu/ediciones/Revistas/Industrial/Vol-XXV/3-2004/83-88%20Identificaci%C3%B3n%20de%20procesos.pdf>.



Bibliografía

1. ALIAGA, F. y BARTOLOMÉ, A. *El Impacto de las Nuevas Tecnologías en Educación*. 2005, Disponible en: <http://www.uv.es/aliaga/curriculum/Aliaga&Bartolome-2005-borrador.pdf>.
2. PLANAS, I. G. *Intervención Especial de Ignacio González Planas Ministro de la Informática y las Comunicaciones de Cuba en Mesa Redonda de Alto Nivel*. 2003, Disponible en: http://www.cubaminrex.cu/Sociedad_Informacion/Fasel.htm#3.
3. CUBA, M. D. L. I. Y. L. T. D. *¿Qué es Informatización de la Sociedad?* Disponible en: <http://www.mic.gov.cu/hinfosoc.aspx>.
4. VALDÉS, Y. V. y PÉREZ, L. R. *Sistema automatizado para la gestión de información de la secretaría docente: Módulo bajas, traslados y comisiones disciplinarias*. Universidad de las Ciencias Informáticas., 2007.
5. MARTÍN, R. R. y APARICIO, L. E. *Glosario de Términos Bibliotecológicos y de Ciencias de la Información* Disponible en: http://www.uh.cu/facultades/fcom/portal/interes_glosa_terminos.htm.
6. ENCARTA, E. *Sistema (Informática)* Disponible en: [http://es.encarta.msn.com/encyclopedia_761560074/Sistema_\(inform%C3%A1tica\).html](http://es.encarta.msn.com/encyclopedia_761560074/Sistema_(inform%C3%A1tica).html).
7. SKUPIEÑ, D. A. *MANUAL SIMPLIFICADO DE OPERACION Y CONTROL - COMANDOS ESPECIALES - AS/400* Disponible en: <http://ecom.chaco.gov.ar/AS400/Subsistemas.htm>.
8. DELGADO, Y. C. *GESTACAD. SISTEMA PARA LA GESTIÓN ACADÉMICA* Disponible en: <http://www.posgrados.frc.utn.edu.ar/congreso/trabajos/7.doc>.
9. HERNÁNDEZ, A. I. C. y PÉREZ, D. G. *SUBSISTEMA DE GESTIÓN DE LOS DEPARTAMENTOS DOCENTES DE LA FACULTAD 9*. Universidad de las Ciencias Informáticas., 2007.
10. EUROPEO., C. D. L. C. E. C. D. L. C. A. C. Y. A. P. *Definición de las TIC*. Disponible en: http://www.ecomaccess.com/iim/docs/ict_es.doc.
11. MUSCIANO, C. y KEMEDY, B. *HTML La Guía Completa*. 1999, Disponible en: <http://bibliodoc.uci.cu/pdf/reg01313.pdf>.
12. POWELL, T. A. *Diseño de Stios Web Manual de Referencia*. 2000, Disponible en: <http://bibliodoc.uci.cu/pdf/reg01293.pdf>.
13. WELLING, L. y THOMSON, L. *Desarrollo Web con PHP y MySQL*. Disponible en: <http://bibliodoc.uci.cu/pdf/reg02819.pdf>.



14. VALDÉS, D. P. *Los diferentes lenguajes de programación para la web*. Disponible en: <http://www.maestrosdelweb.com/principiantes/los-diferentes-lenguajes-de-programacion-para-la-web/>
15. BRANDENDAUGH, J. *Aplicaciones JavaScript*. 2000, Disponible en: <http://bibliodoc.uci.cu/pdf/reg00009.pdf>.
16. PROAÑO, D. J. B. *ANALISIS COMPARATIVO DE BASES DE DATOS DE CODIGO ABIERTO VS CODIGO CERRADO*. 2006, Disponible en: <http://www.mysql-hispano.org/articulos/num43/analisis-comparativo.pdf>.
17. POSTGRESQL, E. E. D. D. *Manual del Usuario de PostgreSQL* Disponible en: <http://okulto.net/Postgresql-es/web/navegable/user/user.html>.
18. GONZÁLEZ, C. D. *Base de Datos PostgreSQL, SQL avanzado y PHP* Disponible en: <http://www.usabilidadweb.com.ar/postgre.php>.
19. RUMBAUGH, J.; JACOBSON, I., et al. *El Lenguaje Unificado de Modelado*. 2000, Disponible en: <http://bibliodoc.uci.cu/pdf/reg03050.pdf>.
20. *Conferencia 1: Introducción a la Ingeniería de Software*. Disponible en: http://teleformacion.uci.cu/file.php/42/Clases_Curso_2007-2008/conferencias/Conferencia_1/Profesores/Conferencia_1.pdf.
21. RUBIO, F. Ó. G. y SANTOS, C. B. *Metodologías de Desarrollo de Software*. Disponible en: http://alarcos.inf-cr.uclm.es/per/fgarcia/isoftware/doc/tema3_1xh.pdf.
22. ESPAÑA., M. D. A. P. D. *Metrica Version 3: Introduccion*. Disponible en: <http://www.csi.map.es/csi/metrica3/introduccion.pdf>.
23. ESCRIBANO, G. F. *Introducción a Extreme Programming*. Disponible en: <http://www.dsi.uclm.es/asignaturas/42551/trabajosAnteriores/Presentacion-XP.pdf>.
24. RUMBAUGH, J.; JACOBSON, I., et al. *El Proceso Unificado de Desarrollo de Software*. 2000, Disponible en: <http://bibliodoc.uci.cu/pdf/reg00060.pdf>.
25. *Qué es Kumbia Framework?* Disponible en: <http://www.kumbia.org/>.
26. GUTIERREZ, A. F. *Porque Programar debería ser más fácil*. 2008, Disponible en: <http://www.kumbia.org/LibroDeKumbia.pdf>.
27. WINDMÜLLER, J. *Introducción a CakePHP*. Disponible en: http://cakephp-es.org/doku.php?id=introduccion_a_cakephp.



BIBLIOGRAFÍA

28. LIBROSWEB.ES. *Symfony en pocas palabras*. nº Disponible en: http://www.librosweb.es/symfony/capitulo1/symfony_en_pocas_palabras.html.
29. BLANCO, R. G. y TOBALINA, S. P. *LESE-2 Introducción a Rational Rose* Disponible en: <http://ateam.lsi.upc.es/~es-e/web/documents/lab/0304Q2/lessons/lese-2/LESE-2%20-%20Introduccion%20a%20Rational%20Rose.ppt?PHPSESSID=0ade5aba4d6b566cb794d3b7def7cd08>.
30. PAVÓN, E. L. *Visual Paradigm, una herramienta de lo más útil*. . 2007, Disponible en: <http://sliion2000.blogspot.com/2007/04/visual-paradigm-una-herramienta-de-lo.html>.
31. *SERVIDOR WEB APACHE*. Disponible en: tutatis.albacete.org/trabajos2005/Apache/Apache.ppt.
32. *Arquitectura del servidor Apache*. Disponible en: <http://www.desarrolloweb.com/articulos/1112.php>.
33. HEREDIA, M. G. *Los mejores IDEs para Php*. 2007, Disponible en: <http://www.tufuncion.com/ide-php>.
34. ANAISA HERNÁNDEZ GONZÁLEZ. *IDENTIFICACIÓN DE PROCESOS DE NEGOCIO*. 2004, Disponible en: <http://www.cujae.edu.cu/ediciones/Revistas/Industrial/Vol-XXV/3-2004/83-88%20Identificaci%C3%B3n%20de%20procesos.pdf>.
35. POTENCIER, F. y ZANINOTTO, F. *Symfony: la guía definitiva*. 2007, nº Disponible en: <http://www.librosweb.es/symfony>.



Glosario de Términos

ASI: Análisis del Sistema de Información.

ASP: Active Server Pages.

CDI: Centro de Diagnóstico Integral.

CECAM: Centro de Cibernética Aplicada a la Medicina.

CSI: Construcción del Sistema de Información.

CU: Caso de Uso.

CUS: Caso de Uso del Sistema.

DDL: Lenguaje de Definición de Datos.

DIS: Diseño del Sistema de Información.

DML: Lenguaje de Manipulación de Datos.

ECMA: Organización internacional basada en membrecías de estándares para la comunicación y la información.

EVS: Estudio de Viabilidad del Sistema.

GRASP: Patrones Generales de Software para Asignar Responsabilidad (General Responsibility Assignment Software Patterns).

GestAcad: Sistema para la Gestión Académica.

GOF: (Pandilla de los Cuatro) Patrones de Diseño.

HTML: Hypertext Markup Language.

HTTP: Protocolo de transferencia de hipertexto (HyperText Transfer Protocol).

IAS: Implantación y Aceptación del Sistema.



GLOSARIO DE TÉRMINOS

ICSW: Industria Cubana del Software.

IDE: Entorno de Desarrollo Integrado.

IPUEC: Instituto Pre-Universitario de Escuela en el Campo.

IPVCE: Instituto Pre-Universitario Vocacional de Ciencias Exactas.

ISO: Organización Internacional para la Estandarización (International Organization for Standardization).

JSP: Java Server Pages.

MINFAR: Ministerio de las Fuerzas Armadas Revolucionarias.

MININT: Ministerio del Interior.

MVC: Modelo Vista Controlador.

MVCC: Acceso Concurrente Multiversión.

ODBC: Estándar de conectividad abierta de base de datos.

ORDBMS: Sistema de Gestión de Bases de Datos Objeto-Relacionales.

PCC: Partido Comunista de Cuba.

PDF: Formato de Documento Portátil (Portable Document Format).

PHP: PHP Hypertext Pre-processor.

RD: Resolución Decanal.

RDBMS: Sistema de Administración de Bases de Batos Relacionales.

RUP: Proceso Unificado de Desarrollo de Software.

SAIDO: Sistema Automatizado de Información Docente.

SAIDO-F: Sistema automatizado de información docente, facultades.

SID: Sistema de Información Docente.



GLOSARIO DE TÉRMINOS

SQL: Structured Query Language.

TECADIES: Técnico en Trabajo Docente Investigativo de la Educación Superior.

TIC: Tecnologías de la Información y las Comunicaciones.

Tomcat: Servidor web con soporte de servlets (Jakarta Tomcat o Apache Tomcat).

UCI: Universidad de las Ciencias Informáticas.

UJC: Unión de Jóvenes Comunista.

UML: Lenguaje Unificado de Modelado.

VBScript: Lenguaje Interpretado por el Windows Scripting Host de Microsoft (Visual Basic Script Edition).

XP: Programación Extrema.