

**Universidad de las Ciencias Informáticas
Facultad 9**



SISTEMA DE CONTROL PARA LA GESTIÓN DE LA INFORMACIÓN DEL PROFESOR EN LA FACULTAD 9

Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas



Autores: Yuliet Núñez Agüero

Diana Pupo Hidalgo

**Tutor: Dr. José Ortiz Rojas
Co-Tutor: Enrique Pérez Rodríguez
Co-Tutor: Yuniel E. Proenza Arias**

**Ciudad de La Habana, Julio de 2007
Año 49 de la Revolución**



"El hombre ignorante no ha empezado a ser hombre".

José Martí.

***A nuestros padres,** que con su dedicación, paciencia y confianza han hecho posible nuestros sueños de superarnos profesionalmente, gracias por ser nuestros principales guías y por todo el amor que nos han dado.*

AGRADECIMIENTOS



Quisiéramos comenzar agradeciendo ante todo a nuestros padres que hicieron posible que pudiésemos terminar satisfactoriamente nuestros estudios, que estuvieron con nosotros en todos los momentos incondicionalmente, por su paciencia y dedicación. Gracias por su confianza y ayuda. Los queremos mucho.

A nuestros familiares que aunque desde lejos siempre han estado al pendiente de nosotros.

A nuestro tutor Ortiz, por su ejemplo y apoyo en nuestra tesis.

A Yoel Blanco, Yamek Hernández, Yismel Fonseca, Jorge Darías, Taimí García por ayudarnos a lograr lo mejor de este trabajo de diploma .Gracias

A nuestros amigos que nos han brindado todo el apoyo y ayuda que ha estado a su alcance.

A los compañeros de estos cinco años, por ser parte de nosotros, por su comprensión y ayuda.

A todos aquellos que de una manera u otra hicieron posible la realización de este trabajo.

A la Universidad de las Ciencias Informáticas por darnos la posibilidad de ser parte de ella y de esta tropa del futuro.

A Yuniel Proenza por todas sus atenciones, paciencia para con nosotras. Gracias

A Serguey, por todo este tiempo juntos y por regalarme tantos momentos que no voy a olvidar. Diana

DECLARACIÓN DE AUTORÍA



Declaramos que somos las únicas autoras de este trabajo y autorizamos a la Facultad 9 de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Autores: Diana Pupo Hidalgo

Yuliet Núñez Agüero

Tutor: José Ortiz Rojas

DATOS DE CONTACTO



Síntesis del Tutor Dr. José Ortiz Rojas

Categoría docente: Profesor Asistente

Categoría científica: Doctor en Ciencias Agrícolas

Años de graduado: 20

e_mail: jortiz@uci.cu

Síntesis del Cotutor Ing. Enrique Pérez Rodríguez

Profesión: Ingeniero Informático

Categoría docente: Instructor recién graduado

Años de graduado: 2

e_mail: enriquepr@uci.cu

Síntesis del Cotutor Ing. Yuniel Eliades Proenza Arias

Profesión: Ingeniero Informático

Categoría docente: Instructor recién graduado

Años de graduado: 1

e_mail: yproenza@uci.cu

RESUMEN



La Universidad de las Ciencias Informáticas se encuentra inmersa en un proceso de informatización. La necesidad de sustituir la forma manual o semi automatizada del trabajo de los directivos de la facultad 9, así como mantener un control sobre los profesores para la toma de decisiones importantes que influyen de manera decisiva en el buen funcionamiento de la facultad, ha llevado a concebir este trabajo de diploma que plantea como solución a la situación problemática encontrada un Sistema de Control para la Gestión de la Información del Profesor en la Facultad 9. La aplicación tiene como objetivos crear una herramienta que facilite el trabajo de los directivos de la facultad, y almacene información de cada profesor de manera organizada y centralizada y como consecuencia haga esta labor más rápida y eficiente.

Para lograr los objetivos planteados se explica conceptos asociados al dominio del problema y el objeto de estudio del trabajo. La explotación de un sistema automatizado como el propuesto en este trabajo, permitirá, a todos los involucrados en la gestión de los profesores, realizar un seguimiento del desempeño de los profesores y de esta forma contar con información fiable para tomar decisiones acertadas en función de la facultad. La aplicación está implementada sobre la plataforma Microsoft .Net, en el lenguaje C#, a través del Visual Studio, además se tuvo en cuenta todos los pasos y métodos que nos ofrece el Proceso Unificado de Desarrollo del Software.

PALABRAS CLAVES



- ✓ Aplicación Web
- ✓ Gestión de Recursos Humanos
- ✓ Plataforma .NET
- ✓ Microsoft Visual Studio
- ✓ Bases de Datos

ABSTRACT



The University of the Computer Sciences is involved in an informatization process. The necessity of replacing the manual or semi automatic style of work of the directive of the 9 Faculty , as well as to maintain a control in the professors for the taking of important decisions that have an important influence in the good operation of the Faculty, has influenced in the decision of doing this diploma work that outlines as solution to the problematic situation a System of Control for the Management of the Professor's Information at the Faculty 9. The objectives of the application are the creation of a tool that facilitates the work of the directive of the Faculty, and store each professor's information in an organized and centralized way and like that this process will be faster and more efficient.

To achieve the outlined objectives are explained concepts associated to the domine of the problem and the aim of the work. The exploitation of an automatic system as the one proposed in this work, will allow, to all those involved in the management of the professors, to fallow the acting of the professors and like this to have reliable information to take right decisions for the Faculty. The application is implemented on the platform Microsoft. Net, in the language C #, through the Visual Studio, besides, all the steps and methods that offer us the Unified Process of Development of the Software were fallowed.

ÍNDICE DE FIGURAS

FIGURA 1 ESTRUCTURA DIRECTIVA DE LA FACULTAD	14
FIGURA 2 MODELO DE PROGRAMACIÓN DE TRES CAPAS	17
FIGURA 3 VOCABULARIO DE UML	26
FIGURA 4 DIAGRAMA DE CASOS DE USO DEL NEGOCIO	31
FIGURA 5 DIAGRAMA DE ACTIVIDADES CASO DE USO: REGISTRAR DATOS PROFESOR.....	35
FIGURA 6 DIAGRAMA DE ACTIVIDADES CASO DE USO: CONTROLAR ASISTENCIA ACTIVIDADES	36
FIGURA 7 DIAGRAMA DE ACTIVIDADES CASO DE USO: GESTIONAR ACTIVIDADES	37
FIGURA 8 DIAGRAMA DE ACTIVIDADES CASO DE USO: OBTENER DATOS PROFESOR	38
FIGURA 9 DIAGRAMA DE OBJETOS	39
FIGURA 10 RELACIÓN ENTRE LOS PAQUETES DEL SISTEMA.....	43
FIGURA 11 DIAGRAMA DE CASOS DE USO DEL SISTEMA PARA PAQUETE AUTENTICACIÓN.....	43
FIGURA 12 DIAGRAMA DE CASOS DE USO DEL SISTEMA PARA PAQUETE ADMINISTRACIÓN	44
FIGURA 13 DIAGRAMA DE CASOS DE USO DEL SISTEMA PARA PAQUETE PROCESAR INFORMACIÓN.....	44
FIGURA 14 DIAGRAMA DE CLASES DE DISEÑO CASO DE USO AUTENTICAR USUARIO	57
FIGURA 15 DIAGRAMA DE CLASES DE DISEÑO CASO DE USO ELIMINAR PROFESOR.....	58
FIGURA 16 DIAGRAMA DE CLASES DE DISEÑO CASO DE USO GESTIONAR ASISTENCIA ACTIVIDADES	59
FIGURA 17 DIAGRAMA DE CLASES DE DISEÑO CASO DE USO GESTIONAR PROFESOR	60
FIGURA 18 DIAGRAMA DE CLASES DE DISEÑO CASO DE USO GESTIONAR ACTIVIDAD	61
FIGURA 19 DIAGRAMA DE CLASES DE DISEÑO CASO DE USO BUSCAR INFORMACIÓN	62
FIGURA 20 DIAGRAMA DE CLASES DE DISEÑO CASO DE USO ADMINISTRAR USUARIO.....	63
FIGURA 21 DIAGRAMA DE CLASES PERSISTENTES.....	65
FIGURA 22 MODELO DE DATOS	66
FIGURA 23 DIAGRAMA DE DESPLIEGUE.....	67
FIGURA 24 DIAGRAMA DE COMPONENTES	68
FIGURA 25 COMPLEJIDAD CICLOMÁTICA	69
FIGURA 26 PRUEBA DE CAJA NEGRA	70

ÍNDICE DE TABLAS



TABLA 1 ACTORES DEL NEGOCIO	29
TABLA 2 TRABAJADORES DEL NEGOCIO	30
TABLA 3 DESCRIPCIÓN TEXTUAL DEL CASOS DE USO REGISTRAR DATOS PROFESOR	32
TABLA 4 DESCRIPCIÓN TEXTUAL DEL CASOS DE USO CONTROLAR ASISTENCIA ACTIVIDADES.....	33
TABLA 5 DESCRIPCIÓN TEXTUAL DEL CASOS DE USO GESTIONAR ACTIVIDADES	33
TABLA 6 DESCRIPCIÓN TEXTUAL DEL CASOS DE USO OBTENER DATOS PROFESOR	34
TABLA 7 DESCRIPCIÓN DE LOS ACTORES DEL SISTEMA	42
TABLA 8 DESCRIPCIONES DEL CASO DE USO DEL SISTEMA AUTENTICAR PROFESOR	45
TABLA 9 DESCRIPCIONES DEL CASO DE USO DEL SISTEMA GESTIONAR PROFESOR	47
TABLA 10 DESCRIPCIONES DEL CASO DE USO DEL SISTEMA ELIMINAR PROFESOR	48
TABLA 11 DESCRIPCIONES DEL CASO DE USO DEL SISTEMA GESTIONAR ASISTENCIA ACTIVIDADES.....	50
TABLA 12 DESCRIPCIONES DEL CASO DE USO DEL SISTEMA MOSTRAR INFORMACIÓN	52
TABLA 13 DESCRIPCIONES DEL CASO DE USO DEL SISTEMA GESTIONAR ACTIVIDAD	54
TABLA 14 DESCRIPCIONES DEL CASO DE USO DEL SISTEMA GESTIONAR USUARIO.....	56
TABLA 15 PRUEBA DE CAJA NEGRA CASO DE PRUEBA 1	71
TABLA 16 PRUEBA DE CAJA NEGRA CASO DE PRUEBA 2	71
TABLA 17 PRUEBA DE CAJA NEGRA CASO DE PRUEBA 3	71
TABLA 18 PRUEBA DE CAJA NEGRA CASO DE PRUEBA 4	72
TABLA 19 FACTOR DE PESO DE LOS ACTORES SIN AJUSTAR.....	75
TABLA 20 FACTOR DE PESO DE LOS CASOS DE USO SIN AJUSTAR	75
TABLA 21 FACTOR DE COMPLEJIDAD TÉCNICA	76
TABLA 22 FACTOR DE AMBIENTE	77
TABLA 23 RELACIÓN ACTIVIDAD –PORCENTAJE	78

ÍNDICE

INTRODUCCIÓN	5
CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA	10
INTRODUCCIÓN.....	10
1.1 ¿QUÉ ES UN SISTEMA DE CONTROL DE GESTIÓN?	10
1.2 NECESIDAD DE UN SISTEMA DE GESTIÓN DE RECURSOS HUMANOS	11
1.2.1 Alcance de un Sistema de Gestión de Recursos Humanos.....	12
1.3 OBJETO DE ESTUDIO.....	13
1.3.1 Sistemas de Gestión de Información de Profesores.	13
1.3.2 El proceso de Gestión de Información de Profesores en la UCI.	13
1.4 ANÁLISIS DE OTRAS SOLUCIONES EXISTENTES	13
1.5 ¿QUE ES UNA APLICACIÓN WEB?	15
1.6 ARQUITECTURA.....	16
1.6.1 Plataforma .NET.....	16
1.6.2 Arquitectura de tres capas.....	16
1.6.3 Arquitectura Cliente-Servidor.....	18
1.7 LENGUAJE DE PROGRAMACIÓN C#.....	20
1.8 MICROSOFT VISUAL STUDIO.NET	21
1.9 ASP .NET.....	22
1.10 PORQUE EL USO DE BASES DE DATOS.....	24
1.10.1 Porque SQL Server.....	25
1.10.2 Embarcadero ERStudio.....	26
1.11 PORQUE EL USO DE UNIFIED MODELING LANGUAGE (UML).....	26
1.12 EL PROCESO UNIFICADO DE DESARROLLO DE SOFTWARE (RUP)	27
1.13 GESTOR DE REFERENCIAS BIBLIOGRÁFICAS: ENDNOTE.....	28
CONCLUSIONES.....	28
CAPÍTULO 2. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA	29
INTRODUCCIÓN.....	29
2.1 MODELO DE NEGOCIO	29
2.1.1 Actores y trabajadores del negocio.....	29
2.1.2 Procesos de negocio.....	30
2.1.3 Diagrama de Casos de Uso del Negocio.....	31
2.1.4 Descripción textual de los Casos de Uso de Negocio.....	32
2.1.5 Diagrama de actividades.....	35
2.1.6 Modelo de Objetos	38
2.2 REQUERIMIENTOS.....	39
2.2.1 Requerimientos Funcionales.....	40
2.2.2 Requerimientos No Funcionales.....	40
2.3 DESCRIPCIÓN DEL SISTEMA PROPUESTO.....	42
2.3.1 Descripción de los actores.....	42

2.3.2 Paquetes con sus relaciones	43
2.3.3 Diagramas por paquetes	43
Paquete Autenticación	43
Paquete Administración	44
Paquete Procesar Información	44
2.3.4 Descripciones de los Casos de Uso del Sistema	45
CONCLUSIONES	56
CAPÍTULO 3. CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA.....	57
INTRODUCCIÓN.....	57
3.1 DIAGRAMAS DE CLASES.....	57
3.2 PRINCIPIOS DE DISEÑO	64
3.2.1 Estándares de la interfaz de la aplicación.....	64
3.3 DISEÑO DE LA BASE DE DATOS	65
3.2.1 Diagrama de clases persistentes.....	65
3.2.2 Modelo de datos.....	66
3.4 GENERALIDADES DE LA IMPLEMENTACIÓN.....	67
3.4.1 Modelo de Despliegue.....	67
3.4.2 Modelo de Implementación.....	68
3.5 PRUEBA DEL SISTEMA PROPUESTO	68
CONCLUSIONES.....	72
CAPITULO 4. ESTUDIO DE FACTIBILIDAD	73
INTRODUCCIÓN.....	73
4.1 PLANIFICACIÓN	73
4.3 BENEFICIOS TANGIBLES E INTANGIBLES	79
4.3.1 Beneficios tangibles.....	79
4.3.2 Beneficios Intangibles.....	79
4.4 ANÁLISIS DE COSTOS Y BENEFICIOS.....	80
CONCLUSIONES.....	80
CONCLUSIONES	81
RECOMENDACIONES.....	82
REFERENCIAS BIBLIOGRÁFICAS.....	83
BIBLIOGRAFÍA.....	85
ANEXOS	86
GLOSARIO TÉRMINOS	103

INTRODUCCIÓN



La informática es una rama que estudia el tratamiento de la información mediante el uso de máquinas automáticas, con el nacimiento de la misma en el mundo se incrementó de manera acelerada la búsqueda de conocimientos y métodos para facilitar la realización de las actividades humanas. En Cuba esta ciencia ha logrado estar al alcance de toda la población, en aras de mejorar la calidad de la educación y de la vida en general.

"El país debe encaminarse resueltamente a la modernización informática mediante un programa integral que involucre a las organizaciones que deben proveer los recursos materiales, financieros e intelectuales y a las entidades económicas, políticas y sociales que deben traducirlos en más y mejores productos y servicios. La industria de los servicios informáticos deberá asegurar la modernidad de su base técnica y organizativa, y la elevación constante del nivel científico-técnico de sus especialistas con vistas a garantizar esos propósitos."¹

Esta industria en Cuba ha ido en ascenso a través de los años y ahora es que está logrando alcanzar una mejor visión en la producción de software, logrando adentrarse en las más disímiles esferas de la economía y la sociedad cubana con el objetivo de alcanzar mayor eficiencia. Hoy día existe un gran número de empresas e instituciones que se encuentran envueltas en un gran flujo de datos procesados de forma manual y de cierta forma desorganizada. Por ello se realizan herramientas informáticas para la gestión de todos estos datos entre ellos los de recursos humanos, debido a su importancia, pues las mismas permiten agilizar el trabajo de aquellos que estén vinculados a esta tarea, además de poder gestionar gran cantidad de información logrando con ello que la toma de decisiones sea de forma rápida y concisa.

En la Universidad de las Ciencias Informáticas creada bajo el desarrollo de la informática en el país, se impulsa una estrategia para informatizar las diferentes áreas administrativas y de servicios. Luego de

¹ Resolución Económica V Congreso del PCC, (1997) Editora Política. La Habana, p. 30.

INTRODUCCION

profundas investigaciones tomando como base la facultad 9 dirigidas al conocimiento de su situación en cuanto a los recursos humanos. Se ha definido la necesidad de sustituir la forma manual o semi automatizada del trabajo de sus directivos, quienes en la actualidad realizan esta actividad apoyados en herramientas propias del Microsoft Office, tales como Word, Excel y en algunos casos PowerPoint, lo que provoca un difícil acceso de todos los datos de un profesor de manera organizada y centralizada, pues diariamente, la facultad recibe un gran número de informaciones del profesor, estas se reciben y se almacenan para ser utilizadas cuando sean necesarias.

Para solucionar esta **situación problemática**, se lleva a cabo un proceso de informatización o de transformación con el objetivo de perfeccionar el trabajo de todos los principales directivos de la facultad, y así lograr concentrar todos los datos personales de los profesores, la asistencia y la gestión de las actividades, que en la actualidad son gestionados por el asistente de control y los directivos de la facultad.

A partir de todo lo anterior, el **problema a resolver** es el siguiente: inexistencia de un proceso automatizado para gestionar la información referente a los profesores de la facultad 9.

Actualidad y necesidad del trabajo.

En la Universidad de Ciencias Informáticas (UCI) la gestión de información de profesores tiene antecedentes en el Módulo de Profesores de Akademos, el cual permite a los jefes de departamentos, decanos de facultades, entre otros, mantener un control de los profesores para la toma de decisiones importantes como directivos de la Universidad. Sin embargo este tema de gestión de profesores es completamente nuevo en la facultad, para ello se efectuó varias entrevistas a diferentes miembros de la facultad, especialmente a directivos, con el objetivo de conocer sus propuestas de cómo resolver este problema.

En la actualidad en la facultad 9 no existe un sistema informático que controle la información del profesor. Esto trae como consecuencia que el trabajo de los jefes de departamento, decano, vice decanos u otros directivos se vea limitado en cuanto a la eficiencia, lo que provoca insatisfacción a aquellos que puedan estar involucrados, molestias innecesarias, pérdida de tiempo en ocasiones y aumento de la dificultad a la hora de realizar el trabajo, debido a los motivos expuestos, es que surge la necesidad de realizar un Sistema de Control para la Gestión de la Información del Profesor en la Facultad 9.

Lo que estimuló a que este trabajo tenga como **objetos de estudio**, todo lo referente al Sistema de Control para la Gestión de la Información del Profesor en la Facultad 9, y como **campo de acción** de la investigación llevada a cabo para el desarrollo del sistema que se propone, se encuentra un sistema de gestión de información del profesor en la facultad 9 a través de la tecnología .Net, para facilitar el trabajo de los dirigentes de la facultad y lograr un mejor manejo de esta información.

En correspondencia con lo anterior, este trabajo tiene como **objetivo general**:

Implementar un sistema automatizado que facilite el control de toda la información del profesor de la facultad 9.

Para la realización de este trabajo se planteó la siguiente **Idea a defender**:

Un Sistema de Control para la Gestión de la Información del Profesor en la Facultad 9 que permita automatizar la forma manual con que se realiza la gestión del profesor y así lograr una mayor organización en la labor de los directivos de la facultad y con ello un mejor funcionamiento de la misma en su conjunto.

Después de analizar la situación problemática, el objetivo general y las ideas a defender se concretó las siguientes **tareas** para el cumplimiento de los objetivos planteados.

- ✓ Comprobación del problema.
- ✓ Estudio de sistemas de gestión de información.
- ✓ Análisis de los sistemas de gestión existentes.
- ✓ Realización de un estudio de las tecnologías a utilizar.
- ✓ Modelación del sistema, utilizando las herramientas y artefactos que nos brindan UML (Unified Modeling Language) y RUP (Rational Unified Process, o Proceso Unificado de Desarrollo de Software).
- ✓ Implementación de la aplicación.
- ✓ Estudio de la factibilidad de la aplicación.
- ✓ Validación de la propuesta.

Como métodos de investigación científica se utilizaron:

Métodos Teóricos:

- ✓ Analítico – sintético: analizar teóricamente el proceso de la facultad 9 y sintetizarlo.
- ✓ Inductivo – deductivo: Se analiza de lo general a lo particular el proceso.
- ✓ Modelación: crear modelos a través de la metodología RUP

Métodos Empíricos:

- ✓ Entrevista: Se realizan entrevistas a los profesores de la Facultad, con el objetivo de precisar el problema a resolver.
- ✓ Observación: Se realizaron visitas a la Facultad para observar los procesos, lográndose entender los procesos de gestión de información en la misma.

Resultados esperados:

- La facultad pueda tener todos los procesos relacionados con la gestión de los profesores completamente automatizados.
- La aplicación presente una fuerte seguridad y fiabilidad.
- La Base de Datos guarde de manera segura todos los datos del sistema.

Estructuración del contenido.

El contenido de este trabajo de diploma está estructurado en cuatro capítulos.

El primero, ***Fundamentación Teórica*** muestra aspectos generales de un Sistema de Control de Gestión, de un Sistema de Gestión de Recursos Humanos, en que consisten los Sistemas de Gestión de Información de Profesores, como se realiza este proceso en la UCI, y como base la Facultad 9 y el motivo de su informatización. Del mismo modo se tratan otros aspectos elementales relacionados con el estudio de las tendencias y tecnologías actuales y cuales usar en el desarrollo de la aplicación.

En el segundo capítulo, ***Presentación de la solución propuesta*** se muestra la modelación del negocio realizada, se recogen los requerimientos que debe cumplir la aplicación en forma de casos de uso, así como la modelación del sistema a desarrollar.

INTRODUCCION

El tercer capítulo, **Construcción de la solución propuesta**, se modelan un grupo de artefactos de los flujos de trabajo de Análisis y Diseño, Implementación y Prueba. Además aborda un conjunto de principios de diseño como estándares de la interfaz de la aplicación.

En el cuarto capítulo **Estudio de Factibilidad** se hace un estudio de la factibilidad de la aplicación, el costo que tiene la misma, así como sus beneficios tangibles e intangibles.

Capítulo 1. Fundamentación Teórica



Introducción

En este capítulo se plantean los principales elementos teóricos que sustentan el problema a resolver del trabajo. El mismo tiene como finalidad ofrecer un análisis del marco teórico del proyecto, en el que se ofrece un estudio sobre los Sistemas de Control de Gestión. Se explica en que consiste la Gestión de Información de Profesores y como se realiza este proceso en una facultad de la UCI, tomando como base el trabajo de dirección y gestión de los recursos humanos de la facultad 9.

Se aborda sobre las principales herramientas y tecnologías que se utiliza en el análisis, diseño e implementación de la aplicación Web. La plataforma y el lenguaje de programación en el que se trabaja. Se describe la arquitectura y el gestor de base de dato que se utiliza. De esta forma se formulará una solución informática para resolver la situación problemática planteada.

1.1 ¿Qué es un Sistema de Control de Gestión?

Un Sistema de Control de Gestión, es un conjunto de procedimientos que representa un modelo organizativo concreto para realizar la planificación y control de las actividades que se llevan a cabo en la organización, el cual queda determinado por un conjunto de actividades y sus interrelaciones, y un sistema informativo [1]. El sistema de control de gestión esta destinado a ayudar a los distintos niveles de decisión a coordinar las acciones, a fin de alcanzar los objetivos de mantenimiento, desempeño y evolución, fijados a distintos plazos por la empresa o institución implicada.

En la actualidad, con el objetivo de acceder a los recursos necesarios, reducir gastos y costos, aumentar la calidad de los productos y servicios, se ha modificado la forma de actuar e interactuar de las organizaciones. El Control de Gestión es: "... el conjunto de mecanismos que puede utilizar la dirección

FUNDAMENTACIÓN TEÓRICA

que permiten aumentar la probabilidad de que el comportamiento de las personas que forman parte de la organización sea coherente con los objetivos de ésta."²

El proceso de control de gestión, ajustado a las necesidades actuales de gestión de información pudiera plantearse en cinco puntos: [9]

1. Conjunto de indicadores de control que permitan orientar y evaluar posteriormente el aporte de cada departamento a las variables claves de la organización.
2. Modelo predicativo que permita estimar (a priori) el resultado de la actividad que se espera que realice cada responsable y/o unidad.
3. Objetivos ligados a indicadores y a la estrategia de la organización.
4. Información sobre el comportamiento y resultado de la actuación de los diferentes departamentos.
5. Evaluación del comportamiento y del resultado de cada persona y/o departamento que permita la toma de decisiones correctivas.

Los sistemas de control deben cumplir con una serie de requisitos para su funcionamiento eficiente: [3]

- Ser entendibles.
- Seguir la forma de organización.
- Rápidos.
- Flexibles.
- Económicos.

Luego de conocer en que consiste un Sistema de Control de Gestión debemos pasar a conocer la necesidad de un Sistema de Gestión de Recursos Humanos.

1.2 Necesidad de un Sistema de Gestión de Recursos Humanos

Probablemente el sistema de información de RR.HH. es el menos automatizado de todos los sistemas de una empresa u organización. Sin embargo, la administración de RR.HH tiene responsabilidad de equilibrar las necesidades del personal con los derechos laborales de manera que se alcancen los objetivos de la organización. Los métodos tradicionales manuales para mantener la información del personal no son

² Joan Ma. Amat (1992, p. 35.)

FUNDAMENTACIÓN TEÓRICA

viabiles en una organización relativamente grande debido a que generar cualquier tipo de información implica horas de trabajos.

En las estrategias sobre Gestión de Recursos Humanos de países en vía de desarrollo es imprescindible, tener muy presente la consideración siguiente, expresada en el referido informe del PNUD: “En el siglo XXI el rápido adelanto de la tecnología y las comunicaciones abrirá la posibilidad de “saltar” varios decenios, pero sólo si los países pobres logran dominar los nuevos conocimientos y competir”. Otro factor importante cuya consideración define el nivel de efectividad y eficiencia en la Gestión de Recursos Humanos es el desarrollo de la informática y las comunicaciones. El aumento de la productividad del trabajo y de la satisfacción laboral, vinculados a las condiciones del trabajo, son objetivos inmediatos fundamentales de la Gestión de Recursos Humanos. Una solución simple que facilite todas estas gestiones es un sistema informático de Gestión de Recursos Humanos que proporcione mejores alternativas, y que tengan como objetivo servir de apoyo a la gestión de los mismos, y pueda utilizarse en el proceso de planificación. [7]

1.2.1 Alcance de un Sistema de Gestión de Recursos Humanos

Un Sistema de Gestión de Recursos Humanos facilita la gestión de todos los datos relacionados con la disponibilidad de los recursos para la definición de categorías, puestos, áreas de trabajos, horarios de trabajo, asistencias del personal, contratación y movimientos del personal, datos necesarios para realizar los controles y verificaciones de todos los datos relacionados al personal.

La implantación de un Sistema de Control para la Gestión de la Información del Profesor en la facultad 9 se convierte en un beneficio a la facultad y por ende a la sociedad cubana, este permite un mejor desarrollo administrativo de la facultad y la satisfacción de todos los vinculados con esta tarea, así como los propios profesores que tendrán a la mano información detallada sobre cualquier profesor o actividad desarrollada y la asistencia a las mismas.

Con este sistema se espera que el profesor pueda registrar digitalmente sus datos personales eliminando el uso de papeles archivados que no están al alcance de cualquier profesor o directivo de la facultad en cualquier momento que pueda necesitarlos, la obtención de reportes que en estos momentos son dificultosos por el número de implicados, como puede ser docentes por departamentos, años de

FUNDAMENTACIÓN TEÓRICA

experiencia, cargos que ocupa x profesor, personas por: edad, sexo, estado civil, antigüedad, nivel académico etc. Debido a la importancia del profesor en la facultad, sus conocimientos, estudios y las diferentes actividades que realiza, se hace necesario el control del mismo a través del sistema que se plantea.

1.3 Objeto de estudio

1.3.1 Sistemas de Gestión de Información de Profesores.

La gestión de Información de Profesores constituye un elemento de gran importancia para el buen funcionamiento de cualquier centro educacional. Los sistemas de gestión de Información de Profesores tienen como tarea fundamental almacenar y procesar toda la información referente a los profesores de un centro de estudios. Este sistema debe recoger todo lo relacionado con los profesores como pueden ser sus datos personales, su currículum, la asistencia las diferentes actividades, así como las asignaturas que imparte. Estos sistemas constituyen una poderosa herramienta de trabajo que le permiten a cualquier centro de estudios hacer eficientes sus procesos, optimizando así sus recursos.

1.3.2 El proceso de Gestión de Información de Profesores en la UCI.

La Universidad de las Ciencias Informáticas (UCI), constituye un centro que está dedicado a la formación de un número creciente de nuevos profesionales, lo que trae aparejado el incremento del número de profesores, lo que demuestra la necesidad de un sistema automatizado para la gestión de los profesores en la universidad que garantice la eficiencia de la gestión de la información, en estos momentos se cuenta con Akademos como una herramienta informática para realizar esta gestión. En una facultad es de vital importancia para su buen funcionamiento la eficacia de la gestión de información de profesores, de ahí que surja la necesidad de emplear sistemas informáticos que automaticen todo este proceso y garantice la eficiencia de la gestión de profesores en la facultad 9.

1.4 Análisis de otras soluciones existentes

En nuestra Universidad se cuenta con un Sistema de Gestión Académica. Akademos que consta de varios módulos. [2]

1. Módulo de plan de estudio: Definición de los planes de estudio.

FUNDAMENTACIÓN TEÓRICA

2. Módulo de matrícula: Gestión de estudiantes y estructuras.
3. Módulo de estudiante: Herramienta que mantiene informado a los estudiantes sobre su desempeño académico.
4. Módulo de registro del profesor: Control del desarrollo del proceso docente, notas y asistencia.
5. Módulo de Planificación: Definición de los planes de estudio. Asignaturas, disciplinas, perfiles.
6. Módulo de reportes: Diseño y generación de reportes personalizados.
7. Módulo de Gestión de Profesores: Gestión del claustro del centro.

Este último permite a los jefes de departamento, decanos docentes de facultades, entre otros, mantener un control de la plantilla de profesores y realizar un balance de la carga docente de estos así como la gestión de toda la información referente al mismo, de forma tal que se puedan utilizar de forma óptima a los profesores. También permite que otros directivos del centro puedan acceder en un momento deseado a parte de la información relacionada con los profesores de la Universidad, pues el alto número de profesores con el que cuenta la UCI actualmente hace engorroso el proceso de gestión y planificación de los mismos.

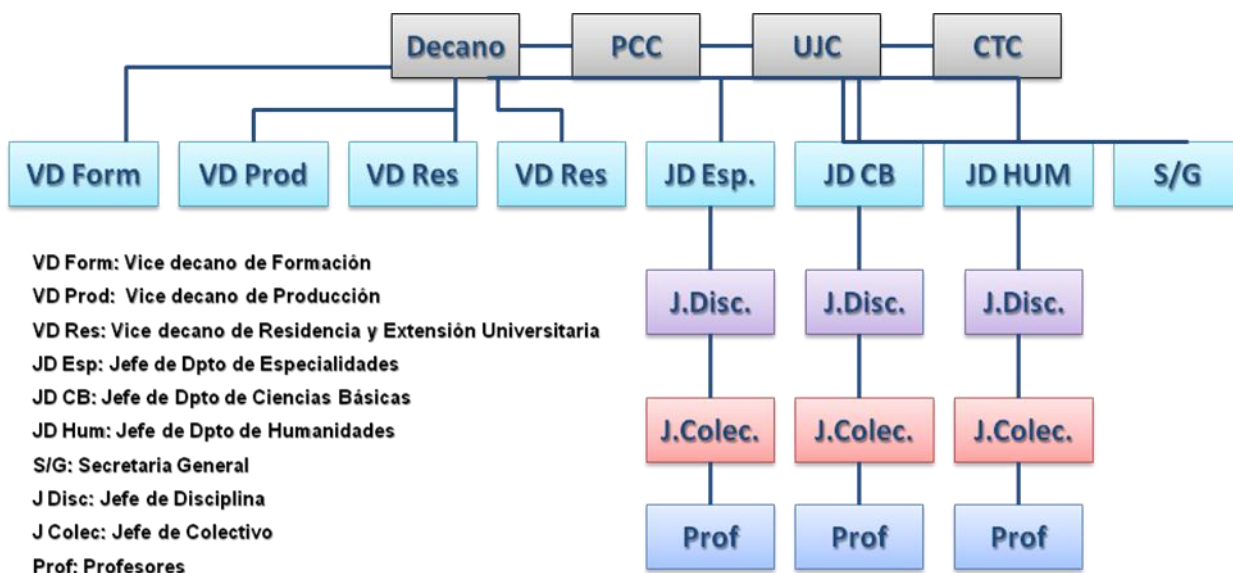


Figura 1 Estructura Directiva de la Facultad

FUNDAMENTACIÓN TEÓRICA

1.5 ¿Que es una aplicación Web?

En informática, las **aplicaciones** son los programas con los cuales el usuario final interactúa a través de una interfaz y que realizan tareas útiles para éste. [21]

Una Aplicación Web es aquella que los usuarios usan accediendo a un servidor Web a través de Internet o de una intranet. Las aplicaciones Web son cada día más populares debido a la practicidad del navegador Web como cliente ligero y la capacidad para actualizar y mantener la aplicación sin distribuir e instalar software en cada estación de trabajo. [8]

El sistema de información de profesores para la facultad 9 necesita ser de fácil uso y actualización pues el directivo debe tener acceso a la información en cualquier momento por esto se decidió implementar una aplicación Web que consta de las siguientes ventajas: [8]

1. **Compatibilidad multiplataforma:** Las aplicaciones Web tienen un camino mucho más sencillo para la compatibilidad multiplataforma que las aplicaciones de software tradicionales. Tecnologías como Java, Flash, ASP, PHP permiten un desarrollo efectivo de programas soportando todos los sistemas operativos principales.
2. **Actualización.:** Las aplicaciones basadas en Web se mantienen siempre actualizadas sin demandar la atención del usuario para descargar e instalar actualizaciones.
3. **Inmediatez de acceso:** Las aplicaciones basadas en Web no necesitan ser descargadas, instaladas y configuradas. Usted accede a su cuenta online y la aplicación se encuentra preparada para trabajar sin importar cuál es su configuración o su hardware.
4. **Precio:** Las aplicaciones basadas en Web no requieren la infraestructura de distribución, soporte técnico y permite mantener su información en la Web en vez de hacerlo en papel. Esto permite que las aplicaciones Web sean muchísimas más económicas.
5. **La posibilidad de ofrecer una mayor cantidad de información y presentarla de varias formas como:** texto, imágenes, animaciones, videos, etc. Además de la posibilidad de interactuar con otros sistemas y/o aplicaciones ligadas a ella para automatizar algunos procesos, además que permite acceder a una base de dato de información.
6. **Por otra parte cuenta con la constante exposición de su información (las 24 hrs. los 365 días del año).**

FUNDAMENTACIÓN TEÓRICA

1.6 Arquitectura

1.6.1 Plataforma .NET

El objetivo de la plataforma .NET es simplificar el desarrollo de aplicaciones Web. Esta plataforma soporta los estándares sobre los cuales se basan los servicios Web. La plataforma .NET es una capa de software que se coloca entre el Sistema Operativo (SO) y el programador y que abstrae los detalles internos del SO. Se optó por utilizar esta plataforma para la construcción de la aplicación debido a que la misma presenta las siguientes características: [4]

- ✓ **Portabilidad:** Debido a la abstracción del programador respecto al SO, una aplicación .NET puede ser ejecutada en cualquier SO de cualquier máquina que disponga de una versión de la plataforma. En estos momentos la plataforma .NET tan solo está disponible para la familia Windows aunque se está desarrollando una versión para Linux de Corel.
- ✓ **Multilinguaje:** Cualquier lenguaje de programación puede adaptarse a la plataforma .NET y ejecutarse en ella.
- ✓ **Interoperabilidad:** La interoperabilidad entre diferentes trozos de código escritos en diferentes lenguajes es total.

1.6.2 Arquitectura de tres capas.

El modelo actual de desarrollo ha demostrado que organizar los elementos de las aplicaciones en capas independientes puede lograr una mayor eficiencia durante el tiempo de desarrollo y mantenimiento. La programación en múltiples capas es la técnica más efectiva para aplicaciones empresariales, dividir los componentes de la aplicación en capas implica una fácil administración y rapidez en entornos cliente-servidor. Esta arquitectura consiste en dividir los componentes primarios de una aplicación, programarlos por separado y después unirlos, ya sea en tiempo de diseño o de ejecución.

Una aplicación Web está comúnmente estructurada como una aplicación de tres capas. La primera capa es el navegador Web, en la capa del medio se encuentran las tecnología Web dinámica (ejemplo: CGI, PHP, Java Servlets o ASP), y en la última capa encontramos la base de datos.

FUNDAMENTACIÓN TEÓRICA

El navegador Web manda peticiones a la capa media, que la entrega valiéndose de consultas y actualizaciones a la base de datos generando una interfaz de usuario. [4]

La arquitectura más popular es la arquitectura de tres capas. Separa los datos de una aplicación, la interfaz de usuario y la lógica en tres componentes distintos. [4]

- Interfaces de usuario (Capa de Presentación).
 - Interactuar con otros usuarios.
 - Interactuar con aplicaciones externas o servicios.
- Procesos de negocios (Capa de Negocios).
 - Cálculos u otros procesos de negocios.
 - Ejecución de reglas de negocios.
 - Validación de datos relacionados al negocio.
- Procesos de datos (Capa de Servicios de Datos).
 - Manipulación de datos.
 - Ejecución de las reglas de datos relacionales.

Modelo de programación de tres capas



Figura 2 Modelo de programación de tres capas.

FUNDAMENTACIÓN TEÓRICA

Se concluyó desarrollar una aplicación basada en una arquitectura en 3 capas debido a: [4]

- ✓ Con la arquitectura de tres capas, la interfaz del cliente no es requerida para comprender o comunicarse con el receptor de los datos. Por lo tanto, esa estructura de los datos puede ser modificada sin cambiar la interfaz del usuario en la PC.
- ✓ El código de la capa intermedia puede ser reutilizado por múltiples aplicaciones si está diseñado en formato modular.
- ✓ Esto puede reducir los esfuerzos de desarrollo y mantenimiento.
- ✓ La separación de roles en tres capas, hace más fácil reemplazar o modificar una capa sin afectar a los módulos restantes.
- ✓ Separando la aplicación de la base de datos, hace más fácil utilizar nuevas tecnologías de agrupamiento y balance de cargas.
- ✓ Separando la interfaz del usuario de la aplicación, libera de gran procesamiento a la estación de trabajo y permite que las actualizaciones de la aplicación sean centralizadas en el servidor de aplicaciones.

1.6.3 Arquitectura Cliente-Servidor.

Luego de analizar los aspectos de la aplicación se conoce que la misma debe contar con un sistema de red que presente una única interfaz al cliente para ello se decidió trabajar con la arquitectura Cliente/Servidor, que no es más que la integración distribuida de un sistema en red, con los recursos, medios y aplicaciones, que definidos modularmente en los servidores, administran, ejecutan y atienden las solicitudes de los clientes; todos interrelacionados física y lógicamente, compartiendo datos, procesos e información. Se establece así un enlace de comunicación transparente entre los elementos que conforman la estructura. [5]

Entre las principales características de la arquitectura Cliente/Servidor, se pueden destacar las siguientes: [5]

- ✓ El servidor presenta a todos sus clientes una interfaz única y bien definida.

FUNDAMENTACIÓN TEÓRICA

- ✓ 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.
- ✓ Los cambios en el servidor implican pocos o ningún cambio en el cliente.

La programación cliente-servidor es utilizada cuando se quiere realizar aplicaciones que utilicen redes y que comuniquen entre sí a varios ordenadores. Consiste en que el programa se divide en dos partes.

La parte Cliente, que reside en el equipo donde está el usuario y se encarga de la interacción con éste y la parte Servidor, que reside en un ordenador conectado a la red permanentemente y se encarga de manipular los datos. [5]

Ambas partes de la aplicación se comunican entre sí utilizando algún protocolo de red TCP/IP. La justificación de este paradigma es la minimización del tráfico de red, sobre todo para evitar ralentizaciones y economizar el ancho de banda. Esta tecnología denominada Cliente -Servidor es utilizada por todas las aplicaciones de Internet/Intranet. Un único servidor típicamente sirve a una multitud de clientes, ahorrando a cada uno de ellos el problema de tener la información instalada y almacenada localmente.

Entre las principales ventajas de este modelo se encuentran: [19]

- ✓ Con el uso de este esquema, se reducen los costos de producción de software y se disminuyen los tiempos requeridos.
- ✓ Reduce el costo del hardware requerido, llevando las aplicaciones a plataformas más baratas, aprovechando el poder de cómputo de los diferentes elementos de la red, y facilitando la interacción entre las distintas aplicaciones de la organización.
- ✓ Contribuye a una disminución de los costos de entrenamiento de personal, pues favorecen la construcción de interfaces gráficas interactivas, las cuales son más intuitivas y fáciles de usar por el usuario final.
- ✓ Facilita el suministro de información a los usuarios.
- ✓ Permite llevar más fácilmente la información a donde se necesita, contribuye a aumentar su precisión pues se puede obtener de la fuente (el servidor) y no de una copia en papel o en medio magnético.

FUNDAMENTACIÓN TEÓRICA

Aplicaciones Cliente/Servidor multicapa.

Estas aplicaciones aportan una flexibilidad adicional en su construcción cuando aumenta su complejidad. Hay partes de la lógica que residen en la computadora denominada cliente, normalmente los que se refieren a interfaz de usuario, mientras que las del negocio y de datos suelen residir en las computadoras destinadas como servidoras, que proporcionan los mecanismos necesarios para el trabajo en entornos multiusuario. Una de las características fundamentales de este modelo reside en la desconexión total entre la lógica de presentación y la lógica de los datos. [11]

1.7 Lenguaje de programación C#.

C# es un lenguaje elegante, sencillo y orientado a objetos, diseñado para ofrecer un desarrollo rápido de aplicaciones y conseguir más con menos líneas de código y menores posibilidades de error. [15]

Aunque es posible escribir código para la plataforma .NET en muchos otros lenguajes, C# es el único que ha sido diseñado específicamente para ser utilizado en ella, por lo que programar el Sistema de Información de Profesores usando C# es mucho más sencillo e intuitivo que hacerlo con cualquiera de los otros lenguajes. Se suele decir que C# es el **lenguaje nativo de .NET**.

En resumen, C# es un lenguaje de programación que toma las mejores características de lenguajes preexistentes como Visual Basic, Java o C++ y las combina en uno sólo. [3]

Características de C# [3]

- ✓ **Sencillez:** C# elimina muchos elementos que otros lenguajes incluyen y que son innecesarios en .NET.
- ✓ **Modernidad:** C# incorpora en el propio lenguaje elementos que a lo largo de los años ha ido demostrándose son muy útiles para el desarrollo de aplicaciones y que en otros lenguajes como Java o C++ hay que simular.
- ✓ **Orientación a objetos:** Como todo lenguaje de programación de propósito general actual, C# es un lenguaje orientado a objetos.

FUNDAMENTACIÓN TEÓRICA

- ✓ **Orientación a componentes:** La propia sintaxis de C# incluye elementos propios del diseño de componentes que otros lenguajes tienen que simular mediante construcciones más o menos complejas.
- ✓ **Seguridad de tipos:** C# incluye mecanismos que permiten asegurar que los accesos a tipos de datos siempre se realicen correctamente, lo que evita que se produzcan errores difíciles de detectar por acceso a memoria no perteneciente a ningún objeto.

1.8 Microsoft Visual Studio.NET

Como herramientas de desarrollo de aplicaciones .NET se encuentran Visual Studio.NET y Web Matrix. Se decidió trabajar con Visual Studio.NET pues permite diseñar, desarrollar, depurar e instalar soluciones basadas en .NET. Todos los lenguajes que son incorporados a Visual Studio pueden acceder a un conjunto común de herramientas, editores y diseñadores. Se pueden crear aplicaciones con formularios Windows o formularios Web que integran datos y lógica propia de la aplicación, así como Servicios Web. Facilita la creación y personalización visual de controles de formulario Windows y Web. Brinda facilidades para el acceso a datos y asistentes para realizar determinadas acciones. Además es una herramienta de desarrollo multilenguaje integrado (Visual Basic .NET, C++, C# y J#) lo que la hace una de las herramientas más completa alcanzadas en la actualidad. [3]

Características de Visual Studio.Net [14]

Detección de errores automáticos: Es posible ahorrar muchas horas de trabajo al utilizar Visual Studio.Net para detectar los errores antes de intentar ejecutar su aplicación. Los errores se subrayan de la misma forma que lo hacen algunos procesadores de textos mientras se escribe.

Herramientas de depuración: Visual Studio.Net mantiene sus herramientas de depuración legendarias que le permiten observar su código en acción y seguir la pista del contenido de las variables.

Diseño de página: Es posible crear una página atractiva con la facilidad de arrastrar y soltar integrada en el diseñador de formulario Web de Visual Studio. Net.

FUNDAMENTACIÓN TEÓRICA

Ventajas de Visual Studio .Net [17]

Arquitectura flexible y ágil de aplicaciones

- ✓ Creada para los servicios Web XML
- ✓ Constituye un modelo sencillo, flexible y basado en estándares para integrar, ampliar y publicar aplicaciones.

Máxima productividad del desarrollador

- ✓ Soporta varios lenguajes de programación.
- ✓ Pueden reutilizar las aplicaciones escritas en cualquier lenguaje de programación.

Operaciones mejoradas

- ✓ Aumenta el rendimiento, la escalabilidad y la fiabilidad.
- ✓ Ofrece al usuario un tiempo en actividad percibido del 100%.
- ✓ Elimina el problema de conflictos entre versiones. Las nuevas aplicaciones pueden ejecutar aplicaciones existentes.

1.9 ASP .NET.

ASP.NET es un marco de trabajo de programación que puede utilizarse en un servidor para generar eficaces aplicaciones Web. ASP.NET es un entorno compilado basado en .NET. Se pueden crear aplicaciones en cualquier lenguaje compatible con .NET, como Visual Basic .NET, C# y JScript .NET.

Ventajas de ASP.NET [18]

- ✓ **Mejor rendimiento:** ASP.NET es un código de Common Language Runtime compilado que se ejecuta en el servidor. A diferencia de sus predecesores, ASP.NET puede aprovechar las ventajas del enlace anticipado, la compilación just-in-time, la optimización nativa y los servicios de caché desde el primer momento. Esto supone un incremento espectacular del rendimiento antes de siquiera escribir una línea de código.
- ✓ **Compatibilidad con herramientas de primer nivel:** El marco de trabajo de ASP.NET se complementa con un diseñador y una caja de herramientas muy completos en el entorno integrado de programación (Integrated Development Environment, IDE) de Visual Studio. La edición

FUNDAMENTACIÓN TEÓRICA

WYSIWYG, los controles de servidor de arrastrar y colocar y la implementación automática son sólo algunas de las características que proporciona esta eficaz herramienta.

- ✓ **Eficacia y flexibilidad:** Debido a que ASP.NET se basa en Common Language Runtime, la eficacia y la flexibilidad de toda esa plataforma se encuentra disponible para los programadores de aplicaciones Web. La biblioteca de clases de .NET Framework, la Mensajería y las soluciones de Acceso a datos se encuentran accesibles desde el Web de manera uniforme. ASP.NET es también independiente del lenguaje, por lo que puede elegir el lenguaje que mejor se adapte a la aplicación o dividir la aplicación en varios lenguajes.
- ✓ **Simplicidad:** ASP.NET facilita la realización de tareas comunes, desde el sencillo envío de formularios y la autenticación del cliente hasta la implementación y la configuración de sitios. Por ejemplo, el marco de trabajo de página de ASP.NET permite generar interfaces de usuario, que separan claramente la lógica de aplicación del código de presentación, y controlar eventos en un sencillo modelo de procesamiento de formularios de tipo Visual Basic.
- ✓ **Facilidad de uso:** ASP.NET emplea un sistema de configuración jerárquico, basado en texto, que simplifica la aplicación de la configuración al entorno de servidor y las aplicaciones Web. Debido a que la información de configuración se almacena como texto sin formato, se puede aplicar la nueva configuración sin la ayuda de herramientas de administración local. Esta filosofía de "administración local cero" se extiende asimismo a la implementación de las aplicaciones ASP.NET Framework. Una aplicación ASP.NET Framework se implementa en un servidor sencillamente mediante la copia de los archivos necesarios al servidor. No se requiere el reinicio del servidor, ni siquiera para implementar o reemplazar el código compilado en ejecución.
- ✓ **Posibilidad de personalización y extensibilidad.** ASP.NET presenta una arquitectura bien diseñada que permite a los programadores insertar su código en el nivel adecuado. De hecho, es posible extender o reemplazar cualquier subcomponente del motor de tiempo de ejecución de ASP.NET con su propio componente escrito personalizado. La implementación de la autenticación personalizada o de los servicios de estado nunca ha sido más fácil.

FUNDAMENTACIÓN TEÓRICA

- ✓ **Seguridad:** Con la autenticación de Windows integrada y la configuración por aplicación, se puede tener la completa seguridad de que las aplicaciones están a salvo.

1.10 Porque el uso de Bases de Datos

En los últimos años, las bases de datos han experimentado profundos cambios y no son ya, como ocurría hace algunos años, competencia exclusiva de grandes instalaciones con sistemas de información que gestione millones de registros, tampoco el diseño de Bases de Datos esta reservado actualmente, como ocurría antaño, a unos pocos “gurús”, mezcla de sabios y magos, especialistas en las técnicas de estructuración de los datos. Por el contrario las bases de datos se han extendido alcanzando a sistemas de tipo medio y pequeño, con moderadas e incluso bajos cargos de trabajo, viéndose implicados en su diseño, administración y manejo muchos profesionales y multitud de usuarios, que reclaman de ellas flexibilidad, sencillez y eficiencia. [16]

Las BD y sus tecnologías tienen un impacto decisivo con el creciente uso de las computadoras. Las BD desempeñan un papel crucial en casi todas las áreas de aplicaciones de las computadoras, como los negocios, la ingeniería, la medicina, el derecho, la educación y la bibliotecología. [6]

Una BD es un conjunto de datos que tiene las siguientes propiedades implícitas: [6]

- ✓ Representa algún aspecto del mundo real, llamado mini mundo o universo de discurso.
- ✓ Es un conjunto de datos lógicamente coherentes, con un cierto significado inherente. Una colección aleatoria de datos no puede considerarse propiamente una BD.
- ✓ Una BD se diseña, construye y puebla con datos para propósito específico. Está dirigida a un grupo de usuarios y tiene ciertas aplicaciones preconcebidas que interesan a distintos usuarios.

Un Sistema Gestor o Manejador de Bases de Datos (SGBD) es un conjunto de programas que permite a los usuarios crear y mantener una BD, por lo tanto, el SGBD es un software de propósito general que facilita el proceso de definir, construir y manipular la BD para diversas aplicaciones. Pueden ser de propósito general o específico. [6]

Ejemplos de SGBD son Oracle y SQL Server de Microsoft. IBM DB2, Interbase.

FUNDAMENTACIÓN TEÓRICA

1.10.1 Porque SQL Server.

Se decidió elegir el SQL Server como el gestor de BD a utilizar, pues el mismo es capaz de poner a disposición de muchos usuarios grandes cantidades de datos de manera simultánea.

Entre sus características figuran: [20]

- ✓ Soporte de transacciones.
- ✓ Gran estabilidad.
- ✓ Gran seguridad.
- ✓ Escalabilidad.
- ✓ Soporta procedimientos almacenados.
- ✓ Permite trabajar en modo cliente-servidor donde la información y datos se alojan en el servidor y las terminales o clientes de la red sólo accedan a la información.
- ✓ Además permite administrar información de otros servidores de datos

Para el desarrollo de aplicaciones más complejas (tres o más capas), Microsoft SQL Server incluye interfaces de acceso para la mayoría de las plataformas de desarrollo, incluyendo .NET. [20]

La aplicación para el Sistema de Control de Información del Profesor de la Facultad 9 trabaja con el gestor de base de dato SQL pues el mismo permite:

- Definir una base de datos mediante tablas
- Almacenar información en tablas.
- Seleccionar la información que sea necesaria de la base de datos.
- Realizar cambios en la información y estructura de los datos.
- Combinar datos para conseguir la información necesaria.
- SQL es el lenguaje de comunicación entre el programa cliente y programa servidor.

FUNDAMENTACIÓN TEÓRICA

1.10.2 Embarcadero ERStudio

Esta herramienta permite construir modelos de datos mediante diagramas Entidad-Relación y generar el SQL correspondiente al modelo. Estos modelos ayudan a visualizar las estructuras de datos de modo que se pueda organizar, manejar y modelar con eficacia la complejidad de los datos, las tecnologías de la base de datos y el ambiente del despliegue, por ello se utiliza esta herramienta para el desarrollo de este sistema.

1.11 Porque el uso de Unified Modeling Language (UML).

Como soporte a la modelación de la solución propuesta se utiliza Unified Modeling Language (Lenguaje de construcción de modelos) debido a que es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra una gran cantidad de software. El UML esta compuesto por diversos elementos gráficos que se combinan para conformar diagramas, es un lenguaje que permite la modelación de sistemas con tecnología orientada a objetos. Un modelo UML describe lo que supuestamente hará un sistema, pero no dice cómo implementar dicho sistema. [12]

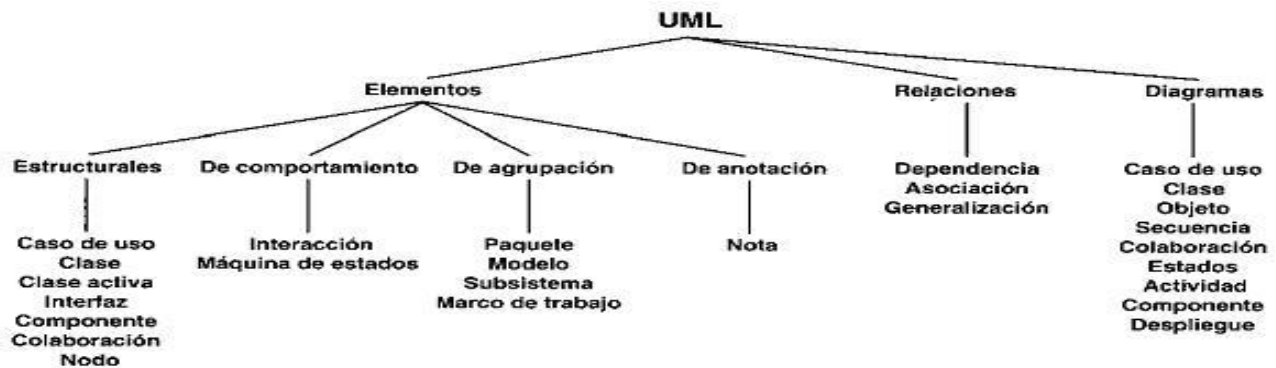


Figura 3 Vocabulario de UML

FUNDAMENTACIÓN TEÓRICA

1.12 El Proceso Unificado de Desarrollo de Software (RUP)

En un proyecto de desarrollo de software la metodología define Quién debe hacer Qué, Cuándo y Cómo debe hacerlo. Una metodología es un proceso. No existe una metodología de software universal. Las características de cada proyecto (equipo de desarrollo, recursos, etc.) exigen que el proceso sea configurable.

En la actualidad existen varias metodologías. Se concluyó utilizar como base en el desarrollo de la solución el **RUP** pues el mismo es un proceso que de manera ordenada define las tareas y quién de los miembros del equipo de desarrollo las hará. Es una guía para usar UML, es el resultado de varios años de desarrollo y uso práctico en el que se han unificado técnicas de desarrollo, a través del UML, y trabajo de muchas metodologías utilizadas por los clientes. En RUP se han agrupado las actividades en grupos lógicos definiéndose 9 flujos de trabajo principales. Los 6 primeros son conocidos como flujos de ingeniería y los tres últimos como de apoyo. [10]

Características de RUP:

- Creado por Jacobson, Rumbaugh y Booch.
- Unifica los mejores elementos de metodologías anteriores.
- Preparado para desarrollar grandes y complejos proyectos.
- Orientado a Objetos.
- Utiliza el UML como lenguaje de representación visual.

Herramientas Cases utilizada.

Se ha probado que el 80% de los errores lógicos de cualquier aplicación se originan en las fases tempranas del análisis y diseño del sistema, no en el proceso de codificación. [13]

La herramienta con la que se trabaja es el Rational Rose, por ser 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.

FUNDAMENTACIÓN TEÓRICA

1.13 Gestor de referencias bibliográficas: Endnote

EndNote es un programa que permite la gestión de referencias bibliográficas de cualquier tipo de documento (presenta hasta **41** tipos de referencias). Estas referencias (ya sean incorporadas manualmente, ya sean importadas desde bases de datos) pueden ser insertadas en documentos Word o editadas como bibliografía. EndNote proporciona hasta **2300** estilos bibliográficos para la edición de las mismas. Para lograr uniformidad en las referencias bibliográficas es que se utiliza esta herramienta. [22]

Conclusiones.

Como conclusiones parciales en este capítulo se plantea la creación de un sistema de gestión de profesores para la facultad 9, el que gestione los datos de sus profesores de una forma eficiente y rápida para un mejor control. El sistema se implementa a través de una aplicación Web que está soportada en la plataforma .NET utilizando el Visual Studio como herramienta de diseño debido a que es una completa herramienta para generar e integrar con rapidez aplicaciones Web. Como gestor de base de dato el SQL, pues es capaz de poner a disposición de muchos usuarios grandes cantidades de datos de manera simultánea. La metodología puesta en práctica es el RUP que utiliza el UML como lenguaje de representación visual a través de la herramienta CASES Rational Rose.

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Capítulo 2. Presentación de la solución propuesta

Introducción

En este capítulo se aborda lo referente a la modelación del negocio y del sistema, de la solución propuesta para el Sistema de Control para la Gestión de la Información del Profesor de la Facultad 9, las descripciones de los casos de uso del negocio y del sistema, actores y trabajadores del negocio, además se muestran los requisitos funcionales y no funcionales que presenta el sistema que se propone.

2.1 Modelo de Negocio

La comprensión de la estructura y la dinámica de la facultad en la cual se va a implantar el sistema automatizado, así como sus problemas actuales, identificando sus mejoras potenciales permiten la obtención de los requerimientos que debe cumplir el sistema a desarrollar. Por lo que se realiza modelo de negocio el que está compuesto por el diagrama de caso de uso del negocio, el modelo de objetos del negocio y la descripción de los casos de usos implicados en el negocio.

2.1.1 Actores y trabajadores del negocio

Actores del Negocio	Justificación
Decano	Es quien solicita la mayoría de las informaciones que fluyen en el negocio.
Interesado	Aquel que pueda estar interesado en conocer cualquier dato referente al profesor, que puede ser el propio profesor u otros directivos de la Facultad.

Tabla 1 Actores del negocio

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Trabajadores del negocio	Justificación
Profesor	Es el encargado de registrar sus datos.
Directivo	Es el que se encarga de controlar la asistencia de profesores a las actividades, así como gestionar las actividades. Puede ser un Jefe de Dpto, Vicedecano.
Asistente de control	Es el encargado de archivar los datos recogidos del profesor.

Tabla 2 Trabajadores del negocio

2.1.2 Procesos de negocio

En la Universidad de las Ciencias Informáticas (UCI) no se utiliza actualmente ningún sistema automatizado que permita la gestión de información de profesores a nivel de facultad, de forma que este, haga eficiente la labor de todos aquellos que necesiten trabajar con estos datos diariamente.

Este proceso se inicia tomando como base la facultad 9, cuando el profesor se incorpora a una facultad donde se le entrega un boleta, con diferentes datos personales y curriculares, que debe llenar, culminando así, el proceso de registro del profesor. Actualmente en la facultad todo este proceso se realiza de forma manual lo que impide una correcta manipulación de la información relacionada con los profesores por parte de todos aquellos directivos que necesiten cualquier información relacionada con los profesores de la misma. Después de realizado este proceso el profesor tiene que cumplir con un conjunto de actividades, en el transcurso del curso, entre las que se encuentran:

- ✓ Participar en actividades políticas convocadas por la brigada, facultad o la UCI, sindicales, de la FEU, el PCC, la UJC y demás.
- ✓ Participación en grupos de investigación – producción.
- ✓ Presentar trabajos en los diferentes eventos de Ciencia y Técnica a nivel de Centro.
- ✓ Participar en los festivales culturales y deportivos, directa o indirectamente apoyando a la representación de la Facultad.

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

- ✓ Participar en las actividades del departamento y colectivo de la asignatura al que pertenezca.
- ✓ Estar involucrado en un proyecto investigativo.
- ✓ Cumplir con las tareas encomendadas por la dirección de la facultad o la universidad.

Esto conlleva a una gran carga de trabajo para los directivos que carecen de una fuente centralizada, que almacene los datos de todos los profesores de la facultad.

Por las razones expuestas se resolvió efectuar la implementación de un Sistema de Control para la Gestión de la Información del profesor de la Facultad 9 que permita aliviar la labor de sus implicados, sustituyendo el trabajo manual por el automatizado. Además ofrezca información referente al profesor en el momento deseado tanto a los propios profesores como a los dirigentes de la facultad permitiendo así a los directivos un mejor uso de la información que se recopile en el sistema.

2.1.3 Diagrama de Casos de Uso del Negocio

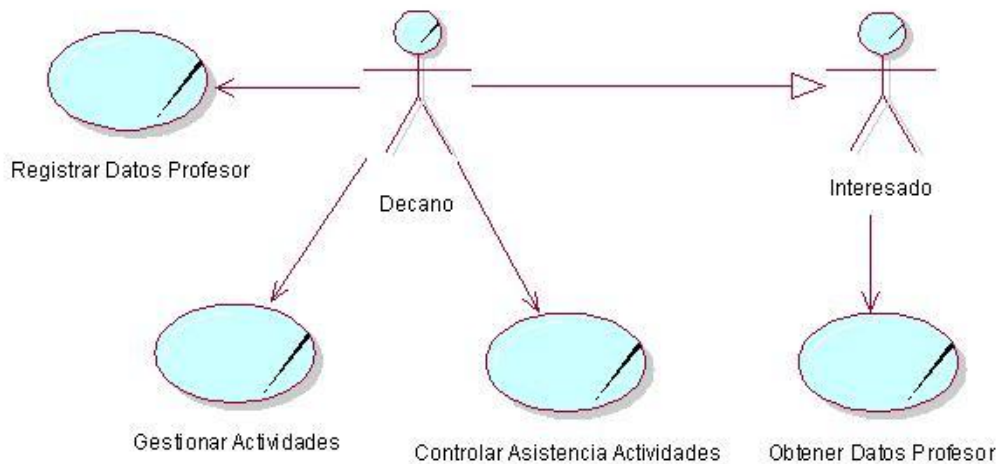


Figura 4 Diagrama de Casos de Uso del Negocio

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

2.1.4 Descripción textual de los Casos de Uso de Negocio

Caso de Uso:	Registrar Datos Profesor.	
Actores:	Decano (inicia)	
Trabajadores:	Profesor, Asistente de Control	
Resumen:	El caso de uso se inicia cuando el decano le orienta al profesor que registre sus datos, el asistente de control le entrega una boleta, este la llena. En caso de estar correcta se archiva, finalizando así el caso de uso.	
Precondiciones:		
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Negocio	
1. El decano orienta registrar los datos del profesor.	1.1. El profesor solicita llenar la boleta. 1.2 El asistente de control entrega al profesor una boleta para que registre sus datos. 1.3 El profesor llena la boleta y se la entrega al asistente de control. 1.4 El asistente de control revisa la boleta. 1.5 El asistente de control archiva la boleta.	
Flujos Alternos		
Acción del Actor	Respuesta del Negocio	
Actividad 1.5. Si la boleta no está correcta el asistente de control le informa el problema al profesor y le devuelve la boleta.		
Poscondiciones	El profesor queda registrado.	

Tabla 3 Descripción textual del Casos de Uso Registrar Datos Profesor

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Caso de Uso:	Controlar Asistencia Actividades	
Actores:	Decano (inicia)	
Trabajadores:	Directivo	
Resumen:	El caso de uso se inicia cuando el Decano le solicita al Directivo que controle la asistencia a las diferentes actividades que debe realizar el profesor.	
Precondiciones:		
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Negocio	
1. El Decano le orienta al Directivo que tiene que controlar la asistencia de los profesores a las actividades.	1.1 El Directivo controla la asistencia. 1.2 Elabora un listado con el nombre de todos los presentes. 1.3 Archiva el listado. 1.4 Le informa la asistencia al Decano.	
Poscondiciones	La asistencia a las actividades queda archivada.	

Tabla 4 Descripción textual del Casos de Uso Controlar Asistencia Actividades

Caso de Uso:	Gestionar Actividades	
Actores:	Decano (inicia)	
Trabajadores:	Directivo	
Resumen:	El caso de uso se inicia cuando el Decano le orienta a un Directivo que planifique sus actividades y le de un reporte de las mismas.	
Precondiciones:		
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Negocio	
1. El Decano orienta al Directivo que tiene que planificar las diferentes actividades de los profesores.	1.1 El directivo planifica las actividades. 1.2 Informa las actividades al profesor. 1.3 Actualiza el registro de actividades. 1.4 Le informa al Decano las actividades.	
Poscondiciones	Las actividades quedan planificadas.	

Tabla 5 Descripción textual del Casos de Uso Gestionar Actividades

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Caso de Uso:	Obtener Datos Profesor	
Actores:	Interesado(inicia)	
Trabajadores:	Asistente Control	
Resumen:	El caso de uso se inicia cuando cualquier interesado le solicita al asistente de control que le muestre los datos de un profesor.	
Precondiciones:		
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Negocio	
1. El interesado le solicita al Asistente de Control que le muestre los datos de un profesor.	1.1 El Asistente de Control busca en el registro de Boletas la boleta del profesor. 1.2 Le pregunta que datos desea conocer	
2. El interesado informa los datos a buscar.	2.1 El asistente busca los datos en la boleta 2.2 Le informa al interesado los datos deseados	
Poscondiciones		

Tabla 6 Descripción textual del Casos de Uso Obtener Datos Profesor

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

2.1.5 Diagrama de actividades

El diagrama de actividad es un grafo que contiene estados en que puede hallarse una actividad.

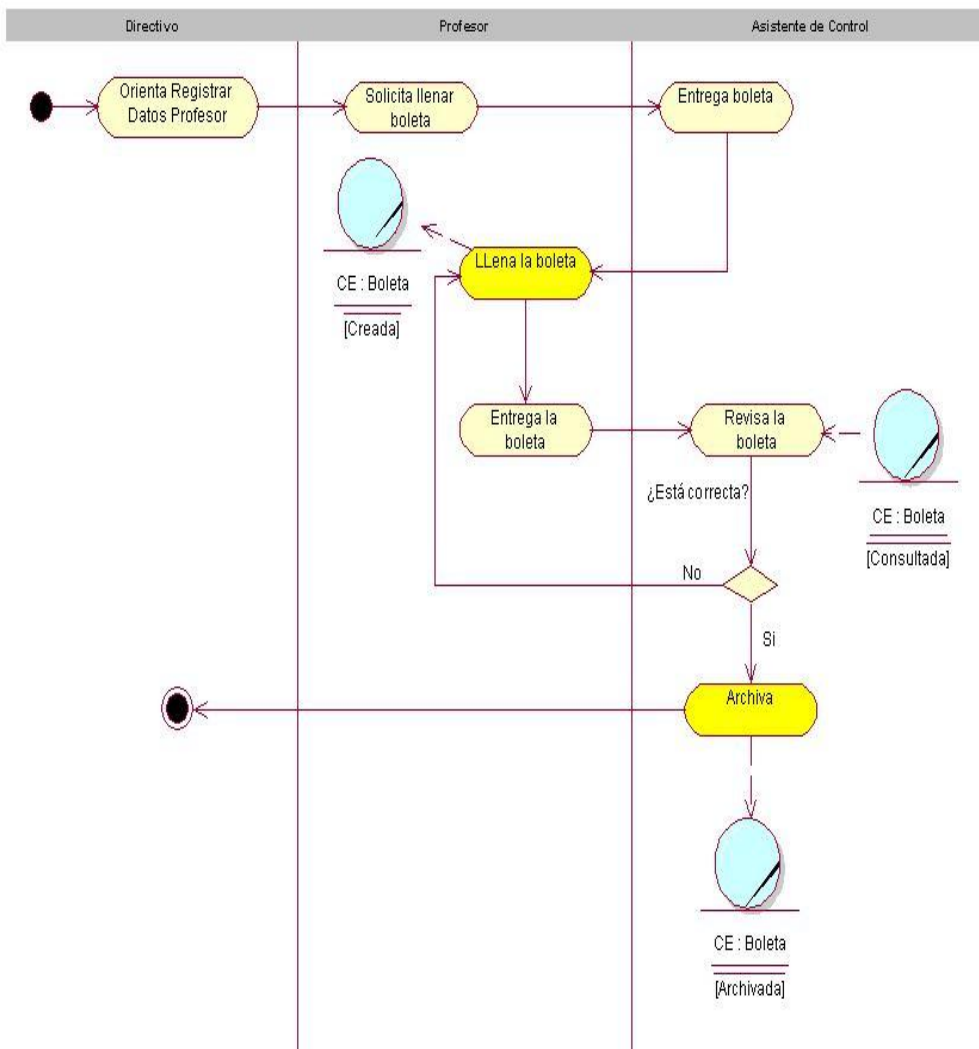


Figura 5 Diagrama de Actividades Caso de Uso: Registrar Datos Profesor

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

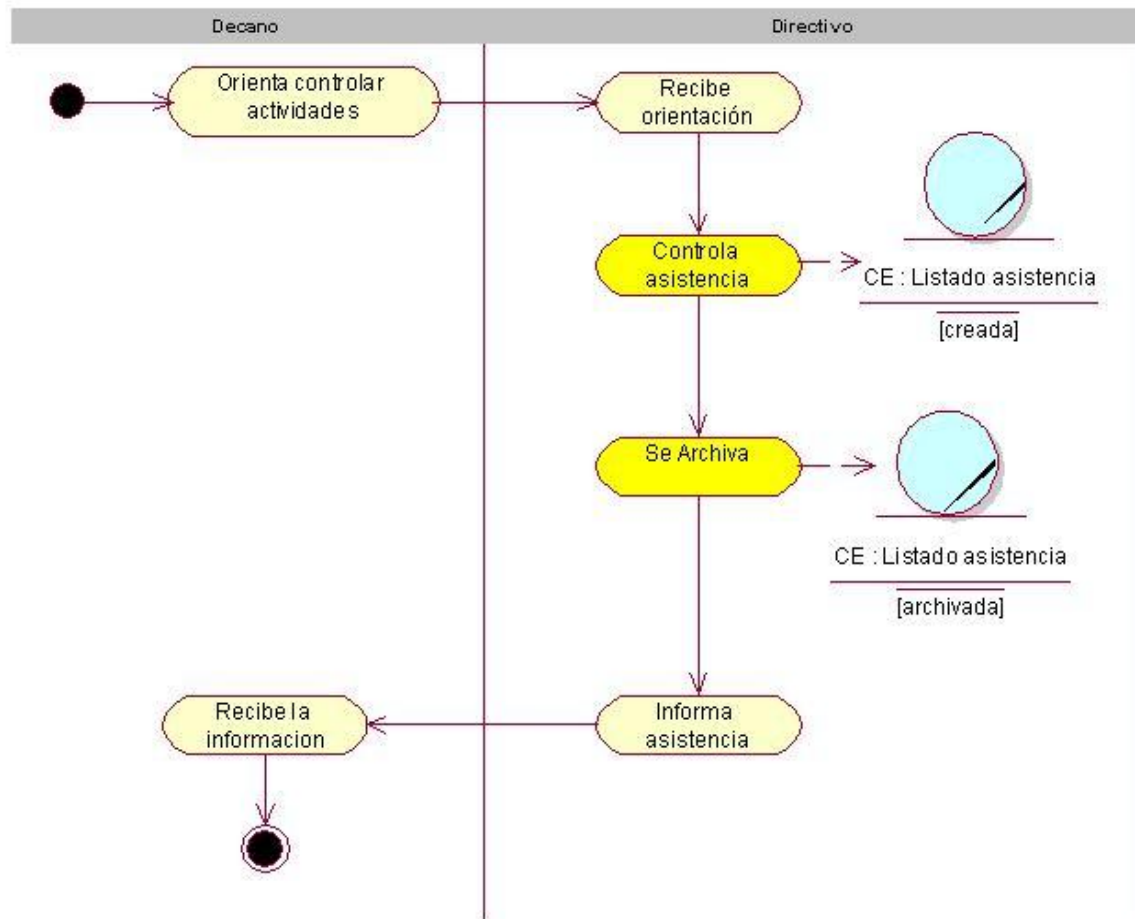


Figura 6 Diagrama de Actividades Caso de Uso: Controlar Asistencia Actividades

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

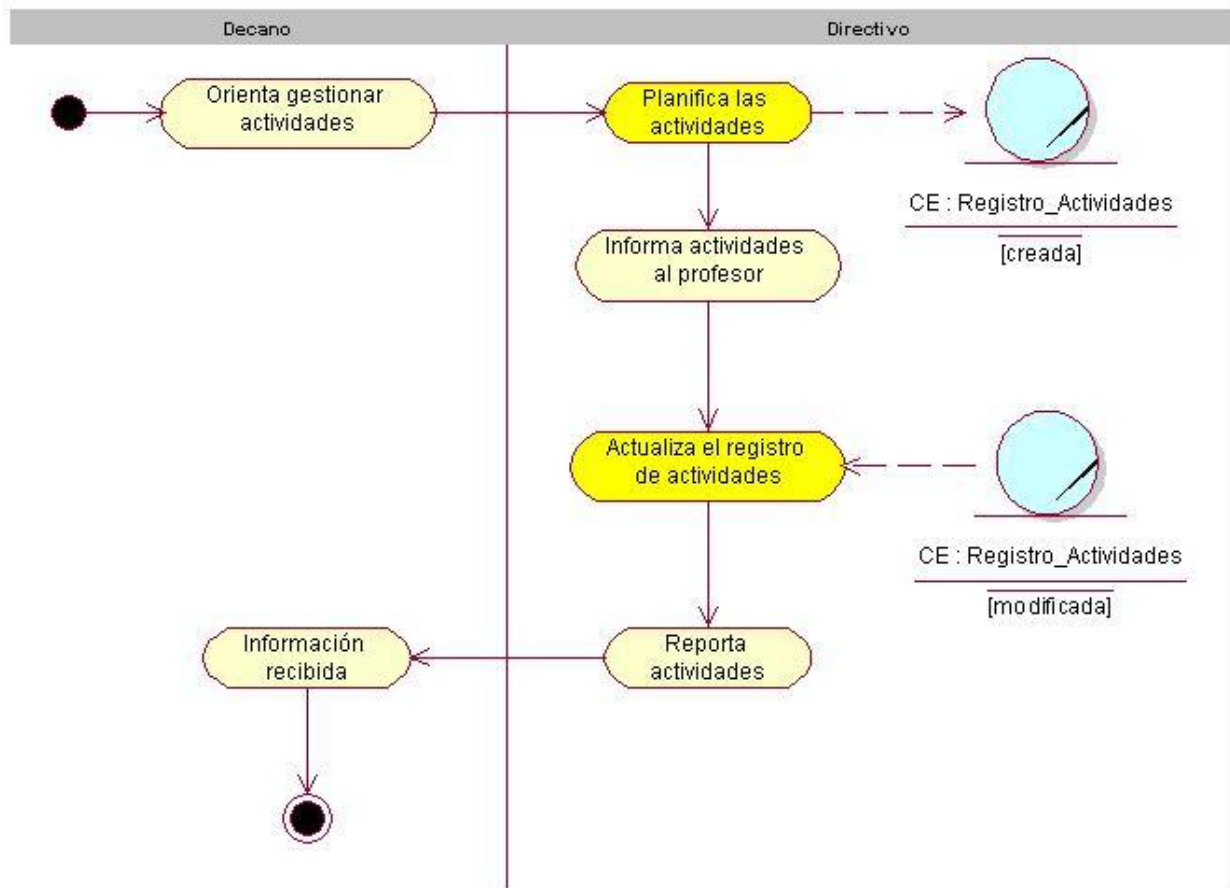


Figura 7 Diagrama de Actividades Caso de Uso: Gestionar Actividades

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

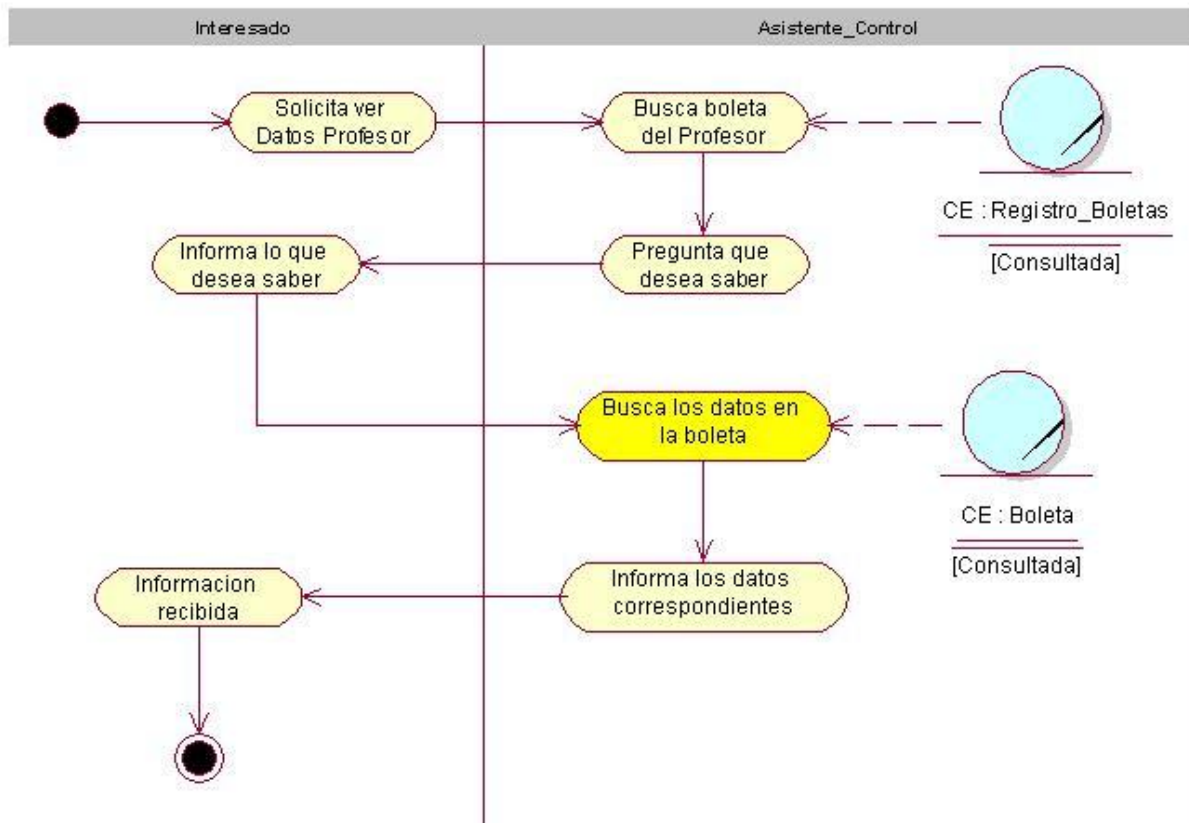


Figura 8 Diagrama de Actividades Caso de Uso: Obtener Datos Profesor

2.1.6 Modelo de Objetos

El modelo de objetos del negocio, muestra la participación de los trabajadores y entidades del negocio y la relación que se establece entre ellos. Describe cómo cada caso de uso del negocio es llevado a cabo por un conjunto de trabajadores que utilizan un conjunto de entidades del negocio.

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

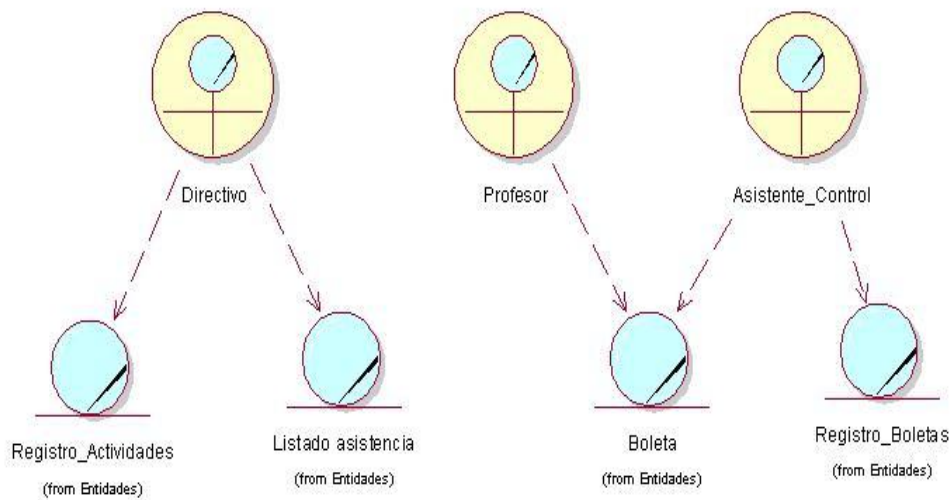


Figura 9 Diagrama de Objetos

2.2 Requerimientos.

- ✓ Condición o capacidad que necesita un usuario para resolver un problema o lograr un objetivo.
- ✓ Condición o capacidad que tiene que ser alcanzada o poseída por un sistema o componente de un sistema para satisfacer un contrato, estándar, u otro documento impuesto formalmente.
- ✓ Define qué es lo que el sistema debe hacer, para lo cual se identifican las funcionalidades requeridas y las restricciones que se imponen.
- ✓ Es una característica que un sistema debe tener para cubrir alguna de las necesidades que lo motivan.

Los requisitos o requerimientos se pueden clasificar en: funcionales y no funcionales. Los requerimientos funcionales son capacidades o condiciones que el sistema debe cumplir, en cambio, los no funcionales son propiedades o cualidades que el producto debe tener.

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

2.2.1 Requerimientos Funcionales

RF1 Autenticar Usuario.

RF2 Gestionar Datos Profesor.

2.1 Registrar Datos Profesor.

2.2 Modificar Datos Profesor.

2.3 Eliminar Datos Profesor.

RF3 Gestionar Asistencia Actividades.

3.1 Registrar Asistencia Actividades.

3.2 Eliminar Asistencia Actividades.

RF4 Gestionar Actividades.

4.1 Registrar Actividades.

4.2 Modificar Actividades.

4.3 Eliminar Actividades.

RF5 Gestionar Usuario

5.1 Registrar Usuario

5.2 Modificar Usuario

5.3 Eliminar Usuario

RF6 Buscar Información

6.1 Buscar Profesor

6.2 Buscar Actividad

6.3 Buscar Asistencia Actividad

2.2.2 Requerimientos No Funcionales

Apariencia o interfaz externa

- ✓ La interfaz debe ser sencilla y fácil de usar por los usuarios.
- ✓ En caso de que no se pueda realizar una acción por un usuario, visualizar un mensaje que especifique por que no se pudo ejecutar la acción.
- ✓ Debe mostrar lo más claramente posible todas las opciones con las que cuenta el sistema con el

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

objetivo de agilizar todo el proceso.

Usabilidad

Para utilizar la aplicación Web es necesario estar registrado en el sistema y en dependencia del rol que posea, tendrá acceso o no, a la información solicitada.

Rendimiento

Se debe garantizar que la respuesta a solicitudes de los usuarios del sistema sea en un tiempo breve.

Seguridad

- ✓ Se les garantizará el acceso a la información a los usuarios autorizados sin que los mecanismos de seguridad oculten o retrasen a los usuarios la obtención de sus datos en el momento deseado.
- ✓ Mantener actualizada la información de los profesores.

Implementación

Será implementada haciendo uso de la tecnología .Net y SQL Server 2000 como Gestor de Base de Datos.

Software

El sistema operativo será Windows 95 o Superior, tanto en la PC cliente como servidora, servidor SQL Server 2000 y .Net Framework.

Hardware

Clientes:

Pentium de 133 MHz o superior.

128 MB de memoria RAM mínima, 256 MB de memoria RAM

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Servidor Base Dato (SQL)

Dual PIII de 933MHz o superior.

512 MB de memoria RAM.

100 GB de disco duro

Servidor Web

Dual PIII de 933MHz o superior

512 MB de memoria RAM.

60 GB de disco duro.

2.3 Descripción del Sistema Propuesto

2.3.1 Descripción de los actores

Actores del Sistema	Justificación
Directivo	Es el encargado de llevar a cabo todos lo relacionado con el control de la asistencia de los profesores a las diferentes actividades así como gestionar las mismas. Puede ser un Jefe de Dpto, Vicedecano, Decano.
Profesor	Es el encargado de registrar todos sus datos en el sistema.
Jefe de Dpto	Es el encargado de eliminar a un profesor.
Administrador	Es el encargado de gestionar los usuarios.
Usuario	Es el encargado de la autenticación de los usuarios así como buscar cualquier información en el sistema.

Tabla 7 Descripción de los actores del sistema

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

2.3.2 Paquetes con sus relaciones

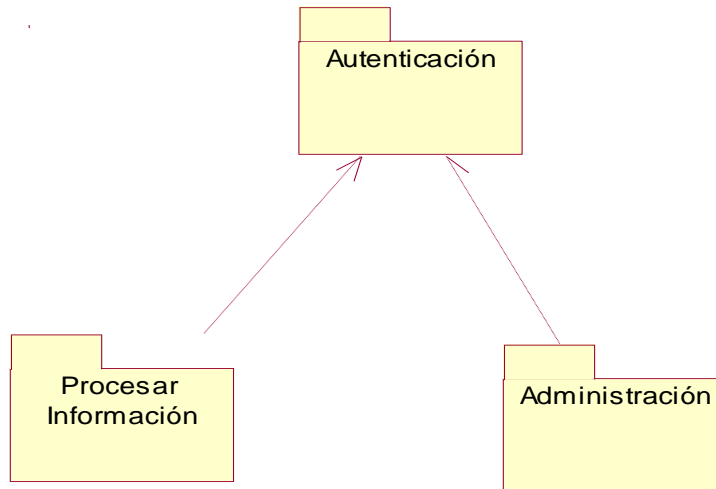


Figura 10 Relación entre los paquetes del sistema

2.3.3 Diagramas por paquetes

Paquete Autenticación

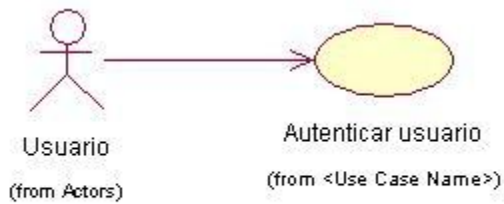


Figura 11 Diagrama de casos de uso del sistema para paquete autenticación

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Paquete Administración

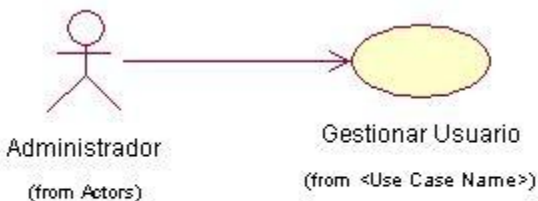


Figura 12 Diagrama de casos de uso del sistema para paquete administración

Paquete Procesar Información

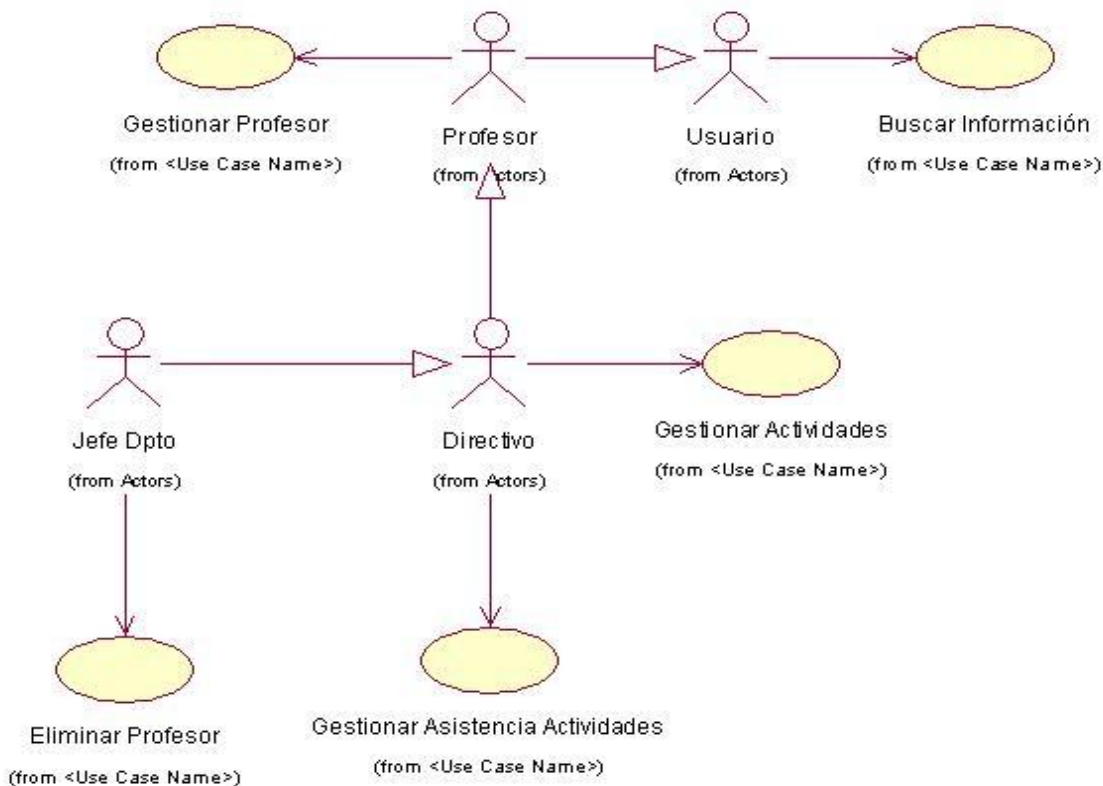


Figura 13 Diagrama de casos de uso del sistema para Paquete Procesar Información

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

2.3.4 Descripciones de los Casos de Uso del Sistema

Paquete Autenticación	
Nombre del CU	Autenticar Profesor
Actores	Profesor (inicia)
Propósito	Permitir autenticarse.
Resumen	El caso de uso se inicia cuando el profesor introduce los datos que se le piden para acceder a la aplicación, estos se verifican y finaliza dándole acceso a la información que le corresponde.
Referencias	RF1
Precondiciones	
Poscondiciones	Se habilitan las funcionalidades según sus privilegios.
Curso Normal de los Eventos	
Acciones del Actor	Respuesta del Sistema
1. El profesor entra usuario y contraseña.	1.1 El sistema encripta la contraseña. 1.2 Busca el usuario y compara la contraseña. 1.3 Se le asignan los permisos.
Flujos alternativos	
Acciones del Actor	Respuesta del Sistema
	1.3 En caso de que no exista se le envía un aviso.
Prioridad:	Crítico
Prototipo de Interfaz	Ver anexos 1

Tabla 8 Descripciones del Caso de Uso del Sistema Autenticar Profesor

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Paquete Procesar Información	
Nombre del Caso de Uso	Gestionar Profesor
Actores	Profesor(inicia)
Propósito	Permitir registrar o modificar los datos del profesor.
Resumen	El caso de uso se inicia cuando el profesor decide gestionar sus datos, escogiendo la sección de “Registrar Datos del Profesor” o “Modificar Datos del Profesor”.
Referencias	RF2
Precondiciones	El profesor debe autenticarse.
Poscondiciones	Se registra o modifican los datos del profesor.
Curso Normal de los Eventos	
Acciones del Actor	Respuesta del Sistema
1. El Profesor necesita registrar o modificar sus datos.	1.1. El sistema ejecuta alguna de las siguientes acciones: a) Si decide registrar sus datos, ir a la sección “Registrar Datos del Profesor” b) Si decide actualizar alguno de sus datos, ir a la sección “Modificar Datos del Profesor”.
Sección “Registrar Datos del Profesor”	
Acciones del Actor	Respuesta del Sistema
2. El Profesor entra los datos que se le pide	2.1 El sistema verifica que todos los campos estén llenos. 2.2 El sistema verifica que el profesor no exista. 2.3 Los datos del profesor se almacenan en el sistema.

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

	2.4 Se muestra un mensaje informándosele al profesor que ya ha sido registrado.
Sección “Modificar Datos del Profesor”	
Acciones del Actor	Respuesta del Sistema
	3.1 El sistema muestra todos los datos que el profesor tiene almacenado.
4. El profesor realiza las modificaciones deseadas.	4.1 Se actualiza la información en la BD del sistema.
Curso Alternativo de los Eventos	
2.1 Si faltan datos obligatorios se presenta un mensaje que indica que elementos obligatorios le faltan.	
2.2 Si el profesor ya existe en el sistema se presenta un mensaje.	
Prioridad:	Crítico
Prototipo de Interfaz	Ver anexos 2 y 3

Tabla9 Descripciones del Caso de Uso del Sistema Gestionar Profesor

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Paquete Procesar Información	
Nombre del Caso de Uso	Eliminar Profesor
Actores	Jefe Dpto. (inicia)
Propósito	Permitir eliminar a un profesor.
Resumen	El caso de uso se inicia cuando el Jefe de Dpto decide eliminar un profesor.
Referencias	RF2
Precondiciones	El profesor debe estar registrado en el sistema.
Poscondiciones	Se elimina el profesor.
Curso Normal de los Eventos	
Acciones del Actor	Respuesta del Sistema
1. El Jefe de Dpto decide eliminar a un profesor.	1.1. El sistema muestra un listado de todos los profesores registrados.
2. El Jefe de Dpto escoge el profesor a eliminar.	2.1 El sistema pide confirmación. 2.2 El sistema elimina el profesor.
Prioridad:	Crítico
Prototipo de Interfaz	Ver anexos 4

Tabla 10 Descripciones del Caso de Uso del Sistema Eliminar Profesor

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Paquete Procesar Información	
Nombre del Caso de Uso	Gestionar Asistencia Actividades
Actores	Directivo(inicia)
Propósito	Permite registrar o eliminar las asistencias.
Resumen	El caso de uso se inicia cuando un directivo decide gestionar la asistencia a las actividades, pudiendo registrar o eliminar la asistencia a las actividades.
Referencias	RF3
Precondiciones	El Directivo debe autenticarse.
Poscondiciones	Se registra o elimina la asistencia a una actividad.
Curso Normal de los Eventos	
Acciones del Actor	Respuesta del Sistema
1. El Directivo necesita registrar o eliminar la asistencia a las actividades que debe realizar el profesor.	1.1. El sistema ejecuta alguna de las siguientes acciones: a) Si decide registrar asistencia actividades, ir a la sección "Registrar Asistencia Actividades" b) Si decide actualizar alguno de sus datos, ir a la sección "Eliminar Asistencia Actividades".
Sección "Registrar Asistencia Actividades"	
Acciones del Actor	Respuesta del Sistema
2. El Directivo entra la asistencia a las actividades.	2.1 El sistema verifica que todos los campos estén llenos. 2.2 La asistencia de la actividad se almacenan en el sistema. 2.3 Se muestra un mensaje informando al directivo que ya ha sido registrada la

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

	asistencia a la actividad.
Sección “Eliminar Asistencia Actividad”	
Acciones del Actor	Respuesta del Sistema
3 El Directivo selecciona la actividad a la que desea eliminar la asistencia.	3.1 El sistema muestra todas las asistencias de la actividad registradas hasta el momento.
4. El Directivo elimina la asistencia deseada.	4.1 El sistema muestra un mensaje para confirmar que desea eliminar
5. El Directivo confirma que desea eliminar la asistencia	5.1 Se actualiza la información en la BD del sistema.
Curso Alternativo de los Eventos	
2.1 Si faltan datos obligatorios se presenta un mensaje que indica que elementos obligatorios le faltan.	
Prioridad:	Crítico
Prototipo de Interfaz	Ver anexos 5 y 6

Tabla 11 Descripciones del Caso de Uso del Sistema Gestionar Asistencia Actividades

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Paquete Procesar Información	
Nombre del Caso de Uso	Buscar Información
Actores	Usuario (inicia)
Propósito	Permite buscar cualquier información del profesor.
Resumen	El caso de uso se inicia cuando el usuario decide ver cualquier información, ya sea los datos del profesor, como su asistencia a las actividades, además de las diferentes actividades que planifican los directivos.
Referencias	RF6
Precondiciones	El usuario debe autenticarse.
Poscondiciones	Se muestra la información deseada
Curso Normal de los Eventos	
Acciones del Actor	Respuesta del Sistema
1. El usuario necesita buscar información	1.1. El sistema ejecuta alguna de las siguientes acciones: a) Si decide ver los datos de un profesor, ir a la sección "Buscar Profesor" b) Si decide ver las asistencias a las actividades, ir a la sección "Buscar Asistencia". c) Si decide ver las actividades, ir a la sección "Buscar Actividad"
Sección "Buscar Profesor"	
Acciones del Actor	Respuesta del Sistema
2. El usuario selecciona los datos por lo cual quiere buscar a un profesor.	2.1. El sistema muestra un listado de todos los profesores registrados con dichos datos.

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Sección “Buscar Asistencia”	
Acciones del Actor	Respuesta del Sistema
3. El usuario selecciona la actividad de la cual quiere conocer la asistencia.	3.1. El sistema muestra un listado con los profesores que asistieron a dicha actividad.
Sección “Buscar Actividad”	
Acciones del Actor	Respuesta del Sistema
4. El usuario selecciona los datos por lo cual quiere buscar la actividad.	4.1. El sistema muestra un listado de todas las actividades registradas con dichos datos.
Prioridad:	Crítico
Prototipo de Interfaz	Ver anexos 7, 8 y 9

Tabla 12 Descripciones del Caso de Uso del Sistema Mostrar Información

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Paquete Procesar Información	
Nombre del Caso de Uso	Gestionar Actividad
Actores	Directivo(inicia)
Propósito	Permite registrar, modificar y eliminar actividades.
Resumen	El caso de uso se inicia cuando el Directivo decide gestionar actividad, pudiendo registrar, modificar o eliminar la actividad.
Referencias	RF4
Precondiciones	El directivo debe autenticarse.
Poscondiciones	Se registra, modifica o elimina una actividad.
Curso Normal de los Eventos	
Acciones del Actor	Respuesta del Sistema
1. El Directivo necesita registrar, modificar o eliminar una actividad.	1.1. El sistema ejecuta alguna de las siguientes acciones: a) Si decide registrar una actividad, ir a la sección "Registrar Actividad" b) Si decide modificar alguno de los datos, ir a la sección "Modificar Actividad". c) Si decide eliminar alguna actividad, ir a la sección "Eliminar Actividad"
Sección "Registrar Actividad"	
Acciones del Actor	Respuesta del Sistema
2. El Directivo registra la actividad.	2.1 El sistema verifica que todos los campos estén llenos. 2.2 Los datos de la actividad se almacenan en el sistema. 2.3 Se muestra un mensaje

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

	informándosele al Directivo que ya se ha registrado la actividad.
Sección “Modificar Actividad”	
Acciones del Actor	Respuesta del Sistema
3. El Directivo selecciona el departamento al que pertenece la actividad que desea modificar.	3.1 El sistema muestra todas las actividades que tiene almacenada de ese departamento.
4. El Directivo realiza las modificaciones deseadas.	4.1 El sistema le muestra un mensaje para confirmar si desea modificar la actividad.
5. El Directivo confirma que desea modificar la actividad.	5.1 Se actualiza la información en la BD del sistema
Sección “Eliminar Actividad”	
Acciones del Actor	Respuesta del Sistema
6. El Directivo selecciona el departamento al que pertenece la actividad que desea eliminar.	6.1 El sistema muestra todas las actividades que tiene almacenada de ese departamento.
7. El Directivo selecciona la actividad deseada.	7.1 El sistema pide confirmación. 7.2 El sistema elimina la actividad 7.3 Se actualiza la información en la BD del sistema.
Curso Alternativo de los Eventos	
2.1 Si faltan datos obligatorios se presenta un mensaje que indica que elementos obligatorios le faltan.	
Prioridad:	Crítico
Prototipo de Interfaz	Ver anexos 10, 11 y 12

Tabla 13 Descripciones del Caso de Uso del Sistema Gestionar Actividad

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Paquete Administración	
Nombre del Caso de Uso	Gestionar Usuario
Actores	Administrador(inicia)
Propósito	El propósito del caso de uso es permitir la gestión de los usuarios
Resumen	El caso de uso inicia cuando el administrador se encarga de gestionar los usuarios de dicha aplicación, definiéndoles sus roles
Referencias	RF5
Precondiciones	El administrador debe de estar autenticado, el usuario al momento de insertarlo no debe estar en la base de datos, todo lo contrario cuando se actualiza o se elimina
Poscondiciones	El usuario ha sido registrado, modificado o eliminado.
Curso Normal de los Eventos	
Acciones del Actor	Respuesta del Sistema
1-El administrador desea registrar, modificar o eliminar usuarios	1.1. El sistema ejecuta alguna de las siguientes acciones: a) Si decide registrar un usuario, ir a la sección "Registrar Usuario" b) Si decide modificar un usuario, ir a la sección "Modificar Usuario". c) Si decide eliminar un usuario, ir a la sección "Eliminar Usuario "
Sección "Registrar Usuario"	
Acciones del Actor	Respuesta del Sistema
2- El administrador elige la opción registrar usuario.	2.1- El sistema muestra el formulario para registrar el usuario 2.2- El sistema verifica y en caso que no exista registra el usuario.

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Sección “Modificar Usuario”	
Acciones del Actor	Respuesta del Sistema
3- El administrador elige la opción de modificar usuario.	3.1- El sistema muestra el formulario para modificar el usuario 3.2- El sistema verifica que exista y modifica el usuario.
Sección “Eliminar Usuario”	
Acciones del Actor	Respuesta del Sistema
4- El administrador elige la opción de eliminar	4.1- El sistema muestra el formulario para eliminar el usuario 4.2- El sistema verifica que no haya sido eliminado y elimina el usuario.
Curso Alternativo de los Eventos	
2.2 Si el usuario ya existe el sistema muestra un mensaje que ya fue registrado. 3.2 Si el usuario no existe el sistema muestra un mensaje que el usuario no existe. 4.2 Si el usuario ya fue eliminado el sistema muestra un mensaje que ya fue eliminado.	
Prioridad:	Crítico
Prototipo de Interfaz	Ver anexos 13 y 14

Tabla 14 Descripciones del Caso de Uso del Sistema Gestionar Usuario

Conclusiones

Como conclusiones de este capítulo se realizó la modelación del negocio pues el conocimiento de las características y actividades que se llevan a cabo en el contexto que se desea automatizar determinaron las mejoras que puede aportar, al negocio, el sistema a desarrollar. Como resultado de esto se obtuvo un diagrama de casos de uso del negocio, un diagrama de caso de uso del sistema, descripciones de los casos de uso del negocio y del sistema respectivamente, así como el modelo de objetos del negocio, y los diagramas de actividades por cada caso de uso. Además se realizó la captura de los requerimientos funcionales del sistema y se obtuvo un listado de requisitos no funcionales a tener en cuenta para el desarrollo de la aplicación.

Capítulo 3. Construcción de la solución propuesta

Introducción

Este capítulo aborda los resultados de los flujos de trabajo de Análisis y Diseño e Implementación, es decir, se transforma los requisitos funcionales en un diseño de clases viendo las relaciones e interacción que existe entre ellos. Se representan los diagramas de clases del diseño Web y los nodos de la aplicación a través del Diagrama de Despliegue, el Diagrama de clases persistentes así como el modelo de datos obtenido a partir del anterior. Además se modela el Diagrama de componentes. Se expone así la construcción de la solución propuesta.

3.1 Diagramas de Clases

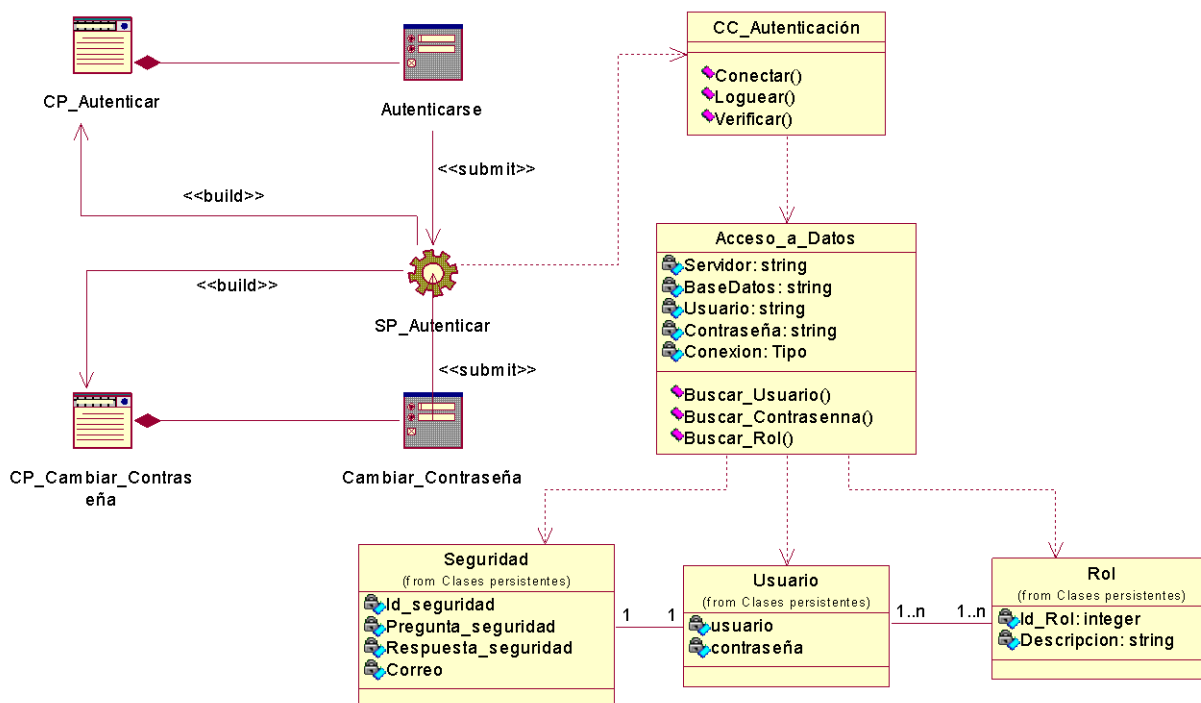


Figura 14 Diagrama de clases de diseño para el Paquete Autenticar caso de uso Autenticar Usuario

CONSTRCCIÓN DE LA SOLUCIÓN PROPUESTA

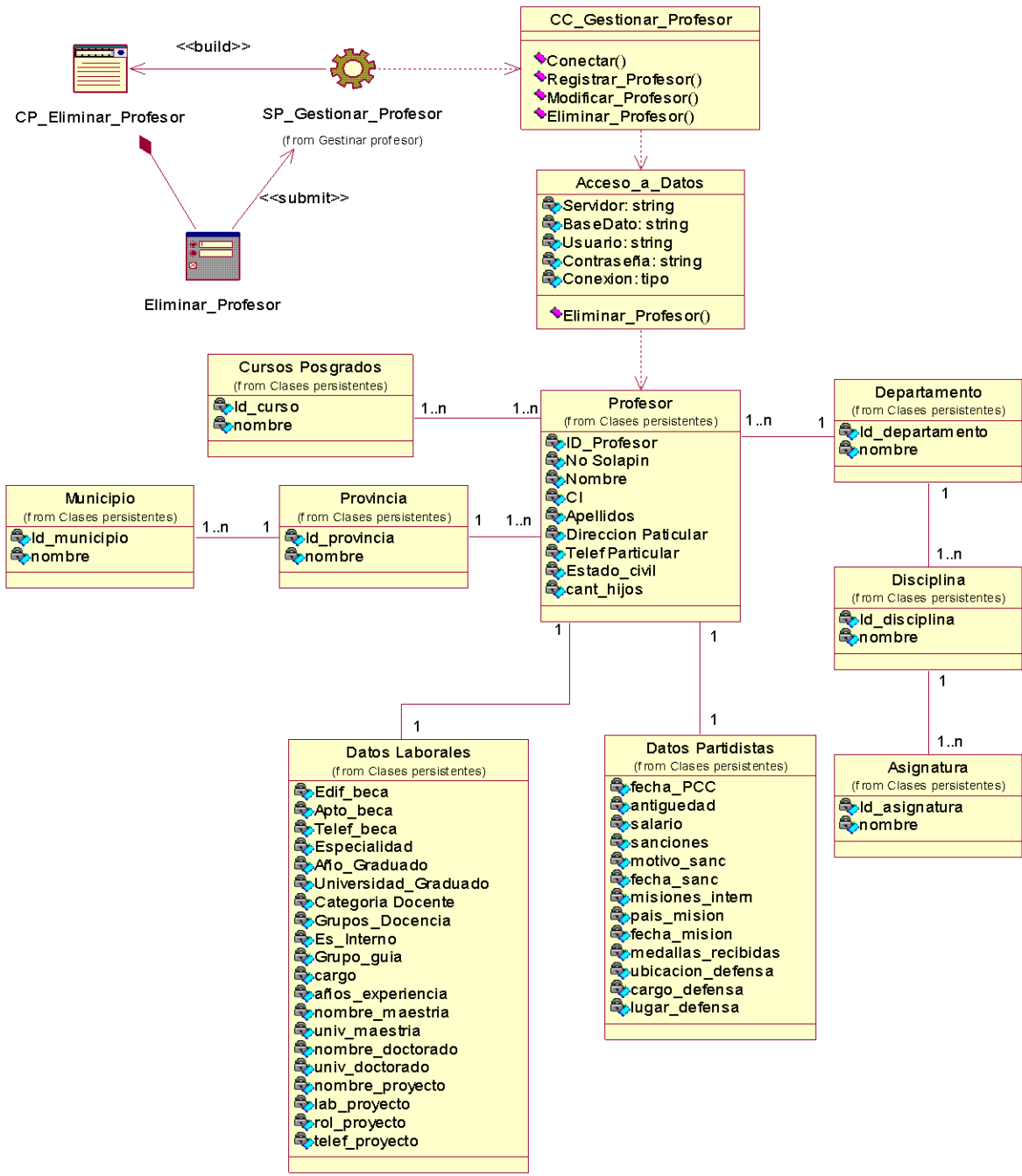


Figura 15 Diagrama de clases de diseño para el Paquete Procesar Información caso de uso Eliminar Profesor

CONSTRCCIÓN DE LA SOLUCIÓN PROPUESTA

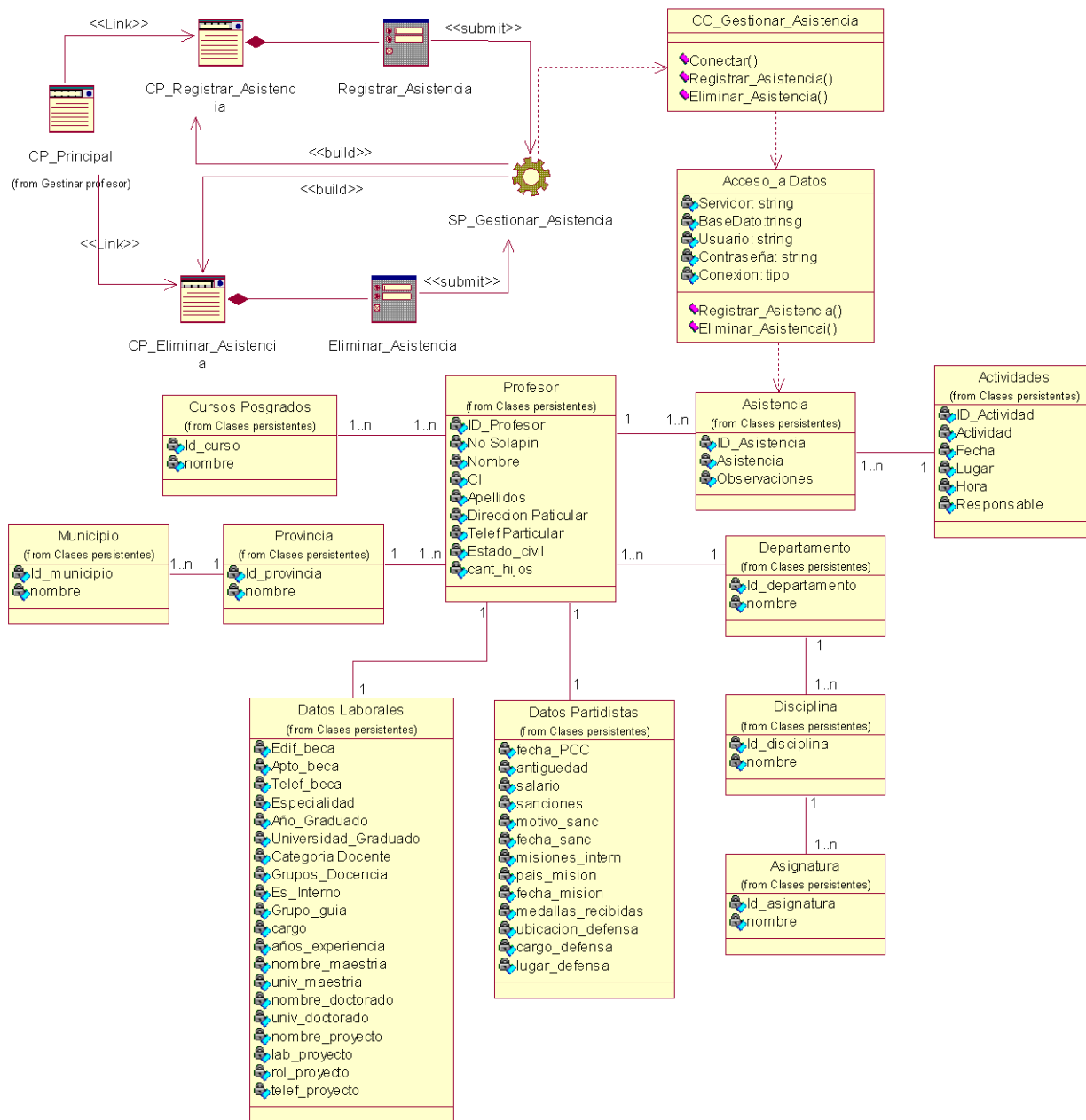


Figura 16 Diagrama de clases de diseño para el Paquete Procesar Información caso de uso Gestionar Asistencia Actividades

CONSTRCCIÓN DE LA SOLUCIÓN PROPUESTA

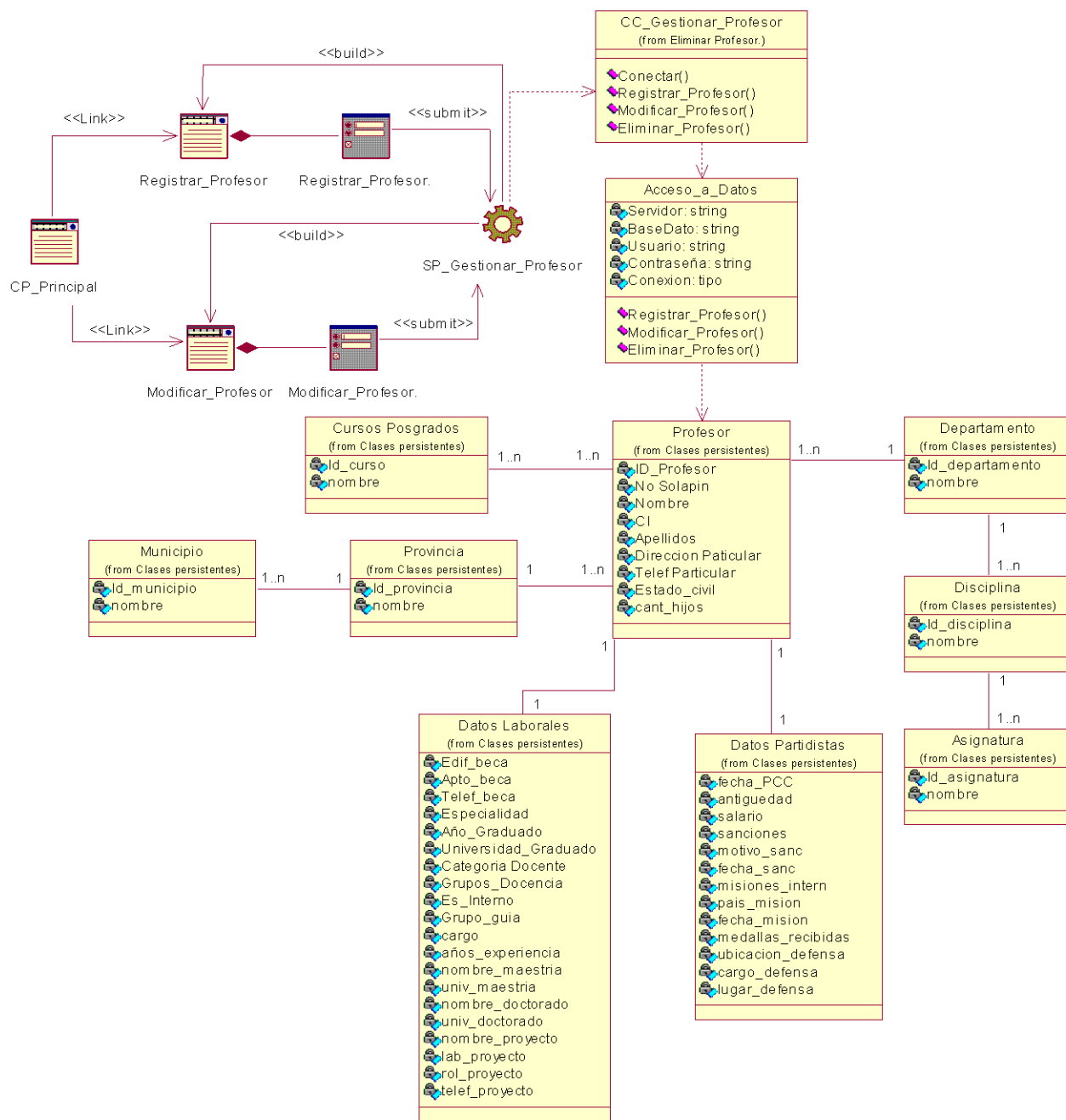


Figura 17 Diagrama de clases de diseño para el Paquete Procesar Información caso de uso Gestionar Profesor

CONSTRCCIÓN DE LA SOLUCIÓN PROPUESTA

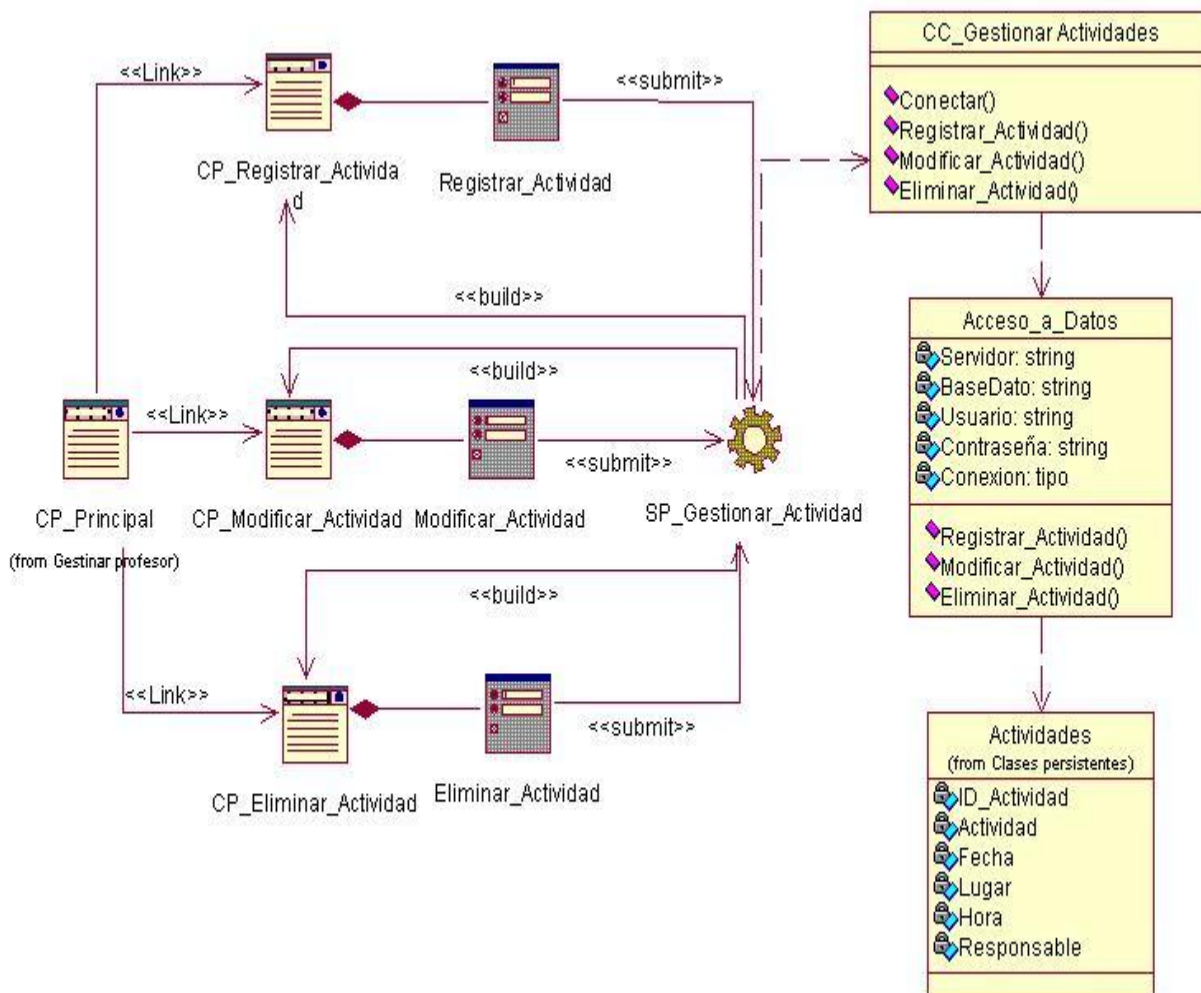


Figura 18 Diagrama de clases de diseño para el Paquete Procesar Información caso de uso Gestionar Actividad

CONSTRCCIÓN DE LA SOLUCIÓN PROPUESTA

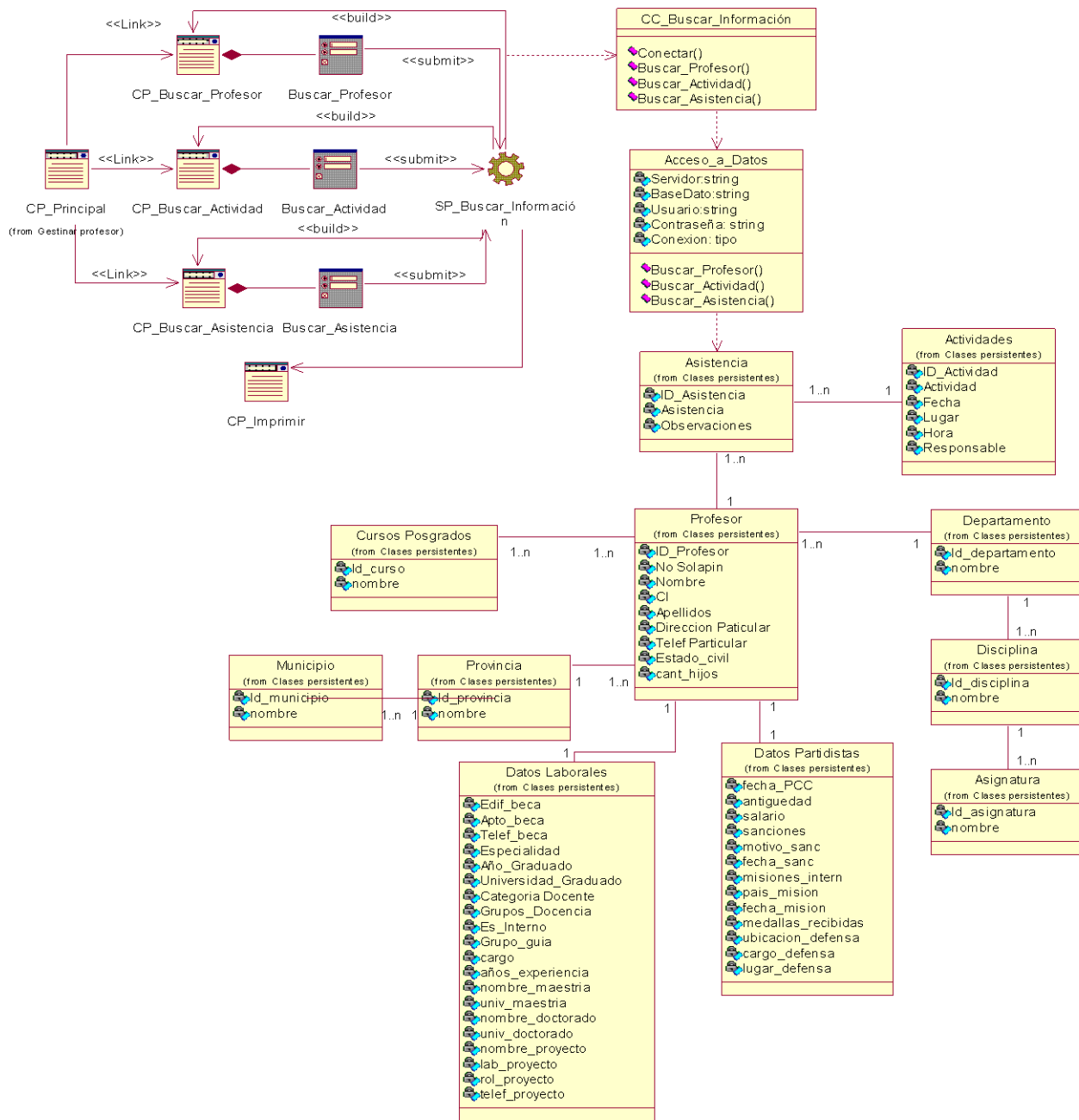


Figura 19 Diagrama de clases de diseño para el Paquete Procesar Información caso de uso Buscar Información

CONSTRCCIÓN DE LA SOLUCIÓN PROPUESTA

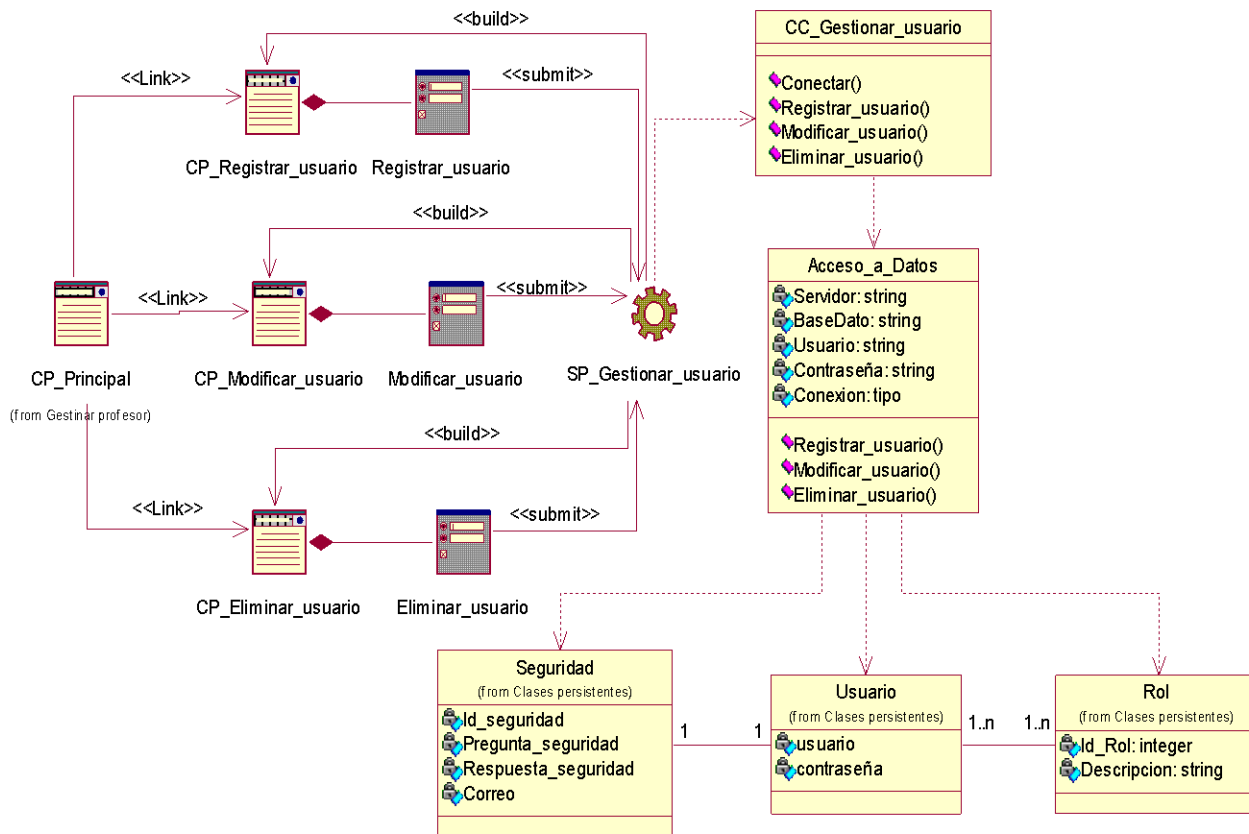


Figura 20 Diagrama de clases de diseño para el Paquete Administración caso de uso Gestionar Usuario

CONSTRCCIÓN DE LA SOLUCIÓN PROPUESTA

3.2 Principios de diseño

3.2.1 Estándares de la interfaz de la aplicación

Los factores de calidad de una aplicación que influyen en la satisfacción del usuario con la misma, se puede clasificar en aquellos relacionados con la calidad de los contenidos, y la calidad del diseño de la aplicación. La importancia del diseño de la aplicación se basa en que éste será el que modele la interacción entre los usuarios y la aplicación, y por tanto posibilitará o no la consecución de los objetivos perseguidos por el usuario (encontrar información, registrar datos, etc.).

Para el diseño de la interfaz del sistema se tuvo en cuenta varios aspectos, como las operaciones que el usuario puede realizar, para ello se ubicaron las mismas en la parte izquierda y superior de la aplicación logrando así que estas estén accesibles. Se trabaja con plantilla para lograr uniformidad entre todas las páginas de la aplicación. Las páginas son formularios lo más entendibles y cómodos posible. La letra usada en casi todo el sistema es Times New Roman 11, y en el caso de los títulos Times New Roman 14 que es una de las letras más usadas a nivel mundial, enfatizando en algunas cosas necesarias con negrita, el fondo es de color #ffffff que por ser claro hace un buen contraste #191970 y #d3d3d3, que facilita la lectura al usuario.

Para evitar errores, primeramente el sistema valida la entrada de los usuarios, dándole acceso de acuerdo al rol del mismo en el sistema. En caso de error por datos incorrectos se muestra un mensaje al usuario.

CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

3.3 Diseño de la Base de Datos

3.2.1 Diagrama de clases persistentes

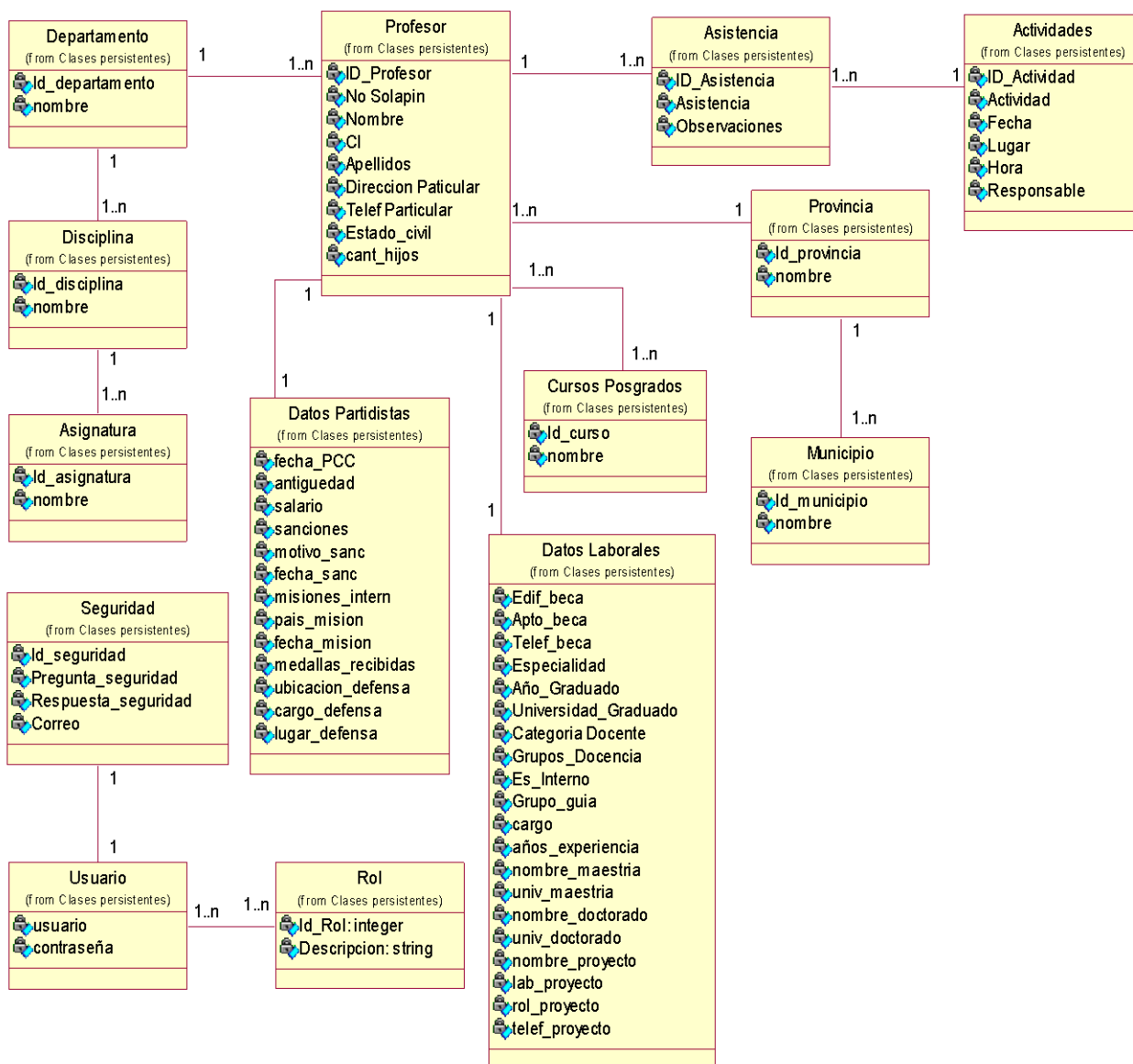


Figura 21 Diagrama de clases persistentes

CONSTRCCIÓN DE LA SOLUCIÓN PROPUESTA

3.4 Generalidades de la Implementación

3.4.1 Modelo de Despliegue

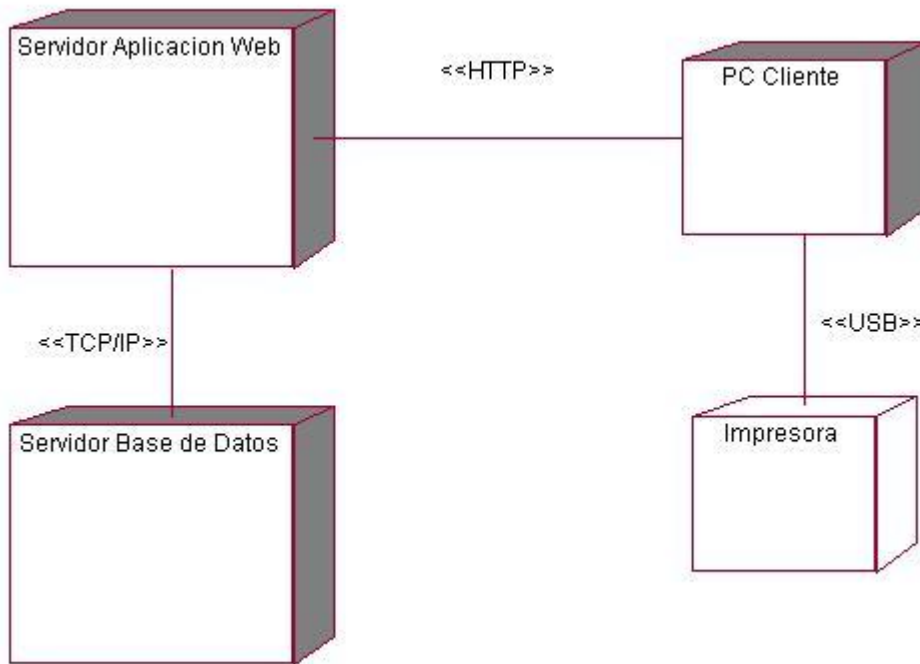


Figura 23 Diagrama de Despliegue

CONSTRCCIÓN DE LA SOLUCIÓN PROPUESTA

3.4.2 Modelo de Implementación

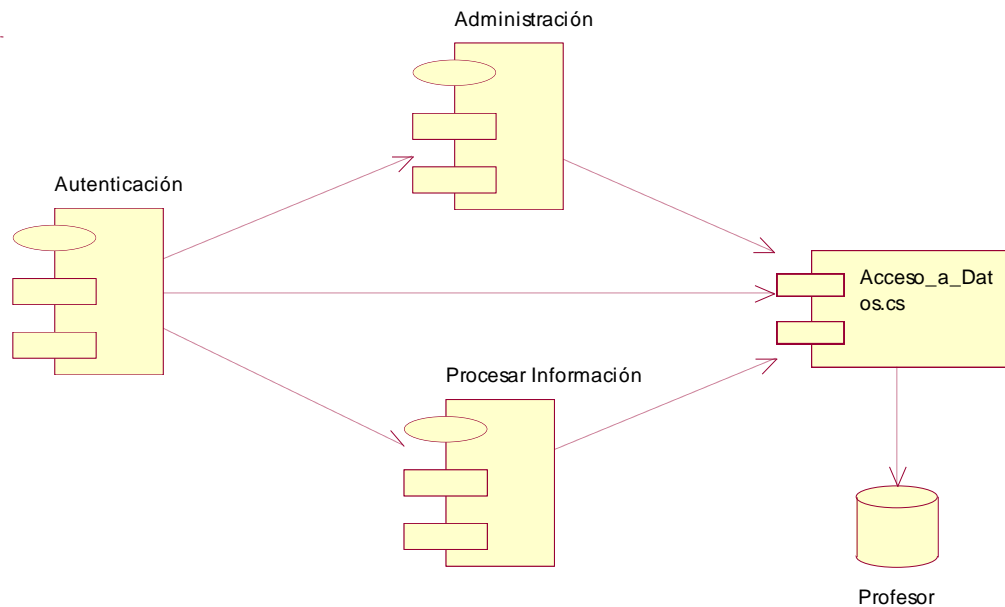


Figura 24 Diagrama de Componentes

3.5 Prueba del sistema propuesto

Prueba de caja blanca

Con la prueba de la caja blanca del software se comprueba los caminos lógicos del software proponiendo casos de prueba que se ejerciten conjuntos específicos de condiciones y/o bucles. Se puede examinar el estado del programa en varios puntos para determinar si el estado real coinciden con el esperado o mencionado.

CONSTRCCIÓN DE LA SOLUCIÓN PROPUESTA

```
public List<string> Seleccionar_Nombres_Profesor() 1
{
    using (SqlConnection _con = new SqlConnection(this.Conexion)) 2
    {
        using (SqlCommand _cmd = new SqlCommand("SELECT * from Profesor ", _con)) 3
        {
            List<string> Lista_Nombre_Profesor = new List<string>(); 4

            try 5
            {
                _con.Open(); 6
                using (SqlDataReader _reader = _cmd.ExecuteReader()) 7
                {
                    while (_reader.Read()) 8
                    {
                        Lista_Nombre_Profesor.Add(_reader["nombre"].ToString()); 9
                    }
                }
                _con.Close(); 10
            }
            catch (Exception exception) 11
            {
                throw exception; 12
            }
            return Lista_Nombre_Profesor; 13
        }
    } 14
}
```

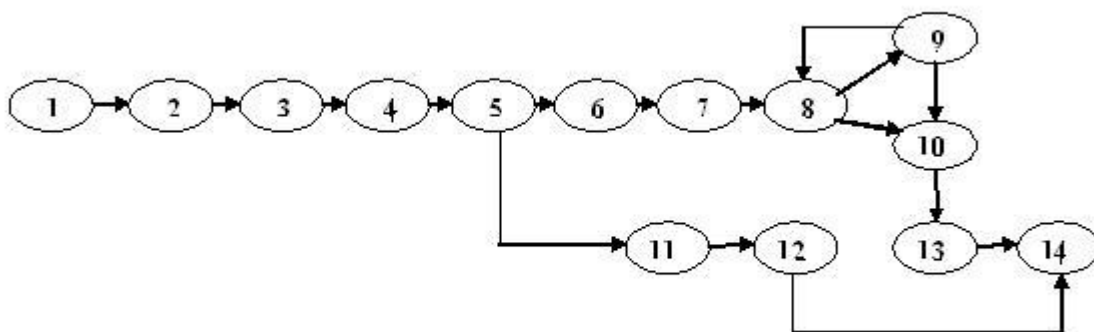


Figura 25 Complejidad Ciclomática

CONSTRCCIÓN DE LA SOLUCIÓN PROPUESTA

Complejidad ciclomática $V(G) = \text{Cantidad de Aristas [A]} - \text{Cantidad de nodos [N]} + 2$.

$$V(G) = A - N + 2$$

$$V(G) = 16 - 14 + 2$$

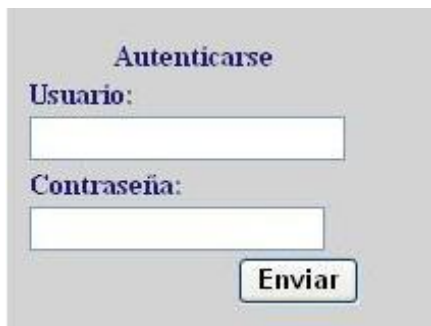
$$V(G) = 4$$

Prueba de caja negra

La prueba de caja negra se refiere a las pruebas que se llevan a cabo sobre la interfaz del software. O sea, los casos de prueba pretenden demostrar que las funciones del software son operativas, que la entrada se acepta de forma adecuada y que se produce un resultado correcto, así como que la integridad de la información externa se mantiene.

Caso de uso: Autenticar

- 1- Usuario y Contraseña que no existan en la base de datos.
- 2- Usuario que no exista en la base de datos y Contraseña que si exista en la base de datos.
- 3- Usuario que exista en la base de datos y Contraseña que no exista en la base de datos.
- 4- Usuario y contraseña bien.



Autenticarse

Usuario:

Contraseña:

Enviar

Figura 26 Prueba de caja negra

CONSTRCCIÓN DE LA SOLUCIÓN PROPUESTA

Caso de Uso	Autenticar profesor
Caso de prueba	1
Entrada	Nombre de usuario="dddd" Contraseña="dddd"
Condiciones de ejecución	Como el usuario que existe en la base de datos es diana y la contraseña diana1*, los datos entrados no coinciden.
Resultado	El sistema muestra un mensaje informando al usuario que el nombre y la contraseña introducida no son correctos.

Tabla 15 Prueba de caja negra caso de prueba 1

Caso de Uso	Autenticar profesor
Caso de prueba	2
Entrada	Nombre de usuario="dddd" Contraseña="diana1*"
Condiciones de ejecución	Como el usuario que existe en la base de datos es diana y la contraseña diana1*, el nombre entrado no coincide.
Resultado	El sistema muestra un mensaje informando al usuario que su nombre no es correcto.

Tabla 16 Prueba de caja negra caso de prueba 2

Caso de Uso	Autenticar profesor
Caso de prueba	3
Entrada	Nombre de usuario="diana" Contraseña="dddd"
Condiciones de ejecución	Como el usuario que existe en la base de datos es diana y la contraseña diana1*, la contraseña entrada no coincide.
Resultado	El sistema muestra un mensaje informando al usuario que su contraseña no es correcta.

Tabla 17 Prueba de caja negra caso de prueba 3

CONSTRCCIÓN DE LA SOLUCIÓN PROPUESTA

Caso de Uso	Autenticar profesor
Caso de prueba	4
Entrada	Nombre de usuario="diana" Contraseña="diana1**"
Condiciones de ejecución	Como el usuario que existe en la base de datos es diana y la contraseña diana1*, los datos entrados son válidos.
Resultado	El sistema muestra un mensaje informando al usuario que sus datos son correctos y permite su acceso.

Tabla 18 Prueba de caja negra caso de prueba 4

Conclusiones

Como resultado del estudio realizado en este capítulo, correspondiente a la etapa de diseño e implementación del sistema, se modelaron los diagramas de clases para cada uno de los casos de uso del sistema. Se obtuvo el diagrama de clases persistentes y el modelo de datos. Se establecieron las pautas para el diseño de la interfaz. Se explicó como debe ser llevado a cabo el tratamiento de excepciones. Finalmente se muestra el diagrama de despliegue, todo esto antes mencionado se logró con el correcto uso de los artefactos correspondientes que nos brinda el Rational Rose.

Capítulo 4. Estudio de Factibilidad



Introducción

El estudio de la factibilidad de un proyecto es muy importante en las decisiones que se tomarán en el futuro por lo que se debe hacer lo suficientemente preciso para que no haya errores que impliquen demoras y gastos innecesarios en personal y tiempo. En este capítulo se presenta el estudio de factibilidad de la aplicación con el objetivo de obtener el cálculo de esfuerzo, tiempo de desarrollo, cantidad de hombres y costo del proyecto, así como un análisis de costo – beneficio. El estudio de factibilidad se lleva a cabo por el método de estimación basado en las funciones y dentro de este, los puntos de casos de uso.

4.1 Planificación

La planificación de proyectos es un proceso imprescindible para lograr alcanzar la máxima calidad del mismo. Estimar los resultados del proyecto y los valores de costo, tiempo y recursos requeridos constituyen el punto en que radica su importancia, pues se puede determinar con anterioridad si el proyecto a realizar que se propone, es costeable, con el objetivo de evitar las pérdidas de recursos que afectan la economía de la entidad donde se lleva a cabo. La planificación, es sólo una idea de como van a transcurrir las cosas. Hay que planificar el trabajo, los recursos humanos y la tecnología.

Objetivos de la planificación.

Su propósito es establecer planes razonables para desarrollar la Ingeniería de Software y manejar los cambios de los proyectos de Software. Incluye la actividad de estimar los resultados del proyecto y los valores de costo, tiempo y recursos requeridos. Establece los pasos necesarios y define el plan de desarrollo.

El objetivo de la planificación se logra mediante un procesamiento de la información que lleve a estimaciones razonables. Es por esto que se requiere de:

ESTUDIO DE FACTIBILIDAD

1. Experiencia, lógica, sentido común: casos anteriores, proyectos similares, etc.
2. Técnicas de estimación: a partir de determinados parámetros se predeterminan variables a estimar como el costo, el esfuerzo y el tiempo necesarios para obtener el software. Donde el esfuerzo se traduce al total de tiempo que gasta una persona trabajando en el desarrollo del proyecto de software (horas persona / mes persona)

Existen 3 variantes para la estimación.

1. Análisis de Puntos de Casos de Uso
2. Análisis de Puntos de Función y COCOMO II
3. Las estimaciones por líneas de código (LDC)

Este trabajo se basó en el Análisis de Puntos de Casos de Uso en el que existe la posibilidad de predecir el tamaño de un sistema a partir de las características de sus requisitos, expresados en los casos de uso. Además se trata de un método de estimación del tiempo de desarrollo de un proyecto mediante la asignación de "pesos" a un cierto número de factores que lo afectan, para finalmente, contabilizar el tiempo total estimado para el proyecto a partir de esos factores.

4.2 Costos

1. Cálculo de Puntos de Casos de Uso sin ajustar.

Se calcula a partir de la siguiente ecuación:

$$\text{UUCP} = \text{UAW} + \text{UUCW}$$

UUCP: Puntos de Casos de Uso sin ajustar

UAW: Factor de Peso de los Actores sin ajustar

UUCW: Factor de Peso de los Casos de Uso sin ajustar

Para calcular UAW

ESTUDIO DE FACTIBILIDAD

Tipo	Descripción	Peso	Cant * peso
Simple	Otro sistema que interactúa con el sistema a desarrollar mediante una interfaz de programación (API, Application Programming Interface)	1	0*1
Medio	Otro sistema que interactúa con el sistema a desarrollar mediante un protocolo o una interfaz basada en texto	2	0*2
Complejo	Una persona que interactúa con el sistema mediante una interfaz gráfica	3	4*3
Total			12

Tabla 19 Factor de Peso de los Actores sin ajustar

- Para calcular UUCW

Tipo	Descripción	Peso	Cant * peso
Simple	El Caso de Uso contiene de 1 a 3 transacciones	5	2*5
Medio	El Caso de Uso contiene de 4 a 7 transacciones	10	4*10
Complejo	El Caso de Uso contiene más de 8 transacciones	15	1*15
Total			65

Tabla 20 Factor de Peso de los Casos de Uso sin ajustar

Luego: $UUCP = 12 + 65$

$UUCP = 77$

2. Ajustar los Puntos de casos de uso

$$UCP = UUCP * TCF * EF$$

ESTUDIO DE FACTIBILIDAD

Para Calcular TCF

Factor	Descripción	Peso	Valor	$\Sigma (\text{Peso}_i * \text{Valor}_i)$
T1	Sistema distribuido	2	0	0
T2	Objetivos de performance o tiempo de respuesta	1	4	4
T3	Eficiencia del usuario final	1	4	4
T4	Procesamiento interno complejo	1	2	2
T5	El código debe ser reutilizable	1	4	4
T6	Facilidad de instalación	0.5	4	2
T7	Facilidad de uso	0.5	4	2
T8	Portabilidad	2	4	8
T9	Facilidad de cambio	1	4	4
T10	Concurrencia	1	4	4
T11	Incluye objetivos especiales de seguridad	1	5	5
T12	Provee acceso directo a terceras partes	1	0	0
T13	Se requieren facilidades especiales de entrenamiento a los usuarios	1	0	0
Total	39			

Tabla 21 Factor de complejidad técnica

$$\text{TCF} = 0.6 + 0.01 * 39$$

$$\text{TCF} = 0.99$$

Para calcular EF

$$\text{EF} = 1.4 - 0.03 * \Sigma (\text{Peso}_i * \text{Valor}_i) \text{ (Donde Valor es un número del 0 al 5)}$$

ESTUDIO DE FACTIBILIDAD

Factor	Descripción	Peso	Valor	Σ (Peso _i * Valor _i)
E1	Familiaridad con el modelo de proyecto utilizado	1.5	3	4.5
E2	Experiencia en la aplicación	0.5	4	2
E3	Experiencia en orientación a objetos	1	3	3
E4	Capacidad del analista líder	0.5	1	0.5
E5	Motivación	1	5	5
E6	Estabilidad de los requerimientos	2	3	6
E7	Personal part-time	-1	2	- 2
E8	Dificultad del lenguaje de programación	-1	3	-3
Total				16.0

Tabla 22 Factor de ambiente

$$EF = 1.4 - 0.03 * 16.0$$

$$EF = 0.92$$

Luego $UCP = 77 * 0.99 * 0.92$

$$UCP = 70.13$$

3. Calcular esfuerzo de FT Implementación

$$E = UCP * CF$$

ESTUDIO DE FACTIBILIDAD

Para calcular CF

Como $\text{Total}_{EF} = 1$

$CF = 20$ horas-hombre (porque $\text{Total}_{EF} \leq 2$)

Luego $E = 70.13 * 20$ horas-hombre

$E = 1402.6$ horas-hombre

4. Calcular esfuerzo de todo el proyecto

Actividad	% esfuerzo	Valor esfuerzo
Análisis	10%	350.65 horas-hombre
Diseño	20%	701.3 horas-hombre
Implementación	40%	1402.6 horas-hombre
Prueba	15%	525.98 horas-hombre
Sobrecarga	15%	525.98 horas-hombre
Total	100%	3506.5horas-hombre

Tabla 23 Relación Actividad –Porcentaje

El esfuerzo total (E_T) = 3506.5 horas-hombre.

Suponiendo que una persona trabaje 8 horas por día, y un mes tiene como promedio 30 días; la cantidad de horas que puede trabajar una persona en 1 mes es 240 horas.

Si $E_T = 3506.5$ horas-hombre y por cada 240 horas yo tengo 1 mes eso daría un $E_T = 14.65$ mes-hombre.

Costo del proyecto

- CHM: Costo Hombre Mes.
- ET: Esfuerzo total (mes-hombre)
- SBM: Salario básico mensual

Se asume un SBM de \$100.00.

$CHM = 2 \times SBM$

$CHM = 200.00$ pesos/mes

ESTUDIO DE FACTIBILIDAD

Costo = CHM x ET

Costo = 200 x 7

Costo = \$ 1400

4.3 Beneficios tangibles e intangibles

4.3.1 Beneficios tangibles

El beneficio fundamental del sistema es contar con una aplicación Web flexible y dinámica que permita realizar la gestión de los profesores manteniendo actualizada la información de los mismos y dar a conocer de forma más precisa otros datos de interés de los profesores desde que inician su vida en la facultad. Los profesores u otros Directivos de la Facultad pueden acceder directamente desde cualquier punto de la UCI, esto trae como consecuencia un ahorro de materiales de oficina tales como papel, tonel para la impresión, etc.

4.3.2 Beneficios Intangibles

Con la elaboración de este Sistema de Control para la Gestión de Información de Profesores para la Facultad 9 de la Universidad de las Ciencias Informáticas (UCI), se facilitará el proceso de gestión de información de profesores en la facultad, pues el mismo permitirá a los profesores acceder directamente a la aplicación desde cualquier computadora de la universidad que se halle conectada a la red. Por otra parte aliviará el trabajo manual que efectúan los implicados en esta labor, ya que con este sistema automatizado se dará solución a una de las tareas más prioritarias de la Facultad.

La información que manipula el sistema, se encuentra actualizada, lo que permitirá llevar un control de toda la gestión de los profesores, aspecto importante para la toma de decisiones en aras de lograr la eficiencia de la facultad en este aspecto. De esta forma se simplificará el trabajo de los implicados en este proceso y se creará una nueva y rápida vía de gestión y actualización de información.

ESTUDIO DE FACTIBILIDAD

4.4 Análisis de costos y beneficios

La gestión de profesores en una universidad de nuevo tipo y tan grande como lo es la UCI, es un pilar fundamental en el desarrollo del centro, por lo que se necesita del uso de una herramienta que automatice este proceso desde la base de las facultades. Luego de analizar los beneficios tangibles e intangibles adquiridos por el sistema que se propone, se considera viable el desarrollo del sistema. De modo que el directivo de la Facultad 9 contará con una herramienta poderosa para el manejo de información, que disminuirá los gastos en tiempo y esfuerzo.

Conclusiones

El estudio de factibilidad realizado en este capítulo nos lleva a concluir que el desarrollo de este proyecto incurriría en un esfuerzo de 3506.5 horas-hombre, con un tiempo de desarrollo de 7 meses aproximadamente para dos personas y un costo de \$1400 considerando un salario mensual de \$100 para dos personas.

Teniendo en cuenta un análisis de los costos en contraposición con los beneficios que reportará el desarrollo de la aplicación, se concluye que es factible desarrollar el producto y que por tanto se debe proceder con la implementación pues el sistema es muy valioso para la gestión de información de profesores en la facultad 9.

CONCLUSIONES



La presente investigación desarrollada para la Facultad 9 nos conlleva a la conclusión de que:

La implantación de un sistema para el control de información del profesor en la Facultad 9 a través de una aplicación Web facilita la gestión de la información del profesor, pues la misma reduce el tiempo de respuesta a los usuarios y los errores que pueden cometer, mejorando la calidad del trabajo de los involucrados en esta labor.

El uso de esta aplicación no solo repercute en la perfección del proceso de Gestión de Profesores en la facultad sino que le permite al propio profesor conocer sus resultados en las disímiles actividades, conjuntamente de otros datos de interés.

La modelación del sistema se basó en el Proceso Unificado de Desarrollo de Software a través de un grupo de artefactos para cada uno de los flujos de trabajo.

El estudio de la factibilidad determinó que el desarrollo de la aplicación es realmente factible, siendo positiva la implantación de la misma en la facultad 9 de la Universidad de las Ciencias Informáticas de Cuba.

RECOMENDACIONES



- Llevar este sistema a todas las facultades de la Universidad de las Ciencias Informáticas.
- Mantener sobre el sistema un estricto cumplimiento del proceso de mantenimiento y actualización periódica, logrando así que se mantenga la fiabilidad y funcionamiento óptimo del sistema y de la información que se gestiona en el mismo.
- Elaborar un manual del usuario para el sistema y hacerlo extensivo a todos los relacionados con el sistema.
- Continuar con la investigación para garantizar mejoras en futuras versiones del sistema.
- Analizar las sugerencias de los profesores y directivos en cuanto al sistema.
- Trabajar en un futuro en una capa de seguridad.

REFERENCIAS BIBLIOGRÁFICAS



- [1] ALHAMA, F. B. Nuevas Formas Organizativas. Instituto de Estudios e Investigaciones del Trabajo ed. La Habana 2004.
- [2] AUTORES, C. D. Akademos. Gestión Académica [Consultado el: 15/01 de 2007]. Disponible en: <http://akademos/akademos/default.aspx>.
- [3] Elementos básicos del lenguaje C#. En Introducción a la Tecnología .NET. 2004-2005
- [4] Desarrollo de aplicaciones por capas. En Introducción a la programación en el servidor. Plataformas de desarrollo. . 2004-2005.
- [5] Aplicaciones Web. En Programación cliente-servidor. 2004-2005.
- [6] Modelo Entidad-Relación (ER). En Introducción a las Bases de Datos. 2005-2006. p. 18.
- [7] ARMANDO, C. S. Tecnología de Gestión de Recursos Humanos. Academia ed. La Habana: 1999.
- [8] EKKA. Tecnología a su alcance. Aplicaciones Web [Consultado el: 22/01 de 2007]. Disponible en: <http://www.ekka.cl/aplicaciones.htm>.
- [9] YABOR, G. S. O. Y. L. D. L. V. Los Sistemas de Control de Gestión Estratégica para las organizaciones [Consultado el: 5/12 de 2006]. Disponible en: www.monografias.com/trabajos15/sistemas-control/sistemas-control.shtml.
- [10] IVARJACOBSON, G. B., JAMES RUMBAUGH. El proceso unificado de desarrollo de software. Pearson Educación S.A ed. 2000.
- [11] CAMALLEA, N. L. N. Gestión de Base de Dato con ADO.Net. Científico-Técnico ed. Ciudad de La Habana: 2004.
- [12] LARMAN, C. UML y Patrones. Félix Varela ed. La Habana: 2004, vol. # 1
- [13] CHIRINO, A. L. Cases. Giga, 1997, vol. # 4
- [14] DONALD, M. M. ASP.Net Manual de Referencia. Mc Graw Hill ed. Madrid: 2002.
- [15] CORPORATION, M. Así es Microsoft Visual Studio .Net. Mc Graw Hill ed. Madrid: 2001.
- [16] MIGUEL CASTAÑO ADORACIÓN, M. G. P. V. Fundamentos y modelos de Bases de Datos. Félix Varela ed. Ciudad de La Habana: 2003.
- [17] Tabla de características de Visual Studio.NET [Consultado el: 28/01 de 2007]. Disponible en: <http://www.danysoft.com/free/vstudio.pdf>

[18] ¿Qué es ASP.NET? [Consultado el: 29/01 de 2007]. Disponible en:

<http://es.gotdotnet.com/quickstart/aspplus/doc/whatisaspx.aspx>

[19] Ventajas del Modelo C/S a las Organizaciones [Consultado el: 15/02 de 2007]. Disponible en:

<http://www.inei.gob.pe/biblioineipub/bancopub/inf/Lib5038/ven2.htm>

[20] Microsoft SQL Server, 1/02/07 [Disponible en: http://es.wikipedia.org/wiki/Microsoft_SQL_Server]

[21] WIKIPEDIA. Aplicación informática. Disponible en: http://es.wikipedia.org/wiki/Aplicaci3n_inform3tica

[22] EndNote: gestor de referencias bibliográficas [Consultado el: 3/3 de 2007]. Disponible en:

<http://www.ubu.es/biblioteca/servicios/endnote/index.htm>

[22] AUTORES, C. D. EndNote: gestor de referencias bibliográficas. [Consultado el: 3/3 de 2007].

Disponible en: <http://www.ubu.es/biblioteca/servicios/endnote/index.htm>

BIBLIOGRAFÍA



- KALEN, D. Y. A fondo Microsoft SQL Server 2000. McGraw-Hill Interamericana de España ed. Madrid: 2001. 904 p.
- ROZIC, S. E. Bases de Datos y su aplicación con SQL. MP Ediciones ed. Buenos Aires: 2004. 145 p.
- SOFTEL, G. B. SQL Server. Giga, 1996, vol. # 4
- LARMAN, C. UML y Patrones. Félix Varela ed. La Habana: 2004, vol. # 2
- IVARJACOBSON, G. B., JAMES RUMBAUGH. El proceso unificado de desarrollo de software. Pearson Educación S.A ed. 2000.

ANEXOS



Anexos 1 Prototipo de Interfaz Autenticar

SIP
Fac 9

Sistema de Información de Profesores


[Inicio](#) | [Mision](#) | [Contactenos](#) | [Intranet](#)

Autenticarse

Usuario:

Contraseña:

Anexos 2 Prototipo de Interfaz Registrar Profesor



[Principal](#) | [Misión](#) | [Contáctenos](#) | [Intranet](#)

[Cerrar sesión](#)
Bienvenido
profe

[Cambiar Contraseña](#)

☐ Gestionar Profesor
 Registrar Profesor
 Modificar Profesor
 Buscar Profesor
 Buscar Actividad
 Buscar Asistencia

Datos Personales del Profesor

CI	<input type="text"/>	No Solapín	<input type="text"/>
Nombre	<input type="text"/>	Apellidos	<input type="text"/>
Provincia	<input type="text" value="v"/>	Municipio	<input type="text" value="v"/>
Dirección	<input type="text"/>	Teléfono	<input type="text"/>
Estado Civil	<input type="text" value="v"/>	Cantidad de Hijos	<input type="text" value="v"/>

[Siguiente](#)

[Cerrar sesión](#)

Bienvenido
profe

[Cambiar Contraseña](#)

- ☐ Gestionar Profesor
 - Registrar Profesor
 - Modificar Profesor
 - Buscar Profesor
 - Buscar Actividad
 - Buscar Asistencia

Datos Laborales

Especialidad Graduado	<input type="text"/>	Año Graduado	<input type="text"/>
Universidad Graduado	<input type="text"/>	Categoría Docente	<input type="text" value="v"/>
Departamento	<input type="text" value="v"/>	Disciplina	<input type="text" value="v"/>
Asignatura	<input type="text" value="v"/>	Grupos Docencia	<input type="text"/>
Grupo Guía	<input type="text"/>	Años Experiencia	<input type="text" value="v"/>
Cargo	<input type="text"/>	Cursos Posgrados	<input type="text"/>
Nombre Maestría	<input type="text"/>	Universidad Maestría	<input type="text"/>
Nombre Doctorado	<input type="text"/>	Universidad Doctorado	<input type="text"/>
Nombre Proyecto	<input type="text"/>	Rol Proyecto	<input type="text"/>
Laboratorio Proyecto	<input type="text"/>	Teléfono Proyecto	<input type="text"/>
Edificio Beca	<input type="text"/>	Apto Beca	<input type="text"/>
Teléfono Beca	<input type="text"/>	<input type="radio"/> Externo <input type="radio"/> Interno	

[Anterior](#) [Siguiete](#)

Bienvenido
profe

[Cambiar Contraseña](#)

- ☐ Gestionar Profesor
 - Registrar Profesor
 - Modificar Profesor
 - Buscar Profesor
 - Buscar Actividad
 - Buscar Asistencia

Datos Partidistas

Sanciones Aplicadas Motivo

Medallas Recibidas Misiones Internacionalistas

País Misión Ubicación Defensa

Cargo Defensa Lugar

Salario Antigüedad

Fecha PCC Fecha Sanción Fecha Misión

≤ junio de 2007 ≥

lu	ma	mi	ju	vi	sá	do
28	29	30	31	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	1
2	3	4	5	6	7	8

≤ junio de 2007 ≥

lu	ma	mi	ju	vi	sá	do
28	29	30	31	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	1
2	3	4	5	6	7	8

≤ junio de 2007 ≥

lu	ma	mi	ju	vi	sá	do
28	29	30	31	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	1
2	3	4	5	6	7	8

[Anterior](#)

Anexos 3 Prototipo de Interfaz Modificar Profesor

Principal | Misión | Contáctenos | Intranet

[Cerrar sesión](#)
Bienvenido
profe

[Cambiar Contraseña](#)

☐ Gestionar Profesor
 Registrar Profesor
 Modificar Profesor
 Buscar Profesor
 Buscar Actividad
 Buscar Asistencia

Modificar Profesor


Modificar Datos Personales ▾

Solapin


CI	<input type="text" value="12234534534"/>	No Solapin	<input type="text" value="345"/>
Nombre	<input type="text" value="pepe"/>	Apellidos	<input type="text" value="dfid"/>
Provincia	<input type="text" value="Matanzas"/>	Municipio	<input type="text" value="Cárdenas"/>
Dirección	<input type="text" value="fonseca"/>	Teléfono	<input type="text" value="3423"/>
Estado Civil	<input type="text" value="Casado"/>	Cantidad de Hijos	<input type="text" value="5"/>

Bienvenido al Sistema de Información de Profesores

Anexos 4 Prototipo de Interfaz Eliminar Profesor



Sistema de Información de Profesores



[Principal](#) | [Misión](#) | [Contáctenos](#) | [Intranet](#)

[Cerrar sesión](#)
Bienvenido jefe departamento
[Cambiar](#)
[Contraseña](#)

- ☐ Gestionar Profesor
 - Registrar Profesor
 - Modificar Profesor
 - Eliminar Profesor
 - Buscar Profesor
- ☐ Gestionar Actividad
 - Registrar Actividad
 - Modificar Actividad
 - Eliminar Actividad
 - Buscar Actividad
- ☐ Gestionar Asistencia
 - Registrar Asistencia
 - Eliminar Asistencia
 - Buscar Asistencia

Eliminar Profesor


Departamento

	No Solapín	Nombre	Apellidos
Eliminar	345	pepe	dfd
Eliminar	3545	sdf	dsfaf
Eliminar	12687	Carlos	Torres Pupo
Eliminar	6589	fdg	ghfgh
Eliminar			

Microsoft Internet Explorer

¿Está seguro que desea eliminar el profesor?

Anexos 5 Prototipo de Interfaz Registrar Asistencia



SIP Fac 9 *Sistema de Información de Profesores*

Principal | Misión | Contáctenos | Intranet

[Cerrar sesión](#)

Bienvenido
jefe departamento

[Cambiar Contraseña](#)


- ☐ Gestionar Profesor
 - Registrar Profesor
 - Modificar Profesor
 - Buscar Profesor
- ☐ Gestionar Actividad
 - Registrar Actividad
 - Modificar Actividad
 - Eliminar Actividad
 - Buscar Actividad
- ☐ Gestionar Asistencia
 - Registrar Asistencia
 - Eliminar Asistencia
 - Buscar Asistencia

Registrar Asistencia


Departamento Actividad

Profesor Asistencia

Anexos 6 Prototipo de Interfaz Eliminar Asistencia



Sistema de Información de Profesores



Principal | Misión | Contáctenos | Intranet

[Cerrar sesión](#)

Bienvenido jefe departamento

[Cambiar Contraseña](#)

- ☐ Gestionar Profesor
 - Registrar Profesor
 - Modificar Profesor
 - Buscar Profesor
- ☐ Gestionar Actividad
 - Registrar Actividad
 - Modificar Actividad
 - Eliminar Actividad
 - Buscar Actividad
- ☐ Gestionar Asistencia
 - Registrar Asistencia
 - Eliminar Asistencia
 - Buscar Asistencia

Eliminar Asistencia

	ID Asistencia	Departamento	Nombre Actividad	Nombre Profesor	Apellidos Profesor
Eliminar	63	especialidad	bsgf	pepe	dfd

Microsoft Internet Explorer

¿Está seguro que desea eliminar la asistencia?

Anexos 7 Prototipo de Interfaz Buscar Profesor



Sistema de Información de Profesores



[Principal](#) | [Misión](#) | [Contáctenos](#) | [Intranet](#)

[Cerrar sesión](#)
Bienvenido
profe

[Cambiar Contraseña](#)

Gestionar Profesor

- Registrar Profesor
- Modificar Profesor
- Buscar Profesor
- Buscar Actividad
- Buscar Asistencia


Buscar Profesor

Provincia	Seleccione ▼	Municipio	Seleccione ▼
Departamento	Seleccione ▼	Disciplina	Seleccione ▼
Asignatura	Seleccione ▼	Categoría Docente	Seleccione ▼
Proyecto	Seleccione ▼	Rol	Seleccione ▼
Tipo persona	Seleccione ▼		

<input type="text"/>	<input type="text"/>
nombre	apellidos



Anexos 8 Prototipo de Interfaz Buscar Asistencia



SIP
Fac 9

Sistema de Información de Profesores

Principal | Misión | Contáctenos | Intranet

[Cerrar sesión](#)
Bienvenido
profe


[Cambiar Contraseña](#)

- ☐ Gestionar Profesor
 - Registrar Profesor
 - Modificar Profesor
 - Buscar Profesor
 - Buscar Actividad
 - Buscar Asistencia

Buscar Asistencia

Departamento Actividad

Anexos 9 Prototipo de Interfaz Buscar Actividad



Principal | Misión | Contáctenos | Intranet

[Cerrar sesión](#)
Bienvenido
profe

[Cambiar Contraseña](#)

☐ Gestionar Profesor
 Registrar Profesor
 Modificar Profesor
 Buscar Profesor
 Buscar Actividad
 Buscar Asistencia

Buscar Actividad

Nombre: Departamento:

Fecha: Lugar:

Responsable:

Anexos 10 Prototipo de Interfaz Registrar Actividad



Sistema de Información de Profesores



Principal | Misión | Contáctenos | Intranet

[Cerrar sesión](#)

Bienvenido jefe departamento

[Cambiar Contraseña](#)

- ☐ Gestionar Profesor
 - Registrar Profesor
 - Modificar Profesor
 - Buscar Profesor
- ☐ Gestionar Actividad
 - Registrar Actividad
 - Modificar Actividad
 - Eliminar Actividad
 - Buscar Actividad
- ☐ Gestionar Asistencia
 - Registrar Asistencia
 - Eliminar Asistencia
 - Buscar Asistencia

Registrar Actividad

Nombre de la Actividad

Departamento

Responsable

Lugar

Hora

Fecha

≤ junio de 2007 ≥						
lu	ma	mi	ju	vi	sá	do
28	29	30	31	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	1
2	3	4	5	6	7	8

Anexos 11 Prototipo de Interfaz Modificar Actividad

Principal | Mision | Contáctenos | Intranet

[Cerrar sesión](#)

Bienvenido
jefe departamento

[Cambiar Contraseña](#)

- [-] Gestionar Profesor
 - Registrar Profesor
 - Modificar Profesor
 - Buscar Profesor
- [-] Gestionar Actividad
 - Registrar Actividad
 - Modificar Actividad
 - Eliminar Actividad
 - Buscar Actividad
- [-] Gestionar Asistencia
 - Registrar Asistencia
 - Eliminar Asistencia
 - Buscar Asistencia

Modificar/Eliminar Actividad

	Nombre	Departamento	Fecha	Lugar	Hora	Responsable
Editar	Eliminar		Mar 7		Ene 1 900	fgdfg
Editar	Eliminar				15AM	
Editar	Eliminar				ene 1 900	b2
Editar	Eliminar				:00PM	
Editar	Eliminar				ene 1 900	ff
			12:00AM		1:10AM	
Editar	Eliminar	sdfg	Ago 7 2001	dgf	Ene 1 1900	fg
			12:00AM		8:24AM	
Editar	Eliminar	fg	May 4 2003	dsfg	Ene 1 1900	sdfgf
			12:00AM		8:20AM	
Editar	Eliminar	vf	Ago 7 2005	csdf	Ene 1 1900	dfg
			12:00AM		1:40AM	

Microsoft Internet Explorer

¿Está seguro que desea eliminar la actividad?

Anexos 12 Prototipo de Interfaz Eliminar Actividad

de Profesores
Principal | Mision | Contáctenos | Intranet

[Cerrar sesión](#)

Bienvenido jefe departamento

[Cambiar Contraseña](#)

- Gestionar Profesor
 - Registrar Profesor
 - Modificar Profesor
 - Buscar Profesor
- Gestionar Actividad
 - Registrar Actividad
 - Modificar Actividad
 - Eliminar Actividad
 - Buscar Actividad
- Gestionar Asistencia
 - Registrar Asistencia
 - Eliminar Asistencia
 - Buscar Asistencia

Modificar/Eliminar Actividad

	Nombre	Departamento	Fecha	Lugar	Hora	Responsable
Editar	Eliminar		Mar 7		Ene 1 900 :15AM	fgdfg
Editar	Eliminar				ne 1 900 :00PM	b2
Editar	Eliminar				ne 1 900	ff
				12:00AM	1:10AM	
Editar	Eliminar	sdfg	Ago 7 2001	dgf	Ene 1 1900	fg
				12:00AM	8:24AM	
Editar	Eliminar	fg	May 4 2003	dsfg	Ene 1 1900	sdfgf
				12:00AM	8:20AM	
Editar	Eliminar	vf	Ago 7 2005	csdf	Ene 1 1900	dfg
				12:00AM	1:40AM	

javascript: __doPostBack('ctl00\$ContentPlaceHolder1\$GridView2\$ctl03\$LinkButton2', '')

Intranet local

Anexos 13 Prototipo de Interfaz Registrar Usuario

Principal | Mision | Contáctenos | Intranet

[Cerrar sesión](#)

Bienvenido administrador

[Cambiar Contraseña](#)

Gestionar Usuario

Gestionar Usuario

Registrar Usuario ▾

Administrador
 Directivos
 Jefe Departamento
 Profesor

Regístrese para obtener una nueva cuenta

Nombre de usuario:

Contraseña:

Confirmar contraseña:

Correo electrónico:

Pregunta de seguridad:

Respuesta de seguridad:

Crear usuario

Anexos 14 Prototipo de Interfaz Modificar Usuario



Sistema de Información de Profesores

Principal | Misión | Contáctenos | Intranet

[Cerrar sesión](#)

Bienvenido administrador

[Cambiar Contraseña](#)

Gestionar Usuario

Gestionar Usuario

Modificar y elimi ▼

Nombre de usuario	Dirección de correo		
administrador	admon@uci.cu	Editar	Eliminar
bs	u@uci.cu	Editar	Eliminar
diana	dp@uci.cu	Editar	Eliminar
jefe departamento	jefe@uci.cu	Editar	Eliminar
profe	sef@uci.cu	Editar	Eliminar
profesor	p@uci.cu	Editar	Eliminar
yuliet	ye@uci.cu	Editar	Eliminar
yuly	y@uci.cu	Editar	Eliminar

Anexos 15 Prototipo de Interfaz Cambiar Contraseña



Cambiar Contraseña

Contraseña:

Nueva contraseña:

Confirmar la nueva contraseña:

GLOSARIO TÉRMINOS



ASP: Active Server Pages (**ASP**) es una tecnología del lado servidor de Microsoft para páginas Web generadas dinámicamente.

PNUD: Programa de las Naciones Unidas para el Desarrollo.

SQL (Structured Query Language): Lenguaje de consulta estructurado, en informática, un lenguaje utilizado en bases de datos para consultar, actualizar y manejar bases de datos relacionales. Conjunto estándar de comandos para gestionar bases de datos relacionales por sus mismas características relacionales.

SGBD: Es el software que permite la utilización y/o la actualización de los datos almacenados en una (o varias) base(s) de datos por uno o varios usuarios desde diferentes puntos de vista y a la vez.

Servidor: Una aplicación informática o programa que realiza algunas tareas en beneficio de otras aplicaciones llamadas clientes.

UML "Unified Modeling Lenguaje": Lenguaje gráfico que brinda un vocabulario y reglas para especificar, construir, visualizar y documentar los artefactos de un sistema.

Usuario: Es la persona que utiliza o trabaja con algún objeto o algún servicio público o privado, empresarial o profesional.

Base de Dato: Es un conjunto, colección o depósito de datos almacenados en un soporte informático no volátil.