Universidad de Cienfuegos “Carlos Rafael Rodríguez”
Facultad de Informática
Carrera de Ingeniería Informática

“Sitio Web de la UJC de la Universidad de Cienfuegos”

Trabajo de diploma para optar por el título de Ingeniería en Informática

Autor:
Surama Quesada Velasco

Tutor:
Msc. Oscar Luís Muñoz González.

Consultante:
Msc. Maikel Betancourt Dueñas.

Cienfuegos, Cuba
Curso 2006 – 2007

Declaración de autoría

Declaro que soy la única autora del trabajo de diploma titulado “Sitio Web de la UJC de la Universidad de Cienfuegos”, y autorizo al Secretario General de la Organización de la UJC en la Universidad de Cienfuegos y al Departamento de Informática de la Facultad de Informática en la Universidad de Cienfuegos “Carlos Rafael Rodríguez”, para que hagan el uso que estimen pertinente con mismo.

Para que así conste firmo la presente a los ____ días del mes de Junio del 2007.

_____

Firma del Autor

Surama Quesada Velasco

_____

Firma del Tutor

M. Sc. Oscar Luis Muñoz

Los abajo firmantes certificamos que el presente trabajo ha sido revisado según acuerdo de la dirección de nuestro centro y el mismo cumple los requisitos que debe tener un trabajo de esta envergadura referente a la temática señalada.

Firma Tutor

Firma ICT

Firma Vicedecano

Agradecimientos.

A mi papito lindo y a mi mamá por todo su apoyo durante toda la carrera.

A mi prima Cristo por saberme comprender en estos últimos años de estudio

A Rodny, quien me brindo su apoyo durante gran parte del estudio y la seguridad que me permitió ganar confianza en mí misma y salir adelante.

A mis compañeras de cuarto que me soportaron por tanto tiempo, me ayudaron y apoyaron cuando lo necesite.

A gran amiga Yailin por su amistad y por ayudarme a materializar este documento.

A Maylen por su amistad y ayuda en la elaboración del documento.

A todos mis profes por su ayuda, dedicación y amor, en especial a Daymarelis.

A mi querido tutor Oscar por ayudarme en la tesis y por enseñarme a programar años anteriores.

A mis compañeros de aula, con quienes compartí estos 5 años, los quiero y no me olvidare de ustedes.

Al decano por permitirme regresar de la CUJAE.

Dedicatoria

A mis Padres queridos por su amor y dedicación.

A mis abuelos Delia, Isabel y Erasmo con todo el amor del mundo.

A mi precioso hermanito Sander con deseo de que me dedique también su trabajo de diploma en un futuro.

A mi prima Cristinita con mucho amor.

A mis amigas y amigos que se han ganado un pedacito de mi corazón, los quiero mucho.

Resumen

La Unión de Jóvenes Comunistas (UJC) es la organización política de la juventud cubana y basa su empeño en formar íntegramente a cada uno de sus miembros y a todos aquellos que se encuentran en su radio de acción, así como la formación integral y multifacético de las nuevas generaciones. En la Universidad de Cienfuegos en todos los cursos se integran a nuestras filas numeroso estudiantes, y también se realizan traslados a los que culminan sus estudios, estas tareas desarrollan una mayor complejidad en el proceso de gestión de documentos importantes para el buen funcionamiento de la organización, la cual abarca un gran radio de extensión. Por tal razón y dada la seriedad e importancia de las actividades que se realizan, y el alto grado de comprometimiento de nuestros militantes, se hace necesaria la aplicación de nuevas tecnologías para un buen funcionamientote la UJC dentro del centro.

Este trabajo constituye una aplicación Web encargada de automatizar todos los procesos manuales que se realizan en la oficina de la UJC, facilitando la consulta de documentos y obtención de los diferentes reportes que aquí se manejan, además de una actualización rápida, confiable y eficiente para garantizar una mayor calidad en el control y organización de esta estructura, esto trae consigo el mejoramiento del trabajo en las filas de la UJC.

Para construir la aplicación Web se utiliza el lenguaje PHP, para el almacenamiento de la información el gestor de base de datos MySql y para el análisis y diseño de la aplicación, la metodología RUP.

Índice

Introducción 1
Capítulo 1 – Fundamentación teórica 8
1.1– Introducción 8
1.2– Breve historia de la UJC 8
1.3– La UJC en la “Universidad de Cienfuegos” 10
1.4 – Descripción del dominio del problema 11
1.5 – Descripción del objeto de estudio 13
1.5.1 - Objetivos estratégicos de la organización 13
1.5.2 - Flujo actual de los procesos y análisis crítico de la ejecución de estos. 16
1.6 – Descripción de los sistemas existentes 17
1.7 – Aplicaciones Web. 18
1.8 – Tendencias, metodologías y/o tecnologías actuales. 21
1.8.1 – Macromedia Dreamweaver MX 2004. 21
1.8.2 – UML 23
1.8.3 – Proceso Unificado de Desarrollo (RUP) 25
1.8.4 – Rational Rose 27
1.8.5 – PHP (Professional Home Pages) 28
1.9 – Conclusiones 34
Capítulo 2 – Construcción de la solución propuesta. 36
2.1 – Introducción 36
2.2 – Descripción del modelo de negocio 36
2.3 – Reglas del negocio a considerar 36
2.4 – Modelo de casos de uso del negocio 38
2.4.1 – Actores del negocio. 38
2.4.2 – Trabajadores del negocio. 39
2.4.3 – Diagramas de casos de uso del negocio 40
2.4.4 – Descripción de los casos de uso del negocio. 40
2.4.5 – Diagramas de actividades del negocio. 44
2.5 – Modelo de objetos del negocio. 47
2.6 – Descripción del sistema propuesto 47

2.6.1 – Concepción general del sistema 47
2.6.2- Requerimientos funcionales 48
2.6.3 – Requerimientos no funcionales 50
2.7 – Modelo de casos de uso del sistema 54
2.7.1 – Actores del sistema 54
2.7.2 – Paquetes y sus relaciones 54
2.7.3 – Diagramas de casos de uso del sistema 56
2.7.4 – Descripción de los casos de uso del sistema. 57
2.8 – Diagrama de clases del diseño Web. 76
2.8.1- Módulo Actualizar Datos. 77
2.8.2- Módulo Seguridad. 77
2.8.3- Módulo Reporte. 77
2.9 – Diseño de la base de datos 78
2.9.1 Diagrama de clases persistentes. 78
2.10 – Diagrama de implementación 78
2.11 – Principios de diseño 79
2.11.1 – Estándares en la interfaz de la aplicación 79
2.11.2 – Tratamiento de errores 81
2.11.3 – Concepción General de la ayuda 81
2.12 – Conclusiones 81
Capitulo 3 – Estudio de Factibilidad 83
3.1 Planificación por puntos de función. 83
3.2 – Determinación de los costos 89
3.4 Beneficios tangibles e inteligentes. 92
3.5 Análisis de costos y beneficios. 92
3.6 Conclusiones. 93
Conclusiones 94
Recomendaciones 95
Bibliografía 96
Referencias bibliográficas 97
Glosario de Términos 100

Anexos 101
Anexo 1 –Prototipos 101
Anexo 2 – Diagrama de clases del diseño Web. 102
Anexo 3 – Diagrama de clases del diseño Web. 102
Anexo 4 – Diagrama de clases del diseño Web 102
Anexo 5 –Modelo Físico de la Base de Datos 133
Anexo 6 –Modelo Lógico de la Base de Datos. 137

Introducción

La Unión de Jóvenes Comunistas (UJC) es el fruto inmediato de la unidad cimentada entre los jóvenes que combatieron a la tiranía desde las filas de las diferentes organizaciones revolucionarias Es resultado, sobre todo, de los que han realizado innumerables jornadas par la defensa de la Revolución: jóvenes "Cinco Picos", alfabetizadotes, milicianos, combatientes de Giran, de la lucha contra bandidos, columnistas del centenario, de macheteros, constructores, estudiantes, maestros, deportistas, científicos, médicos, combatientes y trabajadores internacionalistas, además de otros exponentes surgidos de la clase obrera y el pueblo **[1].**

Es la vanguardia de la juventud cubana, orgánicamente independiente y tiene un carácter selectivo, agrupando a los jóvenes más revolucionarios de todos los sectores de la sociedad. Se rige por los principios del centralismo democrático, la dirección colectiva y la responsabilidad individual, el ejercicio de la crítica y la autocrítica, la disciplina consciente y férrea, y encamina su labor a partir de una estrecha vinculación con las masas Juveniles, a las cuales se debe y representa.

Asume la formación comunista de las nuevas generaciones a través de las organizaciones de masa, sustentada en el patriotismo, la fidelidad al Partido Comunista de Cuba, la defensa de los más altos valores humanos y el espíritu profundamente antiimperialista e internacionalista que ha distinguido a la Revolución Cubana.

La organización está estructurada a lo largo de todo el país, y para dirigir los trabajos entre congreso y congreso cuenta con un Comité Nacional y un Buró Nacional, este último integrado por 26 personas. Su Primer Secretario es Julio Martínez. Mantiene relaciones con 218 organizaciones, de diferentes partes del mundo y esta afiliada a la Federación Mundial de Juventudes Democráticas: a su vez desarrolla vínculos con otras 14 organizaciones internacionales. En ella militan mas de 500 000 jóvenes y la proyección de su trabajo va más allá de sus miembros, está dirigida a toda la población joven del país. **[2]**

En la Universidad de Cienfuegos “Carlos Rafael Rodríguez” se estudian varias carreras, donde se preparan cerca de 1967 estudiantes de los diferentes municipios y de las provincias vecinas de Villa Clara y Sancti Spíritus y de ellos 789 son militantes.

El comité de base es la célula fundamental de la Organización, controla e impulsa la superación cultural y política de sus miembros, elige a sus dirigentes, proponen la admisión de nuevos militantes, aprueba la aplicación de sanciones internas, propone las externas, las bajas naturales y la desactivación, decide sobre el otorgamiento del aval a sus militantes, aprueba los traslados y las incorporaciones. Actualmente en todos los centros ya sean estudiantiles o de trabajo donde existen comités de base se hace muy tedioso y engorroso llevar el control de forma escrita de un gran cúmulo de información, referente a los datos de todos los militantes, de las tareas, actividades y reuniones que se realizan en esta organización, ya que para eso solo cuentan con un documento donde deben realizar todas las anotaciones, y la mayoría de las veces no se cuenta con el material suficiente para la realización de estos.

Por su gran poder organizativo y movilizativo, juega un papel fundamental en la actual situación del país, sus actividades se han incrementado considerablemente y han sido capaces de responder a todas, gracias al espíritu inculcado por nuestro Comandante en Jefe Fidel Castro Ruz. Por estas razones, cada vez se hace más difícil llevar el control de las actividades que se realizan y si a esto le sumamos las actividades que normalmente deben cumplir como organización y que son de vital importancia para el buen funcionamiento, es casi imposible archivar toda la información que se requiere en tiempo. La organización no puede detener sus actividades para dedicarse a escribir, se le hace necesaria la informatización de sus procesos, para así agilizar el trabajo estadístico y poder tener actualizados sus informes, ahorrando tiempo, logrando calidad y eficiencia en el trabajo que se realiza.

La ***situación problémica*** se debe a que en la oficina de la UJC se maneja un gran volumen de información.

Esta información esta organizada en archivos y estos a su vez por carpetas, cada una de ellas está separada según la facultad y el Comité de Base a la que pertenece. Cada carpeta tiene dentro el listado de los militantes, el pago de la cotización, el calendario de reuniones, la estructuración del Comité de Base, la asistencia a las actividades convocadas por la organización y las actas de las reuniones. Cada vez que el Secretario de la Universidad necesita saber los Comité de Base que no se han reunido o los que no han pagado en tiempo la cotización, tiene que revisar una a una cada carpeta para extraer los datos que necesita. Además, existe la posibilidad de pérdida de la información (puede extraviarse alguna carpeta y con ella todo su contenido) pues a pesar del estricto control que mantiene la organización de todos los datos de sus miembros y de todas sus actividades, es una realidad. En ciertas ocasiones ocurren pérdidas, omisiones o errores de información que pueden provocar serias dificultades a la hora de tomar decisiones importantes de la organización.

El creciente volumen de información exige archivos cada vez más amplios y seguros que permitan evitar el deterioro de toda documentación. Personal que es restringido, debido a que es información confidencial que solo les compete a los dirigentes de la organización.

Por otra parte, existe el Universo Juvenil: proceso de crecimiento a las fila y resulta engorroso realizar todo el proceso de alta de los militantes de forma manual.

Mensualmente el Comité de la UJC debe entregar un reporte al Comité Provincial de la organización, con todos los datos de las actividades que realiza normalmente. Si a esto se le suma la actividades que surgen por la actual situación del país, el cúmulo de información se hace aún mayor, para lo cual se necesita más personal y recursos. Se hace necesario crear las condiciones en el departamento para que la información pueda fluir desde la dirección de la universidad hasta los comités primarios y los comités de base.

A partir de estas dificultades y dada la seriedad de esta organización se requiere de un sistema informático confiable que automatice todos estos procesos, para la solución de los problemas antes planteados.

Los ***antecedentes*** del problema están basados en pequeños sistemas diseñados en Excel y Access, que agilizan un poco el proceso, pero no cumplen con las expectativas que se necesitan.

El ***objeto de estudio*** es el procesamiento de la información referente al funcionamiento de la Unión de Jóvenes Comunistas, conjuntamente al estudio de plataformas que accedan a su implementación. Por lo que el ***campo de acción*** es toda la información estadística y los distintos procesos relacionados con el funcionamiento de la UJC en la Universidad de Cienfuegos “Carlos Rafael Rodríguez”.

La ***idea*** a defender se basa en: Con la implantación de un software que gestione la información de la organización de la UJC, se facilitará la consulta de documentos y la obtención de diferentes reportes que agilicen y hagan más eficiente el manejo de los recursos.

El ***objetivo central*** del trabajo es desarrollar un software que facilite la gestión de la actualización de los comités de base de la universidad y garantice el acceso a la información con la calidad, confiabilidad y rapidez requerida para el departamento u otros usuarios.

Como ***objetivo específico*** se tiene:

1. Diseño de una aplicación informática de base de datos para la gestión de la información que se genera de los procesos.
2. Hacer el análisis, diseño e implementación del sistema.
3. Facilitar el control de los militantes, las reuniones, las actas y bajas, la cotización, el universo juvenil, las actas de las reuniones, etc.
4. Permitir hacer el cierre del mes de manera automática.

5. Centralizar la información para garantizar la integridad, solidez y seguridad de los datos.

Las ***tareas*** desarrolladas para cumplir con los objetivos son:

1. Estudiar y definir las características y funcionalidades de una aplicación Web.
2. Estudiar la forma conceptual y tecnológica de concebir y desarrollar una aplicación Web.
3. Realizar la búsqueda de la información necesaria que se pretende mostrar en la aplicación Web.
4. Analizar y definir los procesos que serán automatizados.
5. Realizar una selección de las metodologías, tendencias y tecnologías actuales para la solución del problema de acuerdo a las necesidades del centro.
6. Búsqueda y análisis de sistemas existentes y de propuestas no implementadas
7. Diseñar e implementar la base de Datos de acuerdo con lo que se pretende actualizar.
8. Realizar un análisis de la información que se pretende mostrar en la web.
9. Diseñar la interfaz Web en correspondencia con los requerimientos funcionales.
10. Implementar las funciones y procedimientos que controlaran la aplicación y la manipulación del mismo.
11. Elaborar una ayuda para el sistema.
12. Documentar el sistema.

El ***Aporte Práctico*** se materializa en la posibilidad que brinda el software para gestionar y controlar varios procesos de la UJC; facilitando la actualización eficiente de la información y la obtención de reportes.

Para realizar un buen análisis del trabajo de diploma y para una mejor comprensión del mismo este se encuentra ***estructurado por 3 capítulos*** y consta con un resumen, una introducción, conclusiones, recomendaciones, bibliografías y anexos. Los Capítulos se estructuran de la siguiente forma:

- Capítulo I._ "Fundamentación teórica":

  En este capítulo se presenta una breve historia de la Organización, así como el funcionamiento aquí en nuestro centro. Le sigue la "Descripción del dominio del problema", donde se expone los principales conceptos asociados que son necesarios para comprender el objeto de estudio. Mas adelante se encuentra La "Descripción del objeto de estudio", dentro de ella está los Objetivos estratégicos de la organización y el Flujo actual de los procesos y análisis crítico de la ejecución de estos. La "Descripción de los sistemas existentes" es otro punto del capítulo y en el se realiza un análisis crítico y comparativo con respecto a otros software. La "Tendencias, metodologías y tecnologías actuales" también forman parte de este capítulo, en el mismo se presenta una panorámica de las diferentes tendencias actuales, tecnologías Web más usadas y los lenguajes más propicios para la implementación del sistema propuesto.

- Capítulo II _"Construcción de la Solución Propuesta".

  Aquí se hace una descripción textual del modelo de negocio a través de la definición del funcionamiento y ejecución de los procesos de negocio. También se realiza una descripción detallada de las reglas del negocio, planteamiento de los actores y trabajadores del negocio y los diferentes diagramas de casos de usos con sus respectivas, los diagramas de actividades y el diagrama de clases del modelo de objeto. En este capítulo también se encuentra el modelo del sistema del objeto a automatizar; dentro de este la descripción textual de la concepción general del funcionamiento, se definen los requisitos funcionales y no funcionales, se hace una descripción a través de la definición de los actores, los paquetes,

los diagramas de casos de uso y la descripción de los casos de usos del sistema, el diagrama de clases del diseño que no es más que la representación gráfica de los diagramas de clases del modelo de sistema, el diseño de la base de datos y dentro de esta el modelo lógico de datos y el modelo físico de datos, el diagrama de implementación, los principios de diseño y dentro de este, estándares en la interfaz de la aplicación, tratamiento de errores y la Concepción General de la ayuda.

- Capítulo III _"Estudio de Factibilidad"

  En el mismo se presenta la planificación por casos de usos. Se determinan los costos, donde se calculan los valores de las variables de costos utilizadas en la producción de software, como el cálculo del esfuerzo, el tiempo de desarrollo, la cantidad de hombres y el costo total del sistema. Otro punto son los beneficios tangibles e intangibles, y el análisis de costos y beneficios tangibles e intangibles de la implementación del objeto de automatización.

Capítulo 1 – Fundamentación teórica

1.1– Introducción

En este capítulo se expone una breve historia de la organización UJC y como funciona actualmente en la Universidad, se presenta un análisis de los principales conceptos asociados al dominio del problema con el objetivo de comprender lo que se esta realizando, se ofrece una descripción general del objeto de estudio, resultados de análisis de soluciones existentes en el país similares con la propuesta y los objetivos que se persiguen.

Se describen además las tendencias, metodologías y tecnologías actuales de desarrollo utilizadas para el análisis, diseño e implementación del sistema sobre las que se apoya la propuesta, ellos son: Macromedia Dreamweaver MX 2004, Rational Rose para la etapa de análisis y diseño del sistema, el Lenguaje de Modelamiento Unificado (UML), el lenguaje PHP (professional home pages), una descripción del gestor de base de datos relacional MySQL, el servidor apache y un análisis teórico del Proceso Unificado Racional (RUP).

1.2– Breve historia de la UJC.

La Unión de Jóvenes Comunistas (UJC) es la organización juvenil del Partido Comunista de Cuba, forjada en las concepciones marxista leninista, las ideas y la practica del pensamiento de Martí y de Mella, y el ejemplo imperecedero de Fidel, heredera y continuadora de las tradiciones de lucha de la juventud cubana que a lo largo de nuestra historia ha estado en el destacamento de vanguardia revolucionaria.

Cuando triunfa la Revolución Cubana, el 1ro de Enero de 1959, desaparecen los partidos políticos burgueses y con ellos sus organizaciones juveniles. El movimiento revolucionario en el poder agrupa a lo mejor de la juventud cubana progresista y democrática, integrada, mayoritariamente, por el “Movimiento 26 de Julio”, el “Directorio Estudiantil 13 de Marzo” y la “Juventud Socialista”. **[3]**

Al calor de las conquistas alcanzadas y de las tareas de la Revolución, comienzan a surgir nuevas organizaciones juveniles donde se concentran los jóvenes de los

diferentes sectores, que apoyan el programa del Gobierno Revolucionario y ansían participar en él: por ejemplo: “La Confederación de Estudiantes de Segunda Enseñanza” (CESE), La Asociación Campesina “Frank País”, la cual se convirtió luego en la “Asociación Nacional de Agricultores Pequeños” (ANAP). La “Asociación Juvenil de Ayuda a la Revolución” (AJAR), “Las Patrullas Juveniles”, Los “Grumetes Revolucionarios”, “La Asociación de Jóvenes Rebeldes” (AJR).

El 13 de marzo de 1962, en un acto político en la escalinata de la Universidad de La Habana, en homenaje a los caídos en el “Asalto al Palacio Presidencial” y a “Radio Reloj”, Fidel habló de la posibilidad de cambiar el nombre y el carácter de la “Asociación de Jóvenes Rebeldes”; en su discurso explicó las razones de su propuesta y del porqué la organización juvenil debía llevar, en su nuevo nombre, la palabra comunista.

“(...) La juventud constituye la generación futura, una generación que debe vivir en otras etapas más elevadas, más avanzadas. No en el socialismo, sino en el comunismo (APLAUSOS). Y esto quiere decir, sencillamente -lo comprende cualquiera- que a las futuras generaciones hay que prepararlas para la sociedad futura. Que hay que forjar al hombre del futuro desde ahora. Hay que forjar sus sentimientos, su conciencia, su carácter, su espíritu. Hay que desarrollarlo en una conciencia todavía más generosa, en un espíritu todavía más revolucionario, más avanzado, más nuevo.” **[3]**

Del 31 de marzo al 4 de abril de 1962, se realiza el Primer Congreso de la AJR, es en él donde la organización de la juventud revolucionaria cubana adopta los principios y normas marxistas leninistas y, a propuesta de nuestro Comandante en Jefe Fidel Castro, el nombre de Unión de Jóvenes Comunistas (UJC). **[3]**

1.3– La UJC en la “Universidad de Cienfuegos”

La UJC en Cienfuegos esta estructurada en un comité UJC y cuatro comités primarios, los cuales se corresponden con las facultades, en total son 55 Comités de base, 23 comités de base en Ciencias Económicas y Empresariales, 12 en Informática, 11 en Humanidades y ó en Mecánica, además de 3 independientes que no están dentro de los cuatro comités primarios haciendo un total de 789 militantes, estos comités de base están distribuidos convenientemente según especialidades, años, grupos, etc. Los 3 comités de base independientes que no están dentro de ninguna facultad son: Seguridad y protección, Rectorado y Biblioteca.

Los comités de base están compuestos por militantes, de los cuales algunos dirigen la organización de base con cargos como Secretario General, Secretario Ideológico, Secretario Organizador y Activista de Beca, siendo posible que algún integrante sea miembro del comité primario o del comité UJC.

Los comités primarios tienen una estructura similar, teniendo responsabilidades como Secretario General o pueden tener un segundo secretario, un organizador además de otros dirigentes que se agrupan según diferentes esferas. Estas son: Esfera Ideológica, Esfera de Funcionamiento y Esfera de Beca, todas ellas con un jefe y otros integrantes. Además de tener la posibilidad al igual que los comités de base de tener un militante miembro del comité UJC. Dentro de la Esfera de Funcionamiento tenemos al organizador.

En cuanto al Comité UJC, presenta una estructura similar, teniendo los cargos de Secretario General, Segundo Secretario, Organizador, con las Esferas Ideológicas, de Funcionamiento y Beca, todas con un jefe e integrantes. Es bueno aclarar que esta estructura no es estática, sino que se encuentra actualmente en función en el instituto, pudiendo encontrarse algunas diferencias con respecto a otros centros.

En la organización se realiza diversos procesos políticos, entre ellos tenemos el de crecimiento, el de evaluación, el pase al Partido, además del pago de la cotización, la realización de reuniones, el manejo de datos de los militantes, procesos estos que actualmente se realizan de forma manual o con herramientas informáticas rudimentarias (Excel, Access).

Además existe un control de la vida interna, en ella se recoge: los datos de los militantes que se dan de alta o baja del centro, el carácter de las reuniones (ordinarias o extraordinarias), las cuales de las no efectuadas, la participación de los instructores y miembros de dirección de la organización, la asistencia de los militantes, las causas de los ausente, sanciones, los datos de los que reúnen condiciones para el ingreso al Partido, las evaluaciones de los militantes y el pago de la cotización, entre otras cosas.

1.4 – Descripción del dominio del problema

Gestionar: es coordinar todos los recursos disponibles para conseguir determinados objetivos, implica amplias y fuertes interacciones fundamentalmente entre el entorno, las estructuras, el proceso y los productos que se deseen obtener. **[4]**

Comité UJC: Es el organismo de dirección de la UJC en un territorio. Como norma se le subordinan todos los organismos y comités de base constituidos. La función del comité UJC del centro es unificar, orientar y dirigir los diferentes comités primarios y hacer que la UJC funcione de forma cohesionada y eficiente según sus objetivos. **[1]**

Comités Primarios: Es la estructura jerárquicamente inferior inmediata del Comité UJC y se encarga de orientar, evaluar, controlar y unificar el trabajo de los comités de base bajo su dirección. Solo existen en lugares donde hay gran cantidad de militantes.

Comité de base: Es la célula fundamental, estos pueden constituirse donde haya tres o mas miembros, en todo lugar de residencia, escuela, trabajo y donde haya actividad de los jóvenes. Estos desarrollan una actividad abierta a todos los jóvenes, aplica la orientación general de la UJC, promueve en forma unitaria todas aquellas actividades que responden a los intereses y a las aspiraciones políticas, económicas, sociales, deportivas y culturales de los jóvenes. **[1]**

Reunión: Se efectúan en los comités de base, los comités primarios y Comité de la UJC ordinariamente todos los meses y extraordinaria solo cuando se hace necesario. Serán validas con la participación de más del 50% del total de sus miembros. En ella se define lo que el comité hará en ese mes, se discuten y analizan los problemas que tiene la UJC, se expresa libremente las opiniones y sugerencias de los militantes, se ejerce la critica y la autocrítica, se logra una unidad de pensamientos y acción materializada en la adopción y el cumplimiento de los acuerdos.

Acta: Es obligatorio levantarla en cada reunión, en ella se recoge: el nombre del comité de base, fecha de realización, hora de comienzo, de terminación, carácter de la reunión, total de militantes en plantilla, cantidad que asisten, nombre y apellido de los presentes y de los ausentes, de estos últimos las causas por las cuales no están presentes, si existen visitantes o invitados poner el nombre, apellido y cargo, orden del día, tema de instrucción política, las intervenciones, los acuerdos que se toman, si se sanciona un militante poner las causas y la firma del secretario general del comité de base como constancia de que se realizo la reunión. El comité de base tiene 72 horas después de la reunión para entregar esta acta al archivo del instituto. **[1]**

Cotización: Los Estatutos de la organización definen dentro de los deberes de los militantes abonar las cuotas establecidas a los fondos de la misma.

¿Cómo se cotiza?

- No reciben ingresos $0.20
- De $50.00 a $129.99 $0.50
- De $129.99 a $200.00 $2.00
- De $200.00 a $299.99 $4.00
- De $300.00 a $ 399.99 $8.00
- De $400.00 a $499.99 $15.00
- De $500 a más 3% del salario.

1.5 – Descripción del objeto de estudio

1.5.1 - Objetivos estratégicos de la organización

Los objetivos estratégicos que persigue la organización de la UJC son:

1. Consolidar la política de cuadro y la preparación integral de la reserva.
   - Contar con el levantamiento de cuadros y sus reservas.
   - Evaluar con periodicidad la política de cuadro y sus reservas.
   - Implementar un sistema de preparación de los cuadros.
   - Lograr una estabilidad en la dirección de la organización de base.

2. Fortalecer el trabajo de la UJC, logrando una consolidación del funcionamiento orgánico.
   - Consolidar el funcionamiento de las estructuras internas de la organización en el centro.
   - Dar estricto cumplimiento al ciclo de visitas a la organización de base.
   - Aumentar la disciplina informativa y potenciar la creatividad en las reuniones.
   - Elevar cualitativamente el desarrollo de los programas políticos mediante análisis periódicos.
   - Levar a cabo mensualmente reuniones y despachos con secretarios y orientadores políticos.
   - Aplicar una política de sanciones coherentes

3. Fortalecer la responsabilidad política de la UJC en el cumplimiento de las metas de trabajo de la FEU y su Residencia Estudiantil.
   - Creación de los grupos de trabajo de la UJC en la residencia Estudiantil.
   - Participación periódica en los Consejos de la FEU y la Residencia Estudiantil.
   - Lograr que el Consejo de Beca se convierta en el principal eslabón para la labor político-ideológica en la residencia.
   - Propiciar espacios de reflexión y debate en áreas de la residencia aprovechando el uso de las Mesas Redondas, etc.
   - Aumentar el protagonismo de nuestras militantes en la Residencia Estudiantil

4. Fortalecer el trabajo político-ideológico mediante la preparación directa en los programas de la Batalla de Ideas.
   - Lograr un sistema dinámico de preparación político-ideológico que tengo su base en los contenidos afines de las Mesas Redondas, intervenciones de los principales dirigentes de la Revolución y otras formas que se desarrollan en la Batalla de Ideas.
   - Profundizar en los discursos de las viñas de instrucción política, temas de interés y documentos orientados.
   - Profundizar en el conocimiento de nuestra historia, a través de los matutinos especiales y la realización de actos en efemérides significativas.
   - Lograr la participación consciente en la preparación política-ideológica, ética, moral revolucionaria de la militancia como premisa para la formación integral como futuro miembro del PCC.
   - Trabajar estrechamente con las cátedras Honoríficas y desarrollar acciones que contribuyan a profundizar y difundir el ideario revolucionario de José Martí y Carlos Rafael Rodríguez.

5. Consolidar el papel de la organización en la implementación e impulso de los Programas de la Revolución.
   - Lograr que cada estudiante tenga una tarea de impacto social.
   - Propiciar el conocimiento de los programas siendo protagonista de los mismos

6. Consolidar el papel de la UJC en la formación de un integral de los estudiantes a partir del cumplimiento del Proyecto Docente Educativo**.**
   - Lograr el protagonismo de nuestros militantes en los procesos de integralidad.

7. Perfeccionar desde nuestra organización la atención a los estudiantes de otras nacionalidades.
   - Crear espacios para su participación en tareas del centro y propias.
   - Acompañar el desempeño de la Brigada de Solidaridad.

8. Lograr la implementación de un sistema emulativo que permita conocer la labor de los más destacados.
   - Crear una guía emulativa con parámetros definidos.

Misión:

Asume la formación comunista de las nuevas generaciones, sustentada en el patriotismo, la fidelidad al Partido Comunista de Cuba, la defensa de los más altos valores humanos y el espíritu profundamente antiimperialista e internacionalista que ha distinguido a la Revolución Cubana; propiciando la participación consciente y entusiasta de todos los jóvenes en su obra, expresado en el estudio, el trabajo y la defensa de la Patria, simbolizados en su emblema, a través de los paradigmas: Mella, Camilo y el Che.

Como vanguardia de la juventud cubana:

- Defiende la dignidad y la libertad de los cubanos, convencida de que el socialismo es la única perspectiva de desarrollo, justicia social y democracia en una patria independiente y soberana.
- Asume el principio de no claudicar jamás frente a la amenaza del imperialismo y de los que desde nuestro país le hacen el juego.

- Defiende, sin menoscabo, la unidad y el respeto a la pluralidad de puntos de vista en el seno de la Organización y fuera de ella, en aras de que prevalezcan siempre la honestidad, la honradez, la valentía política, la objetividad en los análisis y que estos reflejen los valores, las luchas y condiciones de nuestra sociedad.
- Es internacionalista y profundamente solidaria con las causas más justas de todos los pueblos y con sus compañeros de fila.

1.5.2 - Flujo actual de los procesos y análisis crítico de la ejecución de estos.

Actualmente en el Comité UJC del instituto se manejan todos los datos de los procesos de manera manual, es decir, a pesar de que los datos que puedan recogerse se almacenen en una tabla en Excel o Access se le hace insuficiente a la hora de emitir algún reporte y tienen que recurrir a la revisión de una por una todas las actas de los comités de base de las facultades.

No cabe duda del estricto control que mantiene la organización de todos los datos de sus miembros y de todas sus actividades, pero es una realidad que a veces cuando se hacen necesaria la búsqueda de algún dato en especifico, se pierde mucho tiempo, material humane y de oficina, y puede tornarse la búsqueda un poco difícil y engorrosa.

En ciertas ocasiones ocurren pérdidas, omisiones o errores de información que pueden provocar serias dificultades a la hora de tomar decisiones importantes de la organización, además se hace necesario un local grande para almacenar todos los expedientes de cada uno de los militantes, y de personal que llene dichos expedientes. Personal que es restringido, debido a que es información confidencial que solo les compete a los dirigentes de la organización.

Mensualmente el Comité de la UJC debe entregar un reporte al Comité Provincial de la organización, con todos los datos de las actividades que realiza normalmente. Si a esto le sumamos la actividades que surgen por la actual situación del país, el cúmulo de información se hace aun mayor, para lo cual se necesita más personal y recursos.

Las causas que originan la situación problemática.

- El cúmulo de información que necesita la organización para trabajar y proyectar las acciones.
- La falta de espacio y el deterioro gradual que sufre la información en soporte de papel.
- La no existencia de un software para el manejo de la información o parte de ella que se manipula en la organización.

Las consecuencias

- La pérdida de documentos y compromisos importantes para el buen funcionamiento de la organización.
- No cumplimiento con las actividades que asignan de la provincia.
- Una mala comunicación entre los comités primarios y los comités de base con la dirección del centro.
- Dificultad a la hora de tomar una decisión.

1.6 – Descripción de los sistemas existentes

En el mundo existen muchas organizaciones que al igual que la UJC responden a los intereses de un grupo de personas, en ninguno de los sitios consultados sobre diferentes organizaciones se encontró ningún sistema que estuviera vinculado a la recopilación de los datos de sus miembros; ni a las principales tareas, todo lo que se encuentra es relacionado con la historia de la organización, la misión, los objetivos, las metas trazadas: etc. Independientemente que no se encontró ninguna aplicación, si se encontraron trabajos de diplomas que hacen referencia a sistemas que de una forma u otra están relacionados con nuestra aplicación Web, ellos son:

1. Sistema Automatizado para el Control de la Militancia UJC-CUJAE, de la autora Yaneysis E. Alcaz Mecías, en el año 2000, del Instituto Superior Politécnico "José Antonio Echeverría" **[5]**. El trabajo tiene como principal objetivo automatizar el control de la militancia, permitiendo de una forma fácil y rápida el manejo de dicha información. Se utiliza el Delphi 5.0 para el tratamiento de la interfaz con el usuario y SOL Server 7.0 para la administración de la base de

datos. El sistema realiza la captación de los datos de los militantes, el funcionamiento de sus organizaciones de base y la generación de algunos reportes muy útiles para el trabajo de la organización. En el momento que fue presentado mejoraba sustancialmente la rapidez y la seguridad del manejo de la información, perfeccionando su control y brindando una valiosa ayuda en la toma de decisiones, además de eliminar el engorroso trabajo manual que esto implica. Tal sistema a pesar de las funcionalidades que presenta, tiene la limitación de ser una aplicación desktop, lo cual imposibilita la potenciación de sus requisitos funcionales.

2. Sistema para la Gestión de Información Integral de la FEU (FEUGest), de los autores Yurima Pérez Calzadilla y Loan Joa Matos, en el ana 2004, del Instituto Superior Politécnico "José Antonio Echeverría" **[6]**. Implementaron un sistema basado en tecnologías Web que permite mejorar y agilizar el proceso de gestión de la información sobre los estudiantes de la CUJAE. El trabajo garantiza de forma ágil y factible el registro, actualización, organización y protección de las informaciones y estadísticas relacionadas con la participación de los estudiantes en los diferentes ámbitos de la vida universitaria. Además, permite tener un control preciso, individual y colectivo, de la magnitud y resultados de la participación de cada estudiante en las actividades que organiza (o es convocada) la universidad. Contribuye a profundizar en los procesos de análisis Integral de los estudiantes, así como en la evaluación y caracterización de la trayectoria universitaria de cada uno, a partir de elementos de juicio precisos y concretos. El trabajo influye decisivamente en el funcionamiento de la Federación Estudiantil Universitaria (FEU), debido a que se manejan en cualquier momento, elementos, datos e informaciones vitales para que la toma de decisiones sea un éxito rotundo. Tal sistema por sus características no se adecua a las funcionalidades y requerimientos específicos que se persigue con la solución propuesta.

1.7 – Aplicaciones Web.

En este siglo de grandes avances tecnológicos, en donde el uso de las computadoras se ha generalizado, a aparecido un nuevo termino: Aplicación Web.

Aplicaciones que sirven a miles de usuarios conectados al unísono y distribuidos por todas partes del mundo, todos conectados a la vez si se requiere.

Una aplicación Web es una aplicación informática distribuida cuya interfaz de usuario es accesible desde un cliente Web, normalmente un navegador Web.

Este concepto tiene diferentes significados para cada desarrollador, pero en este trabajo consideraremos una aplicación Web como un sistema Web (servidor Web, red, HTTP, navegador) donde la entrada del usuario (entrada de datos y navegación) afecta el estado del negocio. Su arquitectura general es la de un sistema cliente/servidor con notables distinciones. Una de sus ventajas más significativas es su fácil instalación y distribución.

Entre los lenguajes del lado del servidor podemos encontrar entre los más sobresalientes por el auge que estos han tenido, algunos como Practical Extracting and Reporting Language (PERL) **[7]**, Active Server Pages (ASP) **[8]**, Professional Home Pages (PHP) **[9]**, Páginas de Servidor Java (JSP) **[10]**, entre otros. Estos se caracterizan por desarrollar la lógica de negocio dentro del servidor, además de ser los encargados del acceso a bases de datos, tratamiento de la información; etc. Del lado del cliente se encuentran principalmente el JavaScript **[11]** y el Visual Basic Script **[12]**, que son los encargados de aportar dinamismo a la aplicación en los navegadores. **[13]**

Instalar una aplicación Web consiste en configurar los componentes del lado del servidor en la red y no es necesaria una instalación o configuración en el lado cliente.

El éxito espectacular de la Web se basa en dos puntales fundamentales: el protocolo HTTP y el lenguaje HTML. Uno permite una implementación simple y sencilla de un sistema de comunicaciones que nos permite enviar cualquier tipo de ficheros de una forma fácil, simplificando el funcionamiento del servidor y permitiendo que servidores poco potentes atiendan miles de peticiones y reduzcan los costes de despliegue.

El otro nos proporciona un mecanismo de composición de páginas enlazadas simple y fácil, altamente eficiente y de uso muy simple.

A diferencia de las aplicaciones tradicionales la comunicación del cliente y el

servidor se realiza a través del protocolo principal de comunicación en una aplicación Web que es el HTTP, el cual funciona normalmente desconectado, es decir, el cliente hace una petición al servidor, este la procesa y le devuelve el resultado, terminando la comunicación entre estos.

La arquitectura básica de una aplicación Web incluye los navegadores, la red y el servidor Web. Los navegadores solicitan las páginas Web al servidor. Cada página Web esta expresada en instrucciones de formato HTML. Algunas páginas incluyen scripts del lado cliente que son interpretadas por el navegador. Estos scripts definen el comportamiento dinámico adicional para la página y pueden interactuar con el navegador, el contenido de la página y los controles adicionales contenidos en ella (Applets de Java, controles Active X y plug-ins).

El usuario ve e interactúa con el contenido de la página. Algunas veces el usuario introduce información en campos de la página y los envía al servidor para procesarlos. El usuario también puede interactuar con el sistema navegando a diferentes páginas a través de los hipervínculos.

Desde la perspectiva del cliente, la pagina Web es siempre un documento HTML. En el servidor, sin embargo, una "pagina Web" puede manifestarse de distintas maneras. En las primeras aplicaciones Web, las páginas dinámicas eran construidas usando CGI (Common Gateway Interface). La CGI procesaba las reglas de negocio y devolvía el resultado siempre en formato HTML.

La arquitectura de una aplicación Web no es muy diferente a la de un sitio Web dinámico, la diferencia radica en su utilización. Ya que en las aplicaciones Web se implementa la lógica de negocios y su uso cambia el estado del negocio. Aspecto muy importante porque define el enfoque del esfuerzo a la hora de modelarlas. La lógica de negocios esta dado por dos conceptos muy importantes: las reglas de negocio y los requerimientos del negocio.

Toda aplicación trata de reflejar parte del funcionamiento del mundo real para automatizar tareas que de otro modo serian llevadas a cabo de modo más ineficiente, o bien no podrían realizarse. Para ello es necesario que cada

aplicación refleje las restricciones que existen en el negocio dado, de modo que nunca sea posible llevar a cabo acciones no válidas. A las reglas que debe seguir la aplicación para garantizar esto se les llama reglas de negocio. Ejemplos de tales reglas son: no permitir crear facturas pertenecientes a clientes inexistentes, controlar que el saldo negativo de un cliente nunca sobrepase cierta cantidad, etc.
Los requerimientos del negocio, por su parte, son restricciones, usualmente impuestas por la propia compañía o entidad, que estas utilizan para ayudarse a operar en su ambiente particular de negocio. Los requerimientos del negocio definen una línea para los requerimientos de la aplicación y proporcionan una guía para los desarrolladores. En términos prácticos, estos requerimientos del negocio son metas que los desarrolladores se esfuerzan por conocer para sus aplicaciones.
Juntos, las reglas y los requerimientos de negocio proporcionan un entorno para desarrollar aplicaciones que pueden ayudar a la compañía a alcanzar sus metas.
Como las aplicaciones Web ejecutan lógica de negocios lo más importante en la modelación es su lógica y el estado del negocio, no los detalles de presentación.

1.8 – Tendencias, metodologías y/o tecnologías actuales.

1.8.1 – Macromedia Dreamweaver MX 2004.

¿Qué es Macromedia Dreamweaver MX 2004?

Macromedia Dreamweaver MX 2004 es uno de los más utilizados en el desarrollo Web a nivel profesional. Su amplio abanico de herramientas permite crear desde la más simple página web personal hasta el sitio web más completo y complejo para una gran empresa.

Desde su aparición en Diciembre de 1997, Macromedia Dreamweaver ha llegado a ser la solución estándar de la industria, para los profesionales del desarrollo Web. Dreamweaver actualmente abarca aproximadamente el 80% del mercado de las herramientas profesionales para este tipo de desarrollo, según el informe de

ventas realizado por la NPD Intelect's de Febrero 2002, y más de 2.4 millones de profesionales de la Web dependen de los productos Dreamweaver.

Macromedia Dreamweaver MX, un producto revolucionario que permite a los desarrolladores diseñar y crear código para una completa gama de soluciones, desde sitios Web hasta aplicaciones para Internet, sin comprometer el enfoque principal del producto para los usuarios sólo de HTML. Dreamweaver MX combina en un único entorno de desarrollo accesible y potente las reconocidas herramientas de presentación visual de Dreamweaver, las características de rápido desarrollo de aplicaciones Web de Dreamweaver UltraDev y Coldfusión Studio, y el extenso soporte de edición de código de HomeSite. Ofrece una completa solución abierta para las tecnologías Web y estándares de hoy, incluyendo la accesibilidad y servicios Web. **[14].**

Entre las ventajas de este programa, destaca que extiende las capacidades de los navegadores de Web y los dispositivos con conexión a Internet. Esto, porque Macromedia MX está diseñado para aprovechar las capacidades del reproductor Macromedia Flash Player, tecnología que posee actualmente la más amplia cobertura en Internet, ya que está presente en el 98% de los usuarios de la Web. Industrias como Apple, IBM, Intel, Liberate, Microsoft, OpenTV, Sony Ericsson y Sun Microsystems ya dieron su respaldo a esta nueva familia de productos.

Dreamweaver MX combina facilidad y potencia en un entorno de desarrollo integrado para los sitios Web ColdFusion, HTML, XHTML, ASP, ASP.NET, JSP, o PHP. El producto permite un control completo sobre el código y el diseño con la precisión de las herramientas de presentación y las potentes características de codificación como sugerencias de código, editor de etiquetas, codificación del color ampliable, selector de etiquetas, fragmentos y validación de código.

El nuevo espacio de trabajo integrado, compartido con Macromedia Flash MX y Fireworks MX incluye ventanas de documentos con fichas, grupos de paneles acoplables, barras de herramientas personalizables y exploración integrada de archivos. También, por primera vez, Dreamweaver incluye con calidad profesional,

presentaciones preconstruidas y código, incluyendo las estructuras del sitio, informes, plantillas de accesibilidad, y funciones de JavaScript para la interactividad del lado del cliente.

Se descubre en este producto, los beneficios de los estándares emergentes y las nuevas tecnologías Web con el soporte para XML, servicios Web y el amplio cumplimiento de accesibilidad para rehacer sitios ya existentes y crear aplicaciones de nueva generación.

De entre todos los editores de páginas HTML actuales, sin duda Dreamweaver ocupa una posición líder con respecto a sus competidores. Las razones para ello son varias:

- Dreamweaver genera un código HTML bastante limpio y compatible con la mayoría de los navegadores.
- Incluye varias herramientas añadidas que permiten aumentar la productividad.
- Se integra perfectamente con otras aplicaciones de macromedia, como Fireworks o Flash, creándose todo un entorno de trabajo autosuficiente.
- Ofrece varias herramientas para optimizar y gestionar el sitio completo, incluso un cliente FTP con opciones para mantener sincronizados el sitio local y remoto.
- Añade algunas funciones predefinidas de Javascript, llamadas Behaviors, que facilitan mucho la inclusión de controles y efectos habituales de los lenguajes de scripting para los no programadores.
- La coordinación entre el modo de edición visual y el modo de edición manual esta muy lograda. Esto permite a los programadores y diseñadores experimentados añadir elementos al HTML no soportados por Dreamweaver.

Por estas y otras razones Dreamweaver es sin duda alguna el editor visual HTML más cotizado actualmente.

1.8.2 – UML

UML surge como respuesta al primer problema reseñado para contar con un lenguaje estándar para escribir planos de software. Muchos han creído ver UML como solución para todos sus problemas sin saber en muchos casos de lo que se trataba en realidad.

El Lenguaje Unificado de Modelado, UML es una notación estándar para el modelado de sistemas software, resultado de una propuesta de estandarización promovida por el consorcio OMG (Object Management Group), del cual forman parte las empresas más importantes que se dedican al desarrollo de software, en 1996.

UML es una notación, es decir, de una serie de reglas y recomendaciones para representar modelos. Permite documentar y especificar los elementos creados mediante un lenguaje común describiendo modelos.

¿Qué es UML?

El Lenguaje de Modelamiento Unificado (UML - Unified Modeling Language) es un lenguaje que permite modelar, construir y documentar los elementos que forman un producto de software que responde a un enfoque orientado a objetos. Este lenguaje fue creado por un grupo de estudiosos de la Ingeniería de Software formado por: Ivar Jacob son, Grady Booch y James Rumbaugh en el año 1995. Desde entonces, se ha convertido en el estándar internacional para definir organizar y visualizar los elementos que configuran la arquitectura de una aplicación orientada a objetos **[15].** Con este lenguaje, se pretende unificar las experiencias acumuladas sobre técnicas de modelado e incorporar las mejores prácticas actuales en un acercamiento estándar.

UML no es un lenguaje de programación sino un lenguaje de propósito general para el modelado orientado a objetos y también puede considerarse como un lenguaje de modelamiento visual que permite una abstracción del sistema y sus componentes. **[16]**

Entre sus objetivos fundamentales se encuentran: **[15]**.

- Ser tan simple como sea posible, pero manteniendo la capacidad de modelar toda la gama de sistemas que se necesita construir.
- Necesita ser lo suficientemente expresivo para manejar todos los conceptos que se originan en un sistema moderno, tales como la concurrencia y distribución, así como también los mecanismos de la ingeniería de software, como son el encapsulamiento y los componentes.
- Debe ser un lenguaje universal, como cualquier lenguaje de propósito general.
- Imponer un estándar mundial.

¿Por qué UML?

La decisión de utilizar UML (Unified Modeling Language – Lenguaje Unificado de Modelado) como notación para el desarrollo del software se debe a que se ha convertido en un estándar que tiene las siguientes características:

- Permite modelar sistemas utilizando técnicas orientadas a objetos (OO).
- Permite especificar todas las decisiones de análisis y diseño, construyéndose así modelos precisos, no ambiguos y completos.
- Puede conectarse con lenguajes de programación (Ingeniería directa e inversa).
- Permite documentar todos los artefactos de un proceso de desarrollo (requisitos, arquitectura, pruebas, versiones, etc.).
- Es un lenguaje muy expresivo que cubre todas las vistas necesarias para desarrollar y luego desplegar los sistemas.
- Existe un equilibrio entre expresividad y simplicidad, pues no es difícil de aprender ni de utilizar.
- UML es independiente del proceso, aunque para utilizarlo óptimamente se debería usar en un proceso que fuese dirigido por los casos de uso, centrado en la arquitectura, iterativo e incremental.

1.8.3 – Proceso Unificado de Desarrollo (RUP)

El Proceso Unificado Racional o RUP (Rational Unified Process), es un proceso de desarrollo de software que utiliza el Lenguaje Unificado de Modelado UML, constituye la metodología estándar utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

- Forma disciplinada de asignar tareas y responsabilidades (quién hace qué, cuándo y cómo).
- Pretende implementar las mejores prácticas en ingeniería de Software.
- Desarrollo iterativo.
- Administración de requisitos.
- Uso de arquitectura basada en componentes.
- Control de cambios.
- Modelado visual del software.
- Verificación de la calidad del software.

RUP es un producto de Rational (IBM). Se caracteriza por ser iterativo e incremental, estar centrado en la arquitectura y guiado por los casos de uso. Incluye artefactos (que son los productos tangibles del proceso como por ejemplo, el modelo de casos de uso, el código fuente, etc.) y roles (papel que desempeña una persona en un determinado momento, una persona puede realizar distintos roles a lo largo del proceso). **[17]**

RUP divide el proceso de desarrollo en ciclos, teniendo un producto final al concluir cada ciclo, en cada uno se analizan las fases siguientes:

- inicio: se hace un plan de fases, se identifican los principales casos de uso y se identifican los riesgos.
- elaboración: se hace un plan de proyecto, se completan los casos de uso y se eliminan los riesgos.
- construcción: se concentra en la elaboración de un producto totalmente operativo y eficiente y el manual de usuario

- transición: se implementa el producto en el cliente y se entrena a los usuarios. Como consecuencia de esto suelen surgir nuevos requerimientos a ser analizados.

Para la elaboración de los diagramas y otros modelos que propone RUP, se hará uso del Rational Rose, herramienta CASE desarrollada por los creadores de UML (Booch, Rumbaugh y Jacob-son), que cubre todo el ciclo de vida de un proyecto: concepción y formalización del modelo, construcción de los componentes, transición a los usuarios y certificación de las distintas fases y entregables.

1.8.4 – Rational Rose

Existen herramientas Case de trabajo visuales como el Analise, el Designe, el Rational Rose, que permiten realizar el modelado del desarrollo de los proyectos, en la actualidad la mejor y más utilizada en el mercado mundial es Rational Rose y es la que se utiliza en la modelación de este proyecto.

Rational Rose es la herramienta de modelación visual que provee el modelado basado en UML. **[18]**

La Corporación Rational ofrece un Proceso Unificado (RUP) para el desarrollo de los proyectos de software, desde la etapa de Ingeniería de Requerimientos hasta la de pruebas. Para cada una de estas etapas existe una herramienta de ayuda en la administración de los proyectos, Rose es la herramienta del Rational para la etapa de análisis y diseño de sistemas. **[18]**

Rose es una herramienta con plataforma independiente que ayuda a la comunicación entre los miembros de equipo, a monitorear el tiempo de desarrollo y a entender el entorno de los sistemas.

Una de las grandes ventajas de Rose es que utiliza la notación estándar en la arquitectura de software(UML), la cual permite a los arquitectos de software y desarrolladores visualizar el sistema completo utilizando un lenguaje común,

además los diseñadores pueden modelar sus componentes e interfaces en forma individual y luego unirlos con otros componentes del proyecto.

La metodología usada para desarrollar el proyecto fue RUP. Este es un proceso que garantiza la elaboración de todas las fases de un producto de software orientado a objeto. RUP utiliza UML, que es un lenguaje que permite la modelación de sistemas con tecnología orientada a objetos.

1.8.5 – PHP (Professional Home Pages).

Es un lenguaje de programación embebido en el HTML, en donde se permite realizar determinadas acciones de una forma fácil y eficaz sin tener que generar programas desarrollados íntegramente en un lenguaje distinto al HTML. Es un desarrollo "Open Source", es decir, es gratuito e independiente de plataforma, rápido, con una gran librería de funciones para la explotación de bases de datos de una manera llena y sin complicaciones y posee mucha documentación. **[13]**
Desde sus inicios en 1994 su creador, Rasmus Lerdorf ha recibido muchas contribuciones de otros programadores para adaptarlo a las necesidades actuales. Es un lenguaje de programación del lado servidor, es decir; que es reconocido, ejecutado e interpretado por un servidor, lo que permite que al cliente se le enviara la información en un formato comprensible para él. Las páginas que se ejecutan en el servidor pueden realizar accesos a bases de datos, conexiones en red, y otras tareas para crear la página final que vera el cliente.
Encontrar un servidor que soporte PHP, para publicar las páginas en Internet es relativamente fácil y barato. PHP esta ampliamente difundido entre los servidores Linux o Unix, que no permiten la programación con ASP, lo que hace de PHP el complemento ideal para que sus servidores puedan realizar la programación de páginas con scripts del servidor.

Ventajas:

- Corre en (casi) cualquier plataforma utilizando el mismo código fuente, pudiendo ser compilado y ejecutado en algo así como 25 plataformas,

incluyendo diferentes versiones de Unix, Windows (95, 98, NT, ME, 2000, XP, etc.). Como en todos los sistemas se utiliza el mismo código base, los scripts pueden ser ejecutados de manera independiente al sistema operativo.

- Es completamente expandible. Está compuesto de un sistema principal, un conjunto de módulos y una variedad de extensiones de código.
- Muchas interfaces distintas para cada tipo de servidor. PHP actualmente se puede ejecutar bajo Apache, IiS, AOLServer, Roxen y THTTPD. Otra alternativa es configurarlo como module CGI.
- Puede interactuar con muchos motores de bases de datos tales como MySOL, MS, SQL, Oracle, Informix, PostgreSOL, y otros muchos.
- Posee una gran variedad de módulos cuando se necesita una interfase para una librería en particular, se puede crear fácilmente una API para esta. Algunas de las que ya vienen implementadas permiten manejo de gráficos, archivos PDF, Flash, Cybercash, calendarios, XML, IMAP, POP, etc. **[19]**
- Generalmente es utilizado como modulo de Apache, lo que lo hace extremadamente veloz. Esta completamente escrito en C, así que se ejecuta rápidamente utilizando poca memoria.
- Soporta en cierta medida orientación a objeto. clases y herencia.
- El análisis léxico para recoger las variables que se pasan en la dirección, lo hace PHP de forma automática. Librándose el usuario de tener que separar las variables y sus valores.
- La ejecución de una Web dinámica no precisa de una maquina muy potente. Un 486 con 16 Mb de RAM y con 500 Mb de disco puede ser suficiente. Por supuesto disponer de una maquina más potente permitirá una ejecución más eficiente y una programación más cómoda.
- Se pueden hacer grandes cosas con pocas líneas de código.

- Viene acompañado por una excelente biblioteca de funciones que permite realizar cualquier labor (acceso a bases de datos, encriptación, envió de correo, XML, creación de PDF).

- Se pueden transferir archivos por FTP a partir de sentencias en nuestro código.
- Posee funciones pensadas para Internet (tratamiento de cookies, accesos restringidos, comercio electrónico) o para propósito general (funciones matemáticas, explotación de cadenas, de fechas, corrección ortográfica, compresión de archivos). A esta inmensa librería se le pueden añadir todas las funciones personales que uno va creando por necesidades propias. **[20]**

Sus desventajas son:

- Todo el trabajo lo realiza el servidor y no delega al cliente. Por tanto puede ser más ineficiente a medida que las solicitudes aumenten de número.
- La legibilidad del código puede verse afectada al mezclar sentencias HTML y PHP.
- La orientación a objetos es aun muy deficiente para aplicaciones grandes.
- El manejo de errores no es tan sofisticado como Cold Fusion o ASP.**[20]**

En el ultimo año, el número de servidores que utilizan PHP se ha disparado, logrando situarse cerca de los 5 millones de sitios y 800 000 direcciones IP, lo que le ha convertido a PHP en una tecnología popular.

Otros lenguajes como Perl. ASP y JSP tienen características similares al PHP aunque poseen rasgos que los marcan y por ello los distingue, ellos son:

- **Multiplataformas**: menos el ASP, que es solamente soportado por la plataforma Windows, los demás lenguajes están soportados en múltiples plataformas.
- **Velocidad de ejecución**: la velocidad es mayor en PHP, seguidos por PERL y JSP.
- **Disponibilidad de recursos**: actualmente los mas utilizados en la Internet son el PHP y el JSP, siendo mas utilizado en la publicación de artículos y códigos de ejemplos. PHP tiene una de las comunidades más grandes en Internet, al igual que la de Java.
- **Familiaridad con el lenguaje**: En la universidad los lenguajes mas utilizados por los programadores es el ASP y el PHP.

Servidor Web Apache.

Un servidor de páginas Web es un programa que permite acceder a páginas Web alojadas en un ordenador. Hoy en día Apache es el servidor Web mas utilizado del mundo, encontrándose muy por encima de sus competidores, tanto gratuitos como comerciales. Es un software de código abierto que funciona sobre cualquier plataforma. Desde su origen ha evolucionado hasta convertirse en uno de los mejores servidores en términos de eficiencia, funcionalidad y velocidad, surgió en abril de 1996 ya en julio del 2002 era utilizado por el 57 % de los sitios Web de Internet. **[21]**

Tiene capacidad para servir páginas tanto de contenido estático, para lo que nos serviría sencillamente un viejo ordenador 486 como de contenido dinámico a través de otras herramientas soportadas que facilitan la actualización de los contenidos mediante bases de datos, ficheros u otras fuentes de información, es muy potente y altamente configurable.

Los servidores Web suministran páginas Web a los navegadores (como por ejemplo. Netscape Navigator, Internet Explorer de Microsoft) que lo solicitan.

En términos más técnicos, los servidores Web soportan el Protocolo de Transferencia de Hipertexto conocido como HTTP (HyperText Transfer Protocol), el estándar de Internet para comunicaciones Web. Usando http, un servidor Web envía páginas Web en HTML y CGI, así como otros tipos de scripts a los navegadores o browsers cuando estos lo requieren. Cuando un usuario hace click sobre un enlace (link) a una pagina Web, se envía una solicitud al servidor Web para localizar los datos nombrados por ese enlace. El servidor Web recibe esta solicitud y suministra los datos que le han sido solicitados (una página HTML, un script interactivo, una página Web generada dinámicamente desde una base de datos, etc.). O bien devuelve un mensaje de error.

El servidor Apache es un software que esta estructurado en módulos, es decir, esta dividido en muchas porciones de código que hacen referencia a diferentes aspectos o funcionalidades del servidor Web.

Esta modularidad es intencionada ya que la configuración de cada modulo se hace mediante la configuración de las directivas que están contenidas dentro del

módulo.

Los módulos del Apache se pueden clasificar en tres categorías:

- Módulos Base: Modulo con las funciones básicas del Apache.
- Módulos Multiproceso: Son los responsables de la unión con los puertos de la maquina, aceptando las peticiones y enviando a los hijos a atender alas peticiones.
- Módulos Adicionales: Cualquier otro módulo que le añada una funcionalidad al servidor.

Las funcionalidades mas elementales se encuentran en el modulo base, siendo necesario un modulo multiproceso para manejar las peticiones. Se han diseñado varios módulos multiproceso para cada uno de los sistemas operativos sobre los que se ejecuta el Apache, optimizando el rendimiento y rapidez del código.

El resto de funcionalidades del servidor se consiguen por media de módulos adicionales que se pueden cargar. Para añadir un conjunto de utilidades al servidor, simplemente hay que añadirle un modulo, de forma que no es necesario volver a instalar el software.

Gestor de base de datos MySQL.

Una base de datos es un conjunto de datos estructurados, almacenados sin redundancia en algún soporte de almacenamiento de datos, y se puede acceder a ella desde uno o varios programas. **[22]**

Actualmente el modelo más utilizado para modelar problemas reales y administrar datos dinámicamente es el de bases de datos relacionales, ya que permite almacenar los datos en tablas separadas en vez de poner todos los datos en un solo lugar, esto agrega velocidad y flexibilidad. Las tablas son enlazadas al definir relaciones que hacen posible combinar datos de varias tablas cuando se necesitan consultar datos. El lenguaje más común y estandarizado para acceder a bases de datos relacionales es el SOL. (Structured Ouery Language)

El Sistema Gestor de Bases de Datos (SGBD) es un conjunto de programas,

procedimientos y lenguajes que proporcionan a los usuarios las herramientas necesarias para operar con una base de datos, es decir, actúa como un intermediario entre los usuarios y los datos. **[22]**

Sus principales funciones están relacionadas con la creación y mantenimiento de la base de datos, el control de accesos, la manipulación de datos de acuerdo con las necesidades del usuario, el cumplimiento de las normas de tratamiento de datos, evitar redundancias e inconsistencias y mantener la integridad.

MySOL usa GPL (Licencia Pública General) lo que significa que es accesible para cualquiera, para usarlo o modificarlo. Es la base de datos de código fuente abierto más usada del mundo, que implementa el modelo relacional. Su ingeniosa arquitectura lo hace extremadamente rápido y fácil de personalizar. La extensiva reutilización del código dentro del software y una aproximación minimalística para producir características funcionalmente ricas, ha dado lugar a un sistema de administración de la base de datos incomparable en velocidad, compactación, estabilidad y facilidad de despliegue. La exclusiva separación del core server del manejador de tablas, permite funcionar a MySOL bajo control estricto de transacciones o con acceso a disco no transaccional ultrarrápido. Además ofrece una rica variedad de funciones. Su conectividad, velocidad y seguridad hacen a MySQL altamente satisfactorio para acceder bases de datos. **[23]**

Es un gestor de base de datos relacionales cliente-servidor que utiliza el SOL para el acceso a bases de datos. Consume muy pocos recursos, tanto de Unidad Central de Procesamiento (CPU) como de memoria. Resulta útil para la administración (backup, recuperación de errores), no tiene límites en el tamaño de los registros, es portable de modo que se puede emplear tanto en servidores Linux como Windows y realiza control de acceso de usuarios y permisos. En el caso de bloquearse el servidor MySOL, no suele perder información ni corromper los datos. Además esta disponible para casi todas las plataformas hardware y sistemas operativos.

Este sistema de bases de datos nació con Internet y es parte fundamental de sitios Web en todo el mundo. Su integración con PHP permite la creación de

aplicaciones Web de forma inmediata y natural. **[24]**

Entre los SGBD comúnmente utilizados en el mundo están Oracle **[23]**, MySOL, Microsoft SOL Server, entre otros. A pesar de que todos presentan un enfoque relacional con un buen basamento matemático centrado en el Algebra Relacional y facilitan el trabajo con la base de datos, tienen características que los diferencian, por ejemplo:

- Oracle: requiere de una licencia para poderlo utilizar, es decir, es necesario pagar para poder utilizarlo **[25]**
- Microsoft SOL Server: no es multiplataforma, solo puede ser utilizado con el sistema operativo Windows que esta patrocinado por la compañía Microsoft. **[26]**
- MySOL: es gratuito y multiplataforma.

1.9 – Conclusiones

En el presente capitulo se expusieron los problemas actuales del objeto de estudio y las características de las aplicaciones relacionadas con el proceso que se desea automatizar, se evidencia la necesidad de desarrollar un sistema informático con el objetivo de controlar toda la información referente a la Unión de Jóvenes Comunistas de Cienfuegos.

Verdaderamente con el auge que ha tomado Internet y la consecuente digitalización de la información se ha acelerado gradualmente el conocimiento científico y la evolución tecnológica. Por ello es muy importante una excelente selección de la tecnología que necesita la aplicación que se debe realizar, para que se logren los mejores resultados y se explote el potencial tecnológico requerido. Por lo antes expresado se cierra este capitulo exponiendo las herramientas utilizadas.

Para desarrollar el sistema se hace uso de la tecnología para la programación de páginas dinámicas el lenguaje PHP, con soporte de base de datos en MySQL y

como servidor Web el Apache, todo esto utilizando como herramienta de desarrollo el editor de páginas Web Dreamweaver MX 2004.

El proceso de desarrollo a utilizar es RUP, el cual esta basado en la orientación a objetos, el desarrollo interactivo y la modelación visual usando UML, lo cual permite incorporar al proceso de desarrollo de software un mejor control de los requerimientos y cambios. Posibilitando la distribución del trabajo en diversos frentes de forma simultanea.

A pesar de ser un proceso desarrollado directamente para el trabajo con clases y objetos, brinda amplias posibilidades para el desarrollo eficiente del diseño e implementación de aplicaciones Web.

Capítulo 2 – Construcción de la solución propuesta.

2.1 – Introducción

Este capítulo presenta la descripción actual del modelo del negocio para facilitar una visión más clara del proceso en cuestión, las reglas de negocio que debe seguir la aplicación para garantizar las restricciones que existen, la definición de los conceptos y el planteamiento de los actores y trabajadores, sus casos de uso y su formato expandido. Se plantean los detalles relacionados con el diseño del sistema que se propone, se analizan los requerimientos funcionales y no funcionales, los casos de uso con sus descripciones y sus actores con sus respectivos diagramas, la representación gráfica de los diagramas de clases del modelo de sistema, el diseño de la base de datos del modelo físico y lógico con sus diagramas, los diagramas de implementación y los principios de diseño con los estándares en la interfaz de la aplicación, los tratamientos de errores y la concepción general de la ayuda.

2.2 – Descripción del modelo de negocio

El proceso de gestión de la información de la UJC en la UCF se hace de forma manual, por lo que no existe un buen intercambio de la misma. Esta información solo se desenvuelve en la oficina de la organización y esto trae consigo una lenta comunicación entre los miembros, además de un atraso en el trabajo.

La información es compartida tanto para los directivos de la Universidad como para los de los comités de base. A los miembros se les hace difícil interactuar con dicha información ya que es un poco tediosa su búsqueda. En caso de que el militante no se encuentre en la Universidad o este fuera de la provincia se le hace difícil la solicitud de la misma.

2.3 – Reglas del negocio a considerar

El negocio comienza cuando el Secretario General de la UJC del instituto, solicita los portes de cotización y funcionamiento, a los Secretarios de Cotización y Funcionamiento respectivamente, estos le permiten valorar la situación actual de los comités de base de la universidad y en caso de algún incumplimiento poder realizar con tiempo un análisis y darle solución al problema surgido.

El Secretario General del Comité, solo le da solución a los problemas mas críticos, es

decir, aquellos que no han podido ser solucionados por las facultades, ya que los secretarios de los comités primarios deben tratar de solucionar sus problemas, siempre que esto sea posible, de la misma manera ocurre en los comités de base. En cada nivel existe un secretario de cotización y de funcionamiento, que son los encargados de dichas funciones y de elevar al nivel superior los resultados, para que los respectivos jefes puedan tomar decisiones encaminadas a resolver los problemas existentes.

Para que el proceso de cotización se realice con mejor efectividad es necesario que se cumplan ciertas reglas:

- La cotización será recogida solo por el Secretario de Cotización, el cual debe anotarlo en el libro de control como constancia de tal efecto.
- La cotización se recogerá mensualmente.
- Es un deber de cada militante abonar la cuota establecida.

De la misma manera ocurre con el proceso de funcionamiento. Las reglas son:

- Los comités de base, los comités primarios y el Comité de la UJC del instituto debe reunirse mensualmente en reunión ordinaria y extraordinariamente siempre que sea necesario.
- En cada reunión se debe levantar un acta, si esto no sucediera, no se considera válida le reunión.
- En el acta deben estar reflejados todos los aspectos que se tratan en la reunión.
- Las actas son recepcionadas solo por el secretario de funcionamiento del instituto, el mismo esta en la obligación de revisarlas y solo darle entrada si recogen todos los datos establecidos por la organización.
- Luego de realizada la reunión el Secretario del Comité de Base tiene 72 horas para entregar el acta, fuera de ese periodo de tiempo, se consideran fuera de fecha.
- Una reunión se considera válida una vez entrada el acta a la oficina de la UJC.
- Siempre que un Comité de Base se reúna, debe fijar la fecha de la próxima reunión y está en la obligación de informarla al Secretario de Funcionamiento con antelación, para que pueda ser visitada.

- Cada militante que arribe al centro debe presentarse en el comité del Instituto para

que sean registrados sus datos y. pueda participar en las actividades que ella convoca.

- Para darle baja definitiva a un militante, es decir, eliminarlo por completo de la base de datos, el comité de base que lo acogerá debe de haber confirmado su aprobación por escrito o telefónicamente. No obstante, en los datos del comité de base debe quedar registrado el lugar de destino, por si pudiera existir alguna cuestión que discutir con el militante.
- Cada reunión que realicen los comités de bases, pueden ser visitadas por miembros del Partido Comunista de Cuba (PCC), profesores guías directivos de las facultades y del instituto, así como dirigentes de la UJC.
- La justificación de los ausentes a las reuniones, no puede ser ninguna otra a las establecidas por la organización, es decir. fuera de esas se considera ausente injustificado.
- Una reunión se considera válida si asiste el 50% +1 de los militantes.

2.4 – Modelo de casos de uso del negocio

2.4.1 – Actores del negocio.

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

En el proceso participan diferentes actores y trabajadores del negocio para lograr la gestión de toda la información. A continuación se muestra en la tabla 2.1 de la solución propuesta, el actor del negocio y su justificación. **[27]**

Actor del negocio	Justificación
Secretario General del Comité UJC	Es la persona interesada en toda la información de los comités de base, ya sea en el funcionamiento o en la cotización.

Tabla 2.1 Descripción del Actor del Negocio.

2.4.2 – Trabajadores del negocio.

Un **trabajador del negocio** es una abstracción de una persona (o grupo de personas), una máquina o un sistema automatizado; que actúa en el negocio realizando una o varias actividades, interactuando con otros trabajadores del negocio y manipulando entidades del negocio. Representa un rol. A continuación se muestra en la tabla 2.2 los trabajadores del negocio y su justificación. **[27]**

Trabajador del negocio	Justificación
Secretario de Cotización.	Es el encargado de recoger la cotización de los comités de base a los cuales le asigna el comprobante de pago y le entrega un reporte al Secretario General de los comités de base que incumplen.
Secretario de Funcionamiento del Comité Primario.	Son los responsables de controlar las estadísticas a su respectivo nivel.
Secretario de Funcionamiento del Intituto	Controla toda la base estadística de la organizacion en el Intituto.
Secretario del Comité de Base	Es el encargado de llevar a cabo todas las actividades que se realizan en el comité de base.

Tabla 2.2 Descripción de los Trabajadores del Negocio

2.4.3 – Diagramas de casos de uso del negocio

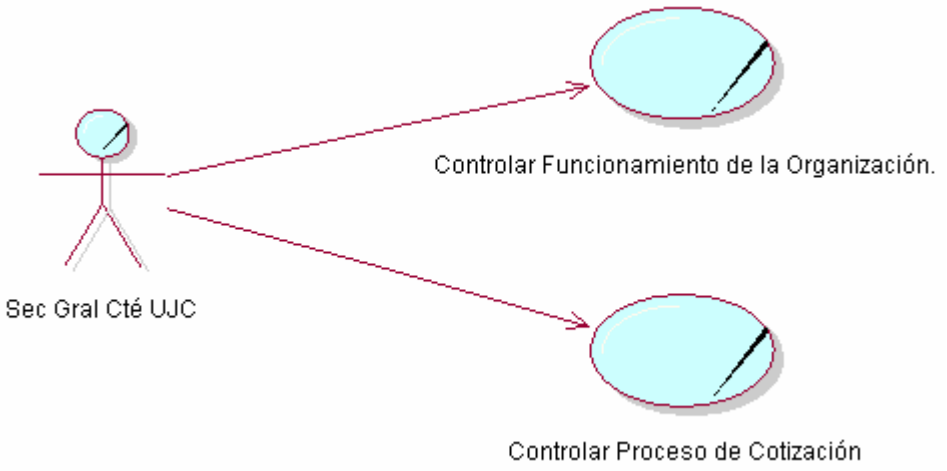


Figura 2.1 Diagrama de casos de uso del negocio.

2.4.4 – Descripción de los casos de uso del negocio.

Caso de uso del negocio: Controlar Funcionamiento de la Organización.
Actores del negocio: Secretario General del Comité de la UJC.
Propósito: Tener un control sobre cada una de las reuniones que se realizan en los diferentes Comités de Base. Registrar toda la información necesaria para analizar el cumplimiento y posteriormente mandar un reporte al Comité Provincial.
Resumen: El caso de uso se inicia cuando el secretario general del Comité UJC del instituto solicita los reportes de vida interna al secretario de funcionamiento, con los cuales se hace el balance general de los comités de base del instituto. El secretario de funcionamiento una vez recopilados todos los reportes de los comités primarios realiza el reporte general. Estos reportes recogen: militantes no incorporados, militantes no físicos, ausentes injustificados y justificados con sus respectivas causas, actas entregadas fuera de fecha y mes, datos de los militantes sancionados, actas entregadas y por entrar, reuniones no visitadas, reuniones sin fecha, etc.

El caso de uso finaliza cuando el secretario general del comité UJC obtiene los reportes solicitados al secretario de funcionamiento.	
Curso normal de los eventos:	
Acción del actor	**Respuesta del Negocio**
1- El Secretario General del Comité UJC del instituto solicita los reportes de vida interna.	2- El Secretario de Funcionamiento busca la información solicitada pero esta no se encuentra.
	3- El Secretario de Funcionamiento se reúne con todos los Secretarios de Funcionamiento del Comité Primario y solicita los reportes.
	4- Los Secretarios de Funcionamiento del Comité Primario se reúnen con los Secretarios de los Comités de Base correspondientes y pide la información correspondiente a cada Comité de Base.
	5- El Secretario del Comité de Base le entrega la información al Secretario de Funcionamiento del Comité Primario.
	6- El Secretario de Funcionamiento del Comité Primario le entrega la información al Secretario de Funcionamiento del Instituto.
	7- El Secretario de Funcionamiento del Instituto le entrega los reportes solicitados al Secretario General del Comité de la UJC.
8- El Secretario General del Comité de la UJC recibe respuesta de la solicitud	

pedida.	
Curso Alternativo de los eventos:	
Paso 2	Encontró los reportes solicitados por el Secretario General del Comité de la UJC y pasa al paso 8.
Prioridad: Desarrollar un software que automatice los distintos problemas que tiene la organización.	
Mejoras: Con la implementación de este mecanismo de gestión de la información en un sistema informático es posible reducir en tiempo y en recursos, se garantiza la confiabilidad de la información y el sistema podrá ser consultado estando dentro del Instituto o fuera del mismo, ya sea fuera de la provincia o de la nacionalidad cubana. Además los informes se verán en formatos Web a través de la red.	

Caso de uso del negocio: Controlar Proceso de Cotización
Actores del negocio: Secretario General del Comité de la UJC.
Propósito: Controlar el pago de la cotización.
Resumen: El caso de uso se inicia cuando el secretario general de Comité UJC del instituto solicita al secretario de cotización, cuales son los comités de base que no han cotizado hasta el momento o no lo han hecho completamente, es decir, cuántos comités de base han cumplido en tiempo el pago, cuáles faltan por cumplir y cuáles son las causas de los que no han entregado completo el dinero. El caso de uso finaliza cuando el secretario general del comité UJC obtiene los reportes solicitados.

Curso normal de los eventos:	
Acción del actor	**Respuesta del Negocio**
1- El Secretario General del Comité de la UJC solicita la cotización del mes.	2- El Secretario de Cotización del Instituto busca si ya todos los Comités de Base han cotizado pero esta no esta completa.
	3- El Secretario de Cotización del Instituto se reúne con los Secretarios de los distintos Comités de Base que no han cotizado
	4- Los Secretarios del Comité de Base le entrega la cotización a los Secretarios del Cotización del Instituto.
	5- El Secretario de Cotización del Instituto le hace entrega de la cotización al Secretario General del Comité de la UJC.
6- El Secretario General del Comité de la UJC recibe la cotización y su comprobante de pago.	
Curso Alternativo de los eventos:	
Paso 2	El Secretario de Cotización le entrega la cotización al Secretario General del Comité de la UJC y pasa al paso 6.
Prioridad:	
Mejoras: Con la implementación de este mecanismo es posible tener un mayor control sobre el pago de la cotización.	

2.4.5 – Diagramas de actividades del negocio.

El diagrama de actividad es un grafo que contiene los estados en que puede hallarse la actividad a analizar. Cada estado de la actividad representa la ejecución de una sentencia de un procedimiento, o el funcionamiento de una actividad en un flujo de trabajo. En resumen describe un proceso que explora el orden de las actividades que logran los objetivos del negocio. **[28]**

Los diagramas de Actividad del Funcionamiento de la Organización y el de Actividad del Proceso de Cotización recogen el orden de las acciones que realizan los actores y trabajadores en el negocio, dichos diagramas se muestran en las figuras 1 y 2 respectivamente.

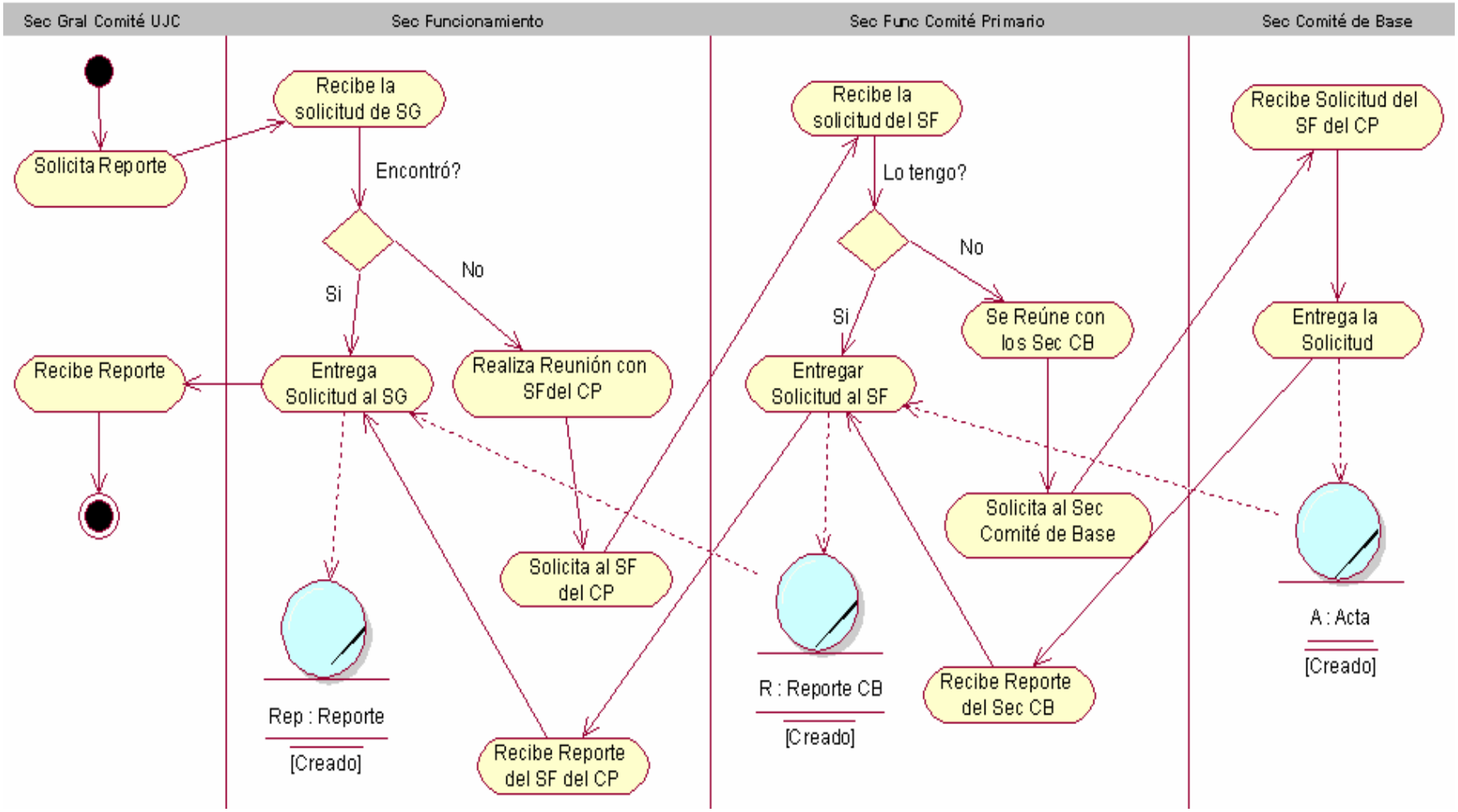


Figura 2.2 Diagrama de Actividad del Funcionamiento de la Organización.

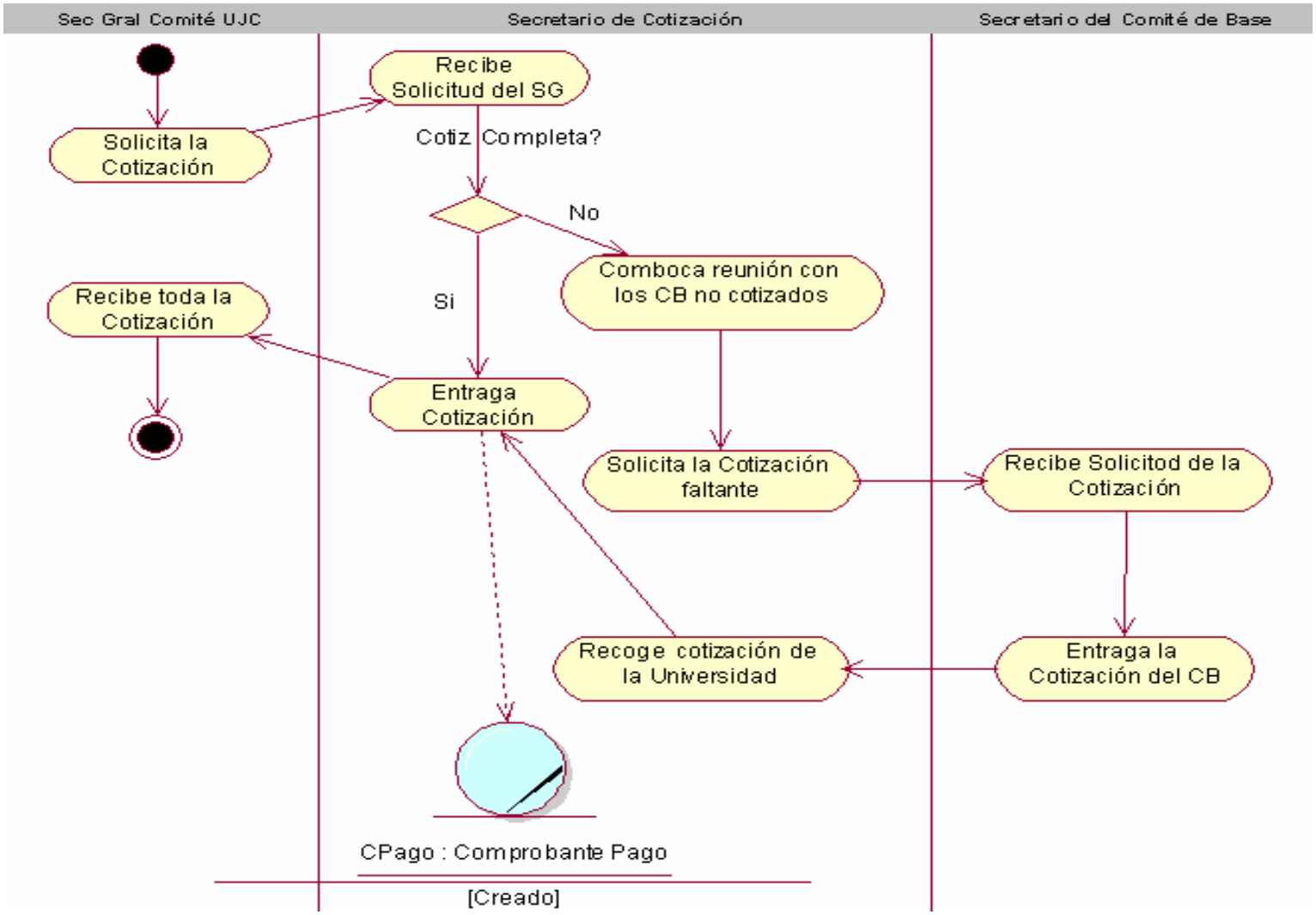


Figura 2.3 -Diagrama de Actividad del Proceso de Cotización.

2.5 – Modelo de objetos del negocio.

Un modelo de objetos del negocio es un modelo interno a un negocio. Describe como cada caso de uso del negocio es llevado a cabo por parte de un conjunto de trabajadores que utilizan un conjunto de entidades del negocio y unidades de trabajo **[29]**.

Una entidad del negocio representa algo, que los trabajadores toman, inspeccionan, manipulan, producen o utilizan en un caso de uso del negocio. El diagrama de clases del modelo de objeto, es un artefacto que se construye para describir el modelo de objetos del negocio.

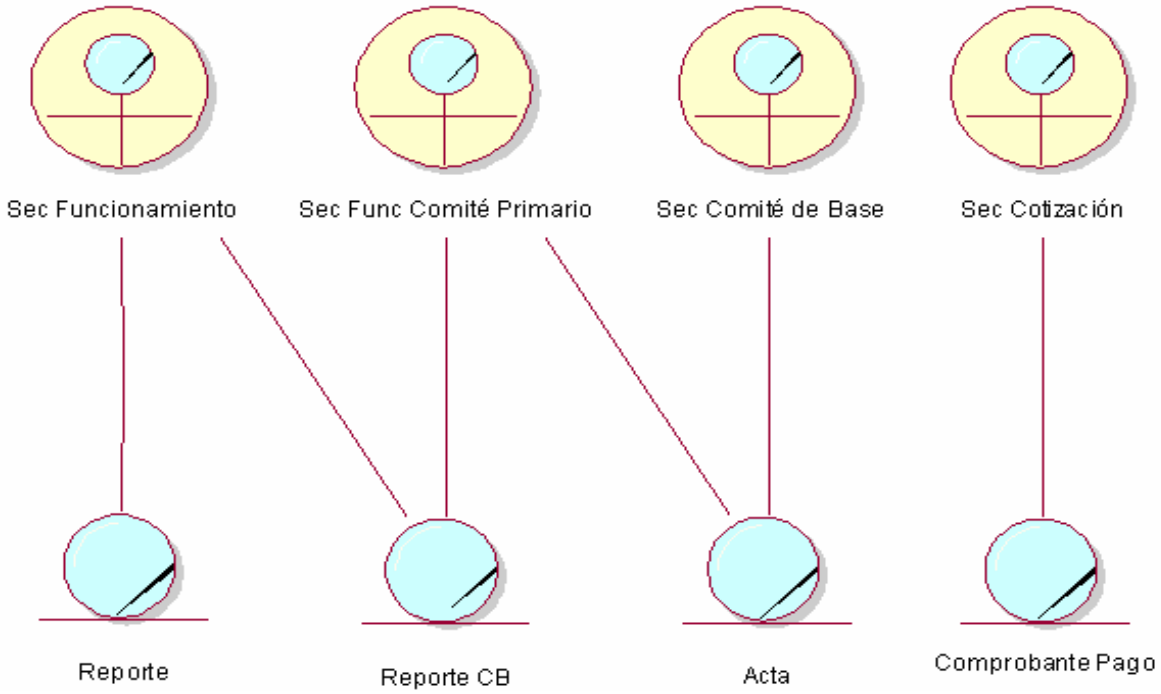


Figura 2.4 -Diagrama de clases del modelo de objetos.

2.6 – Descripción del sistema propuesto

2.6.1 – Concepción general del sistema

Este sistema propuesto pretende a través del desarrollo de una Aplicación Web automatizar el procesamiento de la información referente al funcionamiento de la Unión de Jóvenes Comunistas.

En esencia; el sistema se divide en:

1. En la recopilación y registro de la información de las reuniones de cada comité de base.
2. En el control administrativo y gestión de los militantes de cada comité de base.
3. En la entrega y actualización del pago de la cotización.

La responsabilidad de la recopilación, manejo y actualización del sistema es del Secretario General de la Universidad que hallan asignado por la organización.

2.6.2- Requerimientos funcionales

Los requerimientos funcionales permiten expresar una especificación más detallada de las responsabilidades del sistema que se propone. Ellos permiten determinar, de una manera clara, lo que debe hacer el mismo.

Los requerimientos funcionales del software propuesto son los siguientes:

1. Actualizar datos de los comités de base.
    1.1. Insertar comités de base.
    1.2. Modificar comités de base.
    1.3. Eliminar comités de base.
2. Actualizar datos de un estudiante no militante
    2.1 Insertar un nuevo estudiante no militante.
    2.2 Modificar los datos de un estudiante no militante.
    2.3 Eliminar los datos de un estudiante no militante.
3. Actualizar datos del plan de trabajo
    3.1 Insertar plan de trabajo.
    3.2 Modificar plan de trabajo.
    3.3 Eliminar plan de trabajo.
4. Actualizar pago de cotización
    4.1 Insertar pago de la cotización.
    4.2 Modificar pago de cotización.
    4.3 Eliminar pago de Cotización.
5. Actualizar lo datos de los militantes sancionados.

5.1. Modificar datos de lo militantes sancionados.

5.2. Eliminar datos de lo militantes sancionados.

6. Actualizar datos de una reunión.

6.1. Insertar nuevos datos de una reunión.

6.2. Modificar datos de una reunión.

6.3. Eliminar datos de una reunión.

7. Actualizar militantes.

7.1. Insertar a un nuevo militante estudiante.

7.2. Modificar datos de un militante estudiante.

7.3. Eliminar un militante estudiante.

7.4. Insertar a un nuevo militante trabajador.

7.5. Modificar datos de un militante trabajador.

7.6. Eliminar un trabajador

8. Dar baja a un militante.

9. Insertar sancionados.

10. Actualizar datos de facultad.

10.1. Insertar datos de una nueva facultad.

10.2. Modificar datos de una facultad.

10.3. Eliminar datos de una facultad.

11. Actualizar datos de especialidad.

11.1. Insertar datos de una nueva especialidad.

11.2. Modificar datos de una especialidad.

11.3. Eliminar datos de una especialidad.

12. Actualizar datos de áreas.

12.1. Insertar datos de una nueva áreas.

12.2. Modificar datos de áreas.

12.3. Eliminar datos de áreas.

13. Actualizar datos de departamentos.

13.1. Insertar datos de un nuevo departamento.

13.2. Modificar datos de departamentos.

13.3. Eliminar datos de departamentos.

14. Consultar ausentes.
15. Consultar militantes no físicos (no viene a las reuniones).
16. Consultar militantes no incorporados (esta aquí pero no ha traído el expediente)
17. Consultar listado por comité de base.
18. Consultar actas fueras de fecha.
19. Consultar militantes sancionados.
20. Consultar reuniones por fecha.
21. Consultar comités de base no cotizados hasta la fecha.
22. Actualizar usuario
    - 22.1 Insertar un nuevo usuario
    - 22.2 Modificar datos de un nuevo usuario
    - 22.3 Eliminar un usuario.
    - 25.4 Visualizar los datos de los usuarios

23 Autentificarse

24 Cambiar contraseña

2.6.3 – Requerimientos no funcionales

Requisitos de apariencia o interfaz externa.

La interfaz externa del sistema tendrá una apariencia profesional y estará basada en los estándares que usan las aplicaciones Web.

Deberá ser consistente con el mundo real de manera que los conceptos manejados sean conocidos y familiares por los usuarios para que les sea fácil su uso y aprendizaje.

La interfaz estará diseñada de modo tal que el usuario pueda tener en todo momento el control de la aplicación, lo que le permitirá ir de un punto a otro dentro de ella con gran facilidad. Se cuidará porque la aplicación sea lo más interactiva posible. La interfaz será uniforme de manera que a pesar de tener varios módulos, el usuario los identifique como un mismo sistema.

Requisitos de Usabilidad

Los usuarios del sistema quedan definidos por todos los estudiantes, profesores y personal no docente de la Universidad de Cienfuegos.

La explotación del sistema agilizaría el proceso de trabajo en general puesto que actividades como la obtención de datos y el procesamiento de los mismos, que anteriormente se realizaban manualmente, quedarían automatizadas en su totalidad.

El sistema sólo podrá ser utilizado por personas que posean los conocimientos básicos correspondientes con La Unión de Jóvenes Comunistas en el centro. Para responder a esta restricción, el sistema contará con una política de usuarios que impedirá accesos no autorizados que pudieran introducir errores en la información.

El sistema podrá ser usado por aquellas personas que no tengan experiencia en el uso de la computadora, sólo necesitarían un ligero entrenamiento sobre el funcionamiento de los principales elementos de una interfaz Web.

Requisitos de Rendimiento

Se concibe un sistema diseñado sobre la arquitectura cliente/servidor, de manera que se pueda contar con varios terminales dentro de la institución Universitaria.

En cuanto a los tiempos de respuesta, estos deben ser cortos para el procesamiento de las órdenes de trabajo.

La información deberá estar disponible las 24 horas del día, aunque una caída momentánea del sistema no provocará una crisis en el funcionamiento de la organización; sin embargo, el sistema deberá recuperarse en un corto período de tiempo ante una falla.

Requisitos de Soporte

El sistema deberá dar las posibilidades a futuras mejoras y nuevas opciones que se le quieran incorporar.

Los servicios de instalación y mantenimiento de la aplicación serán responsabilidad del Departamento de Informática, entidad productora del software.

Será necesario instalar un gestor de base de datos y un servidor Web para la difusión de la aplicación; se recomienda tener en cuenta que la implementación fue realizada utilizando el apache, sobre el sistema operativo Windows; y que el usuario pueda utilizar cualquier navegador que soporte JavaScript. Se requiere que la base de datos sea configurable teniendo en cuenta los cambios que puedan ocurrir en la organización.

Requisitos de Portabilidad

Las páginas están desarrolladas con html+javascript+php combinación que puede mantenerse en muchas plataformas. La base de datos esta en MySql y puede soportarse en Linux o Windows, los dos sistemas operativos más comunes en nuestra universidad. Debemos resaltar que la triología php+mysql+apache, además de ser multiplataforma están protegidos bajo la licencia GPL, lo cual posibilita su distribución.

Requisitos Políticos-culturales

Toda la información debe ser destinada a la superación y desarrollo de la militancia en las filas de la Unión de Jóvenes Comunistas. No se considera relevante que una persona de otra nacionalidad pueda visitar el sitio.

Requisitos de Confiabilidad

Es importante que el sistema presente un mecanismo de respuesta rápida ante fallos y que en caso de ocurrencia se minimicen las pérdidas de información.

Requisitos de Interfaz interna

El desktop debe estar configurado con una resolución de 800x600.

Requerimiento de Software

Se debe disponer de Windows NT o XP Server para instalar el servicio Web y para la base de datos un Servidor MySQL. Para la instalación del cliente se requiere Windows 2000 o superior.

Requerimiento de Hardware

Para el desarrollo y puesta en práctica del proyecto se requieren máquinas potentes, por ejemplo los requerimientos mínimos para un servidor Web son:

- Procesador PENTIUM
- 144 MB de RAM
- 4,2 GB de HDD
- Tarjeta de red de 100 Mbps.

Para las estaciones clientes se necesitan:

- Procesador Pentium o superior
- 96 Mbyte de RAM
- 2,1 Gbyte de HDD
- Todos los periféricos necesarios
- Tarjeta de red de 100 Mbps.

Incluso con un servidor de bases de datos y uno para servidor de la aplicaciones, sólo se necesitarían para las estaciones clientes, una máquina con un "Browser" (Internet Explorer).

Restricciones en el Diseño y la Implementación

Una restricción considerable, es el grado de informatización de la red de la Universidad de Cienfuegos, que aunque se presentan algunas deficiencias, existe una voluntad política de resolver el problema.

Solo con el mejor aprovechamiento del hardware se puede obtener una respuesta rápida sin problemas de funcionamiento del sistema.

Requisitos de Seguridad

Se debe garantizar un control estricto sobre la seguridad de la información teniendo en cuenta el establecimiento de niveles de acceso. No se deben permitir accesos sin autorización al sistema de base de datos. Además se debe definir una política de usuarios con roles y privilegios diferentes que garantice que la información pueda ser consultada de acuerdo al nivel de privilegios que puedan tener determinados grupos de usuarios.

Es también requisito de suma importancia garantizar la integridad de los datos que se almacenen en el servidor. La información almacenada deberá ser consistente y se utilizarán validaciones que limiten la entrada de datos irreales.

La información deberá estar disponible a los usuarios en todo momento, limitada solamente por las restricciones que estos tengan de acuerdo a la política de seguridad del sistema.

El software permite que a la información solo acceda quien este autorizado para ello y para el uso a que está autorizado, identifica a la persona a la que se autoriza, es decir, a quien se reconceda permiso para las determinadas tareas, permite que las contraseñas de los usuarios se almacenen y viajen por la red cifradas.

2.7 – Modelo de casos de uso del sistema

Casos de Uso del Sistema

Cada forma en que los actores usan el sistema se representa con un Caso de Uso. Los Casos de Uso son “fragmentos” de funcionalidad que el sistema ofrece para aportar un resultado de valor para sus actores. De manera más precisa, un Caso de Uso especifica una secuencia de acciones que el sistema puede llevar a cabo interactuando con sus actores, incluyendo alternativas dentro de la secuencia **[28].**

2.7.1 – Actores del sistema

Actor del sistema	Justificación
Administrador	Es la persona encargada de modificar el sistema así como de controlar la seguridad del mismo.
Usuario	Es la persona que le interesa ver los diferentes reportes y consultar los documentos de la organización.

Tabla 2.3 Descripción de los actores del sistema.

2.7.2 – Paquetes y sus relaciones

Un sistema grande debe dividirse en unidades más pequeñas, de manera que se pueda ser entendido por los usuarios que deseen realizar consultas y para que los equipos de trabajo puedan trabajar de manera independiente.

Para satisfacer los objetivos propuestos y según los requerimientos planteados, el sistema

que se propone debe estar conformado por tres módulos (figura 2.2):

1. Actualizar datos.
2. Reportes.
3. Seguridad.

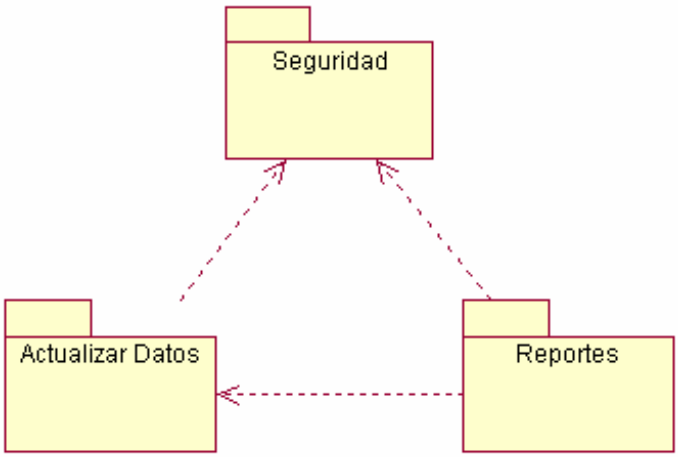


Figura 2.5 Diagrama de Paquetes.

Para hacer el sistema mas seguro, Actualizar Datos y Reportes dependen de Seguridad ya que si una persona no autorizada entra a la aplicación puede causar problemas graves.
Para la utilización del sistema se han definido tres roles, suficientes para lograr la efectividad del mismo, es decir, un usuario puede actuar como Administrador o Usuario.

El módulo de *Actualizar Datos* solo puede ser utilizado por Administrador, este ya debe estar registrado en el sistema. Este módulo es el encargado de hacer todas las entradas que el sistema requiere, es decir que el Administrador puede insertar datos, modificarlos o eliminarlos, no se creo otro usuario que realizara esta tarea para una mayor seguridad del sistema.
Al módulo de *Reportes* pueden acceder los usuarios que previamente hallan sido registrados, ya que en él se encuentran todos los reportes que necesitan conocer los directivos de la organización.

Por último el módulo de *Seguridad* que estará a cargo del usuario Administrador, controlará a todos los usuarios, como máximo responsable tendrá a su cargo la decisión

de quienes serán los Usuarios del sistema, además de poder cumplir cualquier rol, sin ningún tipo de problema.

2.7.3 – Diagramas de casos de uso del sistema

Diagrama de casos de uso del sistema a automatizar. Se incluyen las relaciones de *include*, *extend* y generalización/especialización entre los casos de uso, y de generalización/especialización entre los actores. Si se emplearon paquetes, elaborar un diagrama para cada paquete. No olvidar los casos de uso asociados a la seguridad.

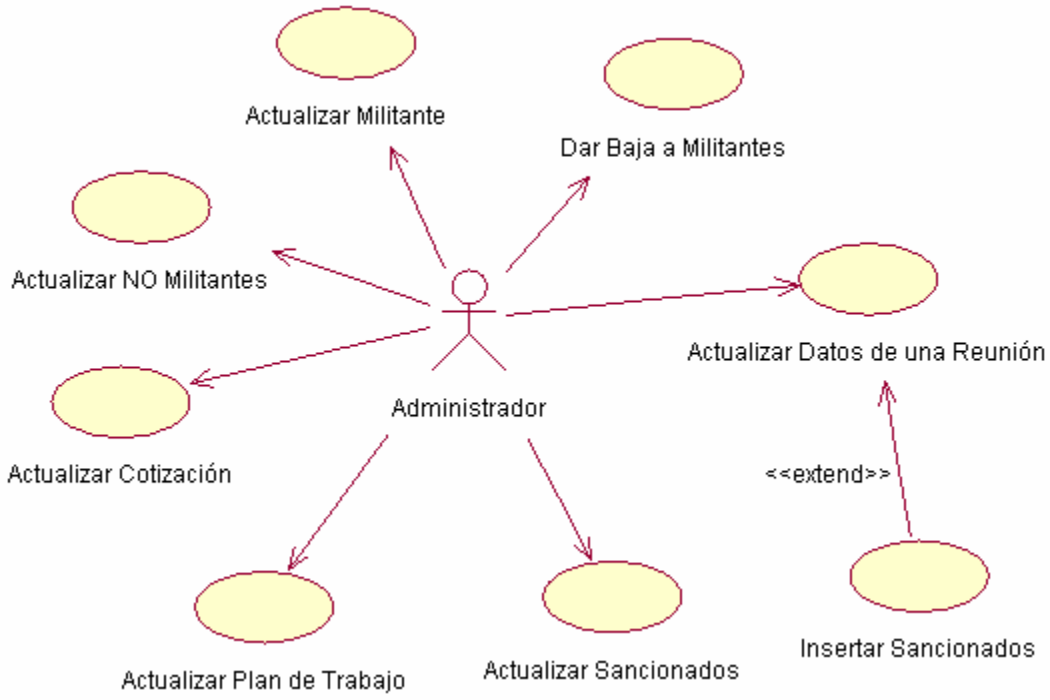


Figura 2.6 Diagramas de casos de uso del sistema del módulo Actualizar Datos.

2.7.4 – Descripción de los casos de uso del sistema.

Módulo Actualizar Datos.

Caso de uso:	Actualizar estudiante no militante.
Actores:	Administrador (inicia).
Propósito:	Mantener actualizado los datos de los estudiantes no militantes
Resumen: El caso de uso se inicia cuando un usuario registrado como Administrador accede al sistema. A través del menú escoge la opción de Actualizar No Militantes, se visualiza una pantalla donde aparecen los datos que son necesarios para insertar un estudiante no militante. Si lo que desea hacer es modificar o eliminar sus datos, en la pantalla de insertar se muestran todos los estudiantes no militantes donde puedes modificarlo o eliminarlo con el botón modificar o eliminar respectivamente. Para modificar se debe cambiar el campo que vas a modificar y presionar el botón Aceptar, para eliminar debes marcar el estudiante no militante y presionar el botón Eliminar. Al presionar los botones de Aceptar o Eliminar, el sistema registra la actualización correspondiente. El sistema devuelve un mensaje informando si se realizó la actualización satisfactoriamente o si por el contrario ocurriera algún problema, muestra un mensaje indicando el error. Los datos que se recogen de los estudiantes son: carné de identidad, nombre, primer apellido, segundo apellido, dirección particular, sexo, grupo, año, teléfono, correo electrónico, especialidad, facultad, municipio, provincia, y número del comité de base al que pertenece. El caso de uso finaliza cuando el sistema registra la actualización correspondiente.	
Referencias:	R2

Precondiciones:	Tiene que estar registrado en la base de datos el comité de base.
Poscondiciones:	
Requisitos Especiales:	Para modificar y eliminar tiene que estar registrado el estudiante no militante.
Prototipo.	

Caso de uso:	Actualizar datos del plan de trabajo.
Actores:	Administrador (Inicia)
Propósito:	Mantener actualizado los datos del plan de trabajo
Resumen: El caso de uso se inicia cuando un usuario registrado como Administrador accede al sistema. A través del menú escoge la opción de Actualizar Plan de Trabajo, se visualiza una pantalla donde aparecen los datos que son necesarios para insertar el plan de trabajo. Cuando desees modificar o eliminar algún plan de trabajo, presione los botones correspondientes. Al presionar los botones de Aceptar o Eliminar respectivamente, el sistema registra la actualización correspondiente. El sistema devuelve un mensaje informando si se realizó la actualización satisfactoriamente o si por el contrario ocurriera algún problema, muestra un mensaje indicando el error. Los datos que se recogen en el plan de trabajo son número del comité de base, número de la actividad, fecha, hora, lugar, responsables, participantes, objetivos y aseguramientos de la actividad. El caso de uso finaliza cuando el sistema registra la actualización correspondiente.	
Referencias:	R3
Precondiciones:	Tiene que estar registrado en la base de datos el comité de base.
Poscondiciones:	

Requisitos Especiales:	Para modificar y eliminar tiene que estar registrada la actividad en el plan de trabajo.
Prototipo:	

Caso de uso:	Actualizar cotización.
Actores:	Administrador (Inicia).
Propósito:	Mantener actualizado los datos del pago de la cotización.
Resumen: El caso de uso se inicia cuando un usuario registrado como Administrador accede al sistema. A través del menú escoge la opción de Actualizar Cotización, se visualiza una pantalla donde aparecen los datos que son necesarios para insertar, modificar o eliminar la cotización. Una vez llenado todos los datos al presionar los botones de Aceptar o Eliminar respectivamente, el sistema registra la actualización correspondiente. El sistema devuelve un mensaje informando si se realizó la actualización satisfactoriamente o si por el contrario ocurriera algún problema, muestra un mensaje indicando el error. Los datos que se recogen en el pago de la cotización son número del comité de base, fecha de entrega, importe total y mes. El caso de uso finaliza cuando el sistema registra la actualización correspondiente.	
Referencias:	R4
Precondiciones:	Tiene que estar registrado en la base de datos el comité base.
Poscondiciones:	
Requisitos Especiales:	Para modificar y eliminar tiene que estar registrado el pago de la cotización.
Prototipo:	

Caso de uso:	Actualizar sancionados.
Actores	Administrador (Inicia).
Propósito:	Mantener actualizado los datos de los militantes sancionados.
Resumen: El caso de uso se inicia cuando un usuario registrado como Administrador accede al sistema. A través del menú escoge la opción de Actualizar Sancionados, se visualiza una pantalla donde se debe insertar el número de carné de identidad del militante, el comité de base al que pertenece, el tipo de sanción y la fecha en que se realizó. Si lo que desea hacer es modificar o eliminar sus datos, puede hacerlo a través de lo botones correspondientes. Al presionar los botones de Aceptar o Eliminar, el sistema registra la actualización correspondiente. El sistema devuelve un mensaje informando si se realizó la actualización satisfactoriamente o si por el contrario ocurriera algún problema, muestra un mensaje indicando el error. Los datos que se recogen de los militantes sancionados son: número de carné de identidad del militante, el tipo de sanción, la fecha de impuesta la sanción y el comité de base al que pertenece. El caso de uso finaliza cuando el sistema registra la actualización correspondiente.	
Referencias:	R5
Precondiciones:	
Poscondiciones:	
Requisitos Especiales:	Para modificar y eliminar tiene que estar registrado el militante sancionado.
Prototipo:	

Caso de uso:	Actualizar datos de una reunión.
Actores	Administrador (Inicia).
Propósito:	Mantener actualizado los datos de las reuniones que se realizan.

Resumen:

El caso de uso se inicia cuando un usuario registrado como Administrador accede al sistema. A través del menú escoge la opción de Actualizar Reunión, se visualiza una pantalla donde aparecen los datos que son necesarios para insertar las reuniones.

Si lo que desea hacer es modificar o eliminar una reunión, en la pantalla se muestran los botones modificar y eliminar. Para modificar se debe cambiar el campo que vas a modificar y presionar el botón Aceptar, para eliminar debes marcar la reunión y presionar el botón Eliminar. Al presionar los botones de Aceptar o Eliminar, el sistema registra la actualización correspondiente.

El sistema devuelve un mensaje informando si se realizó la actualización satisfactoriamente o si por el contrario ocurriera algún problema, muestra un mensaje indicando el error.

Todos los campos son obligatorios llenarlos, porque de ellos dependen todos los reportes que necesita la organización.

Los datos de una reunión son: nombre del comité de base, fecha de realización, tema de instrucción política, carácter de la reunión, cantidad de personas que asisten a la reunión, nombre de los ausentes y sus causas, si la reunión es visitada o no, cantidad de sancionados, cantidad de acuerdos tomados y chequeados, así como la fecha de entrega de las Actas.

El caso de uso finaliza cuando el sistema registra la actualización correspondiente.

Referencias:	R 6 Tiene asociado el caso de Uso Insertar Sancionados (extend)
Precondiciones:	Tiene que estar registrado en la base de datos el comité de base

Poscondiciones:	.
Requisitos Especiales:	Para modificar y eliminar tiene que estar registrada la reunión.
Prototipo:	

Caso de uso:	Actualizar militante.
Actores	Administrador (Inicia).
Propósito:	Mantener actualizados los datos de los militantes.
Resumen: El caso de uso se inicia cuando un usuario registrado como Administrador accede al sistema. A través del menú escoge la opción de Actualizar Militantes, se visualiza una pantalla para escoger insertar un militante estudiante o un militante trabajador. En dependencia de la acción que se escoja se muestran los campos que son necesarios para insertar los datos del militante. Si lo que desea hacer es modificar o eliminar sus datos, en la pantalla de insertar se muestran los estudiantes militantes donde puedes modificarlo con la opción modificar o eliminar con sus respectivos botones. Para modificar se debe cambiar el campo que vas a modificar y presionar el botón Aceptar, para eliminar debes marcar el estudiante no militante y presionar el botón Eliminar. Al presionar los botones de Aceptar o Eliminar, el sistema registra la actualización correspondiente. El sistema devuelve un mensaje informando si se realizó la actualización satisfactoriamente o si por el contrario ocurriera algún problema, muestra un mensaje indicando el error. Los datos de un militante son: comité de base al que pertenece, carné de identidad, nombre, primer apellido, segundo apellido, dirección particular, sexo, grupo, año, teléfono, correo electrónico, especialidad, municipio y provincia. Si es un trabajador los datos son: los mismos de un estudiante más el área y el departamento al que pertenecen, si es militante del partido o no.	
Referencias:	R7
Precondiciones:	Tiene que estar registrado en la base de datos el

	comité de base.
Poscondiciones:	
Requisitos Especiales:	Para modificar y eliminar tiene que estar registrado el militante.
Prototipo:	

Caso de uso:	Dar baja a un militante.
Actores	Administrador (Inicia)
Propósito:	Mantener actualizado los datos de baja de los militantes.
Resumen: El caso de uso se inicia cuando un usuario registrado como Administrador accede al sistema. A través del menú escoge la opción de Dar Baja, se visualiza una pantalla donde debe insertarse el número de carné de identidad, si existe dicho militante, aparece una ventana donde debe ser insertado el lugar de destino. Una vez llenado los datos correspondientes al presionar el botón Aceptar el sistema registra la actualización correspondiente. El sistema devuelve un mensaje informando si se realizó la actualización satisfactoriamente o si por el contrario ocurriera algún problema, muestra un mensaje indicando el error. Para poder efectuar este proceso es obligatorio entrar el destino del militante, mientras esto no suceda, el militante seguirá siendo miembro del comité de base al que correspondía. Los datos que se recogen para darle de baja a un militante son: número del comité de base, carné de identidad, destino y fecha del traslado. El caso de uso termina cuando el sistema registra la actualización correspondiente.	
Referencias:	R8
Precondiciones:	El militante debe existir en la base de datos.
Poscondiciones:	

Requisitos Especiales:	Tiene que estar registrado el militante.
Prototipo:	

Módulo Reportes.

A continuación mostramos el diagrama de casos de uso del sistema correspondiente al módulo (Figura 2.7) y la explicación de los casos de uso correspondientes:

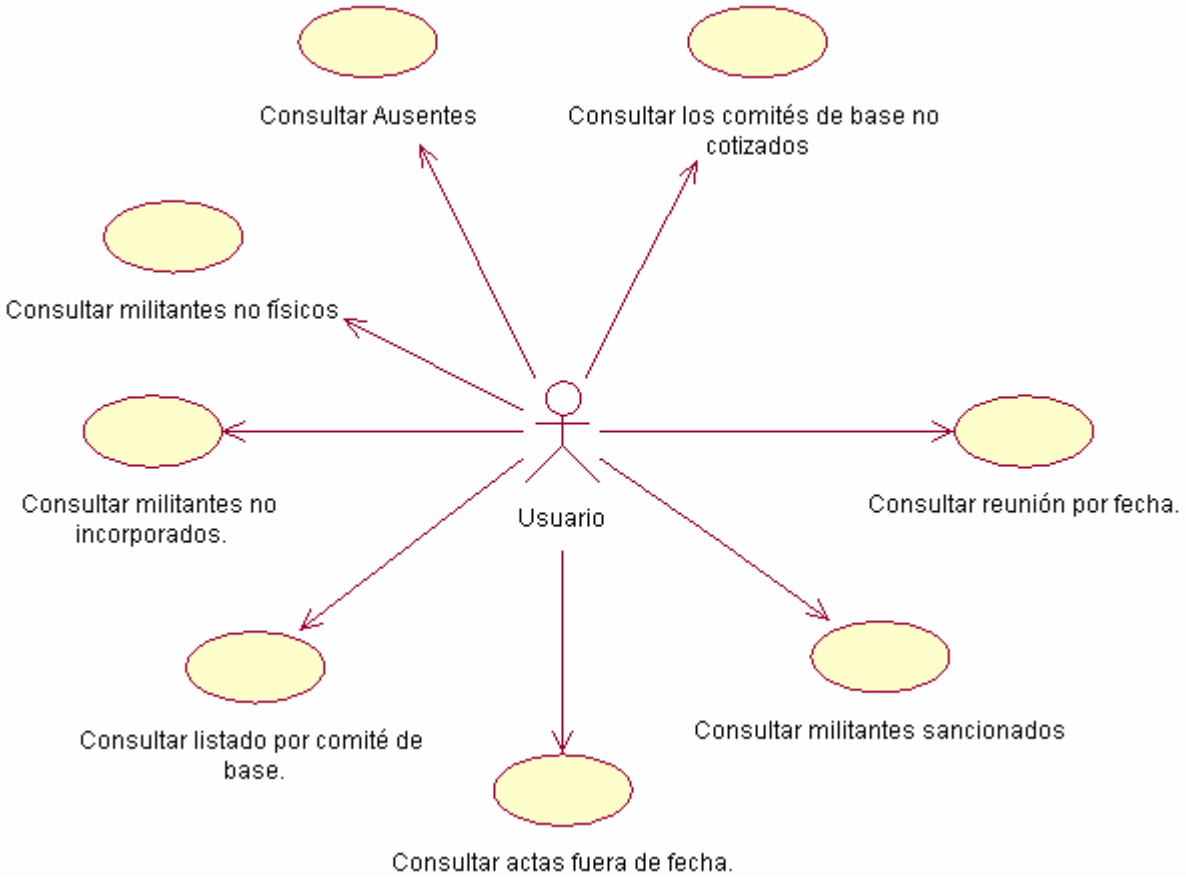


Figura 2.7 Diagrama de casos de uso del Sistema del módulo Reportes.

Caso de uso:	Consultar ausentes.
Actores	Usuario (Inicia).
Propósito:	Publicar los ausentes del mes.
Resumen: El caso de uso se inicia cuando un usuario registrado como Usuario accede al sistema. A través del menú escoge la opción de Mostrar Ausentes, se visualiza una pantalla donde aparecen cuales fueron los ausentes del mes y las causas. El sistema extrae la información solicitada automáticamente, finalizando así el caso de uso.	
Referencias:	R14
Precondiciones:	
Poscondiciones:	Se visualizan los ausentes.
Requisitos Especiales:	Tiene que estar registrada una reunión.
Prototipo:	

Caso de uso:	Consultar militantes no físicos
Actores	Usuario (Inicia)
Propósito:	Publicar los militantes no físicos
Resumen: El caso de uso se inicia cuando un usuario registrado como Usuario accede al sistema. A través del menú escoge la opción de Mostrar No Físicos, se visualiza una pantalla donde aparecen cuales son los militantes no físicos que actualmente tiene el instituto. El sistema extrae la información solicitada automáticamente, finalizando así el caso de uso.	
Referencias:	R15
Precondiciones:	
Poscondiciones:	Se visualizan los militantes no físicos.

Requisitos Especiales:	Tiene que estar registrada una reunión
Prototipo:	

Caso de uso:	Consultar militantes no incorporados.
Actores	Usuario (Inicia).
Propósito:	Publicar los militantes no incorporados.
Resumen: El caso de uso se inicia cuando un usuario registrado como Usuario accede al sistema. A través del menú escoge la opción de Mostrar No Incorporados, se visualiza una pantalla donde aparecen cuales son los militantes no incorporados que actualmente tiene el instituto. El sistema extrae la información solicitada automáticamente, finalizando así el caso de uso.	
Referencias:	R16
Precondiciones:	
Poscondiciones:	Se visualizan los militantes no incorporados
Requisitos Especiales:	Tiene que estar registrada una reunión.
Prototipo:	

Caso de uso:	Consultar listado por comité de base.
Actores	Usuario (Inicia).
Propósito:	Publicar listado por comité de base.
Resumen: El caso de uso se inicia cuando un usuario registrado como Usuario accede al sistema. A través del menú escoge la opción de Listar Militantes, se visualiza una pantalla donde aparece la lista de los militantes, entonces el sistema muestra los datos correspondientes automáticamente, finalizando así el caso de uso.	
Referencias:	R17

Precondiciones:	
Poscondiciones:	Se visualizan los militantes correspondientes al comité de base.
Requisitos Especiales:	Tiene que existir el comité de base.
Prototipo:	

Caso de uso:	Consultar actas fuera de fecha.
Actores	Usuario (Inicia).
Propósito:	Publicar las actas fuera de fecha.
Resumen: El caso de uso se inicia cuando un usuario registrado como Usuario accede al sistema. A través del menú escoge la opción de Mostrar Actas Fuera de Fecha, se visualiza una pantalla donde aparecen cuales son las actas que se entregaron después de las 72 horas que están establecidas por la organización. El sistema extrae la información solicitada automáticamente, finalizando así el caso de Uso.	
Referencias:	R18
Precondiciones:	
Poscondiciones:	Se visualizan las actas fuera de fecha.
Requisitos Especiales:	Tienen que estar registrada una reunión.
Prototipo:	

Caso de uso:	Consultar militantes sancionados.
Actores	Usuario (inicia).
Propósito:	Publicar los militantes sancionados.
Resumen: El caso de uso se inicia cuando un usuario registrado como Usuario accede al	

sistema. A través del menú escoge la opción de Mostrar Militantes Sancionados, se visualiza una pantalla donde aparecen cuales fueron los militantes sancionados del mes y sus causas. El sistema extrae la información solicitada automáticamente, finalizando así el caso de uso.	
Referencias:	R19
Precondiciones:	
Poscondiciones:	Se visualizan los militantes sancionados.
Requisitos Especiales:	Tienen que estar registrada una reunión.
Prototipo:	

Caso de uso:	Consultar reunión por fecha.
Actores	Usuario (Inicia).
Propósito:	Publicar las reuniones del día.
Resumen: El caso de uso se inicia cuando un usuario registrado como Usuario accede al sistema. A través del menú escoge la opción de Mostrar Reuniones por Fechas, se visualiza una pantalla donde aparecen cuales son la reuniones correspondientes a esa fecha. El sistema extrae la información solicitada automáticamente, finalizando así el caso de uso.	
Referencias:	R20
Precondiciones:	
Poscondiciones:	Se visualizan las reuniones por fecha.
Requisitos Especiales:	Tienen que estar registrada una reunión.
Prototipo:	

Caso de uso:	Consultar los comités de base no cotizados
Actores	Usuario (Inicia).

Propósito:	Publicar los comités de base no cotizados.
Resumen: El caso de uso se inicia cuando un usuario registrado como Usuario accede al sistema. A través del menú escoge una opción de Mostrar no Cotizados, se visualiza una pantalla donde aparecen cuales son los comités de base que hasta el momento no han efectuado el pago de la cotización correspondiente al mes. El sistema extrae la información solicitada automáticamente, finalizando así el caso de uso.	
Referencias:	R21
Precondiciones:	
Poscondiciones:	
Requisitos Especiales:	
Prototipo:	

Módulo Seguridad.

A continuación mostramos el diagrama de casos de uso del sistema correspondiente al módulo (Figura 2.8) y la explicación de los casos de uso correspondientes.

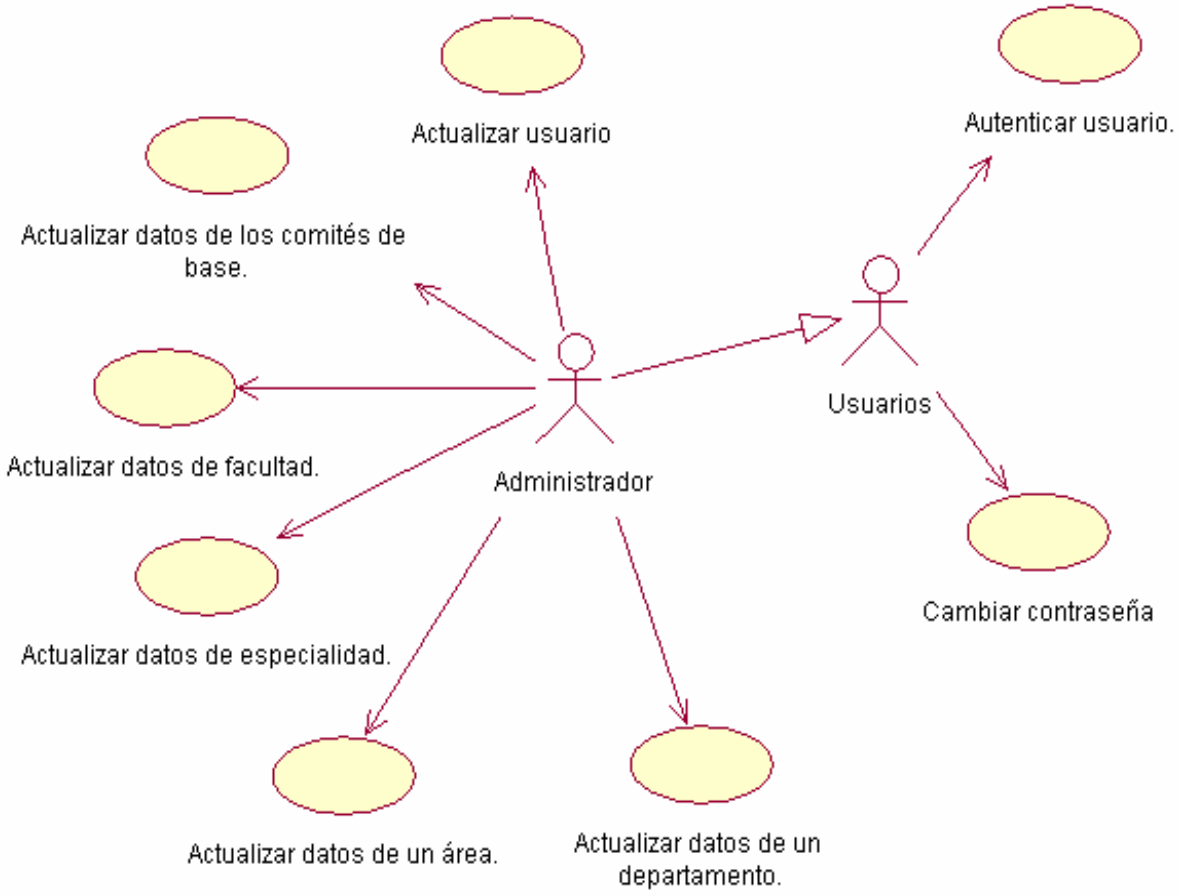


Figura 2.8 Diagrama de casos de uso del Sistema del módulo Seguridad.

Caso de uso:	Actualizar usuario
Actores	Administrador (Inicia).
Propósito:	Mantener actualizados los datos de lo usuarios registrados en el sistema.
Resumen: El caso de uso se inicia cuando un usuario registrado como administrador accede al sistema. A través del menú escoge la opción de Actualizar Usuario, se visualiza una pantalla donde aparecen los datos que son necesarios para insertar, modificar o eliminar los usuarios.	

Una vez llenado todos los datos al presionar los botones de aceptar, Modificar o Eliminar respectivamente, el sistema registra la actualización correspondiente. El sistema devuelve un mensaje informando si se realizó la actualización satisfactoriamente o si por el contrario ocurriera algún problema, muestra un mensaje indicando el error. Los datos de un usuario son: nombre con el que se identificará el usuario para poder acceder al sistema (alias), contraseña, nombre y apellido, cargo y nivel. El caso de uso finaliza cuando el sistema registra la actualización correspondiente.	
Referencias:	R22
Precondiciones:	
Poscondiciones:	
Requisitos Especiales:	Para modificar y eliminar tiene que estar registrado el usuario.
Prototipo:	

Caso de uso:	Autenticar usuario.
Actores	Usuario (Inicia)
Propósito:	Identificar al usuario para personalizar las opciones de información restringidas en dependencia del acceso que tenga asignado.
Resumen: El caso de uso se inicia cuando el usuario desea identificarse para poder acceder a las opciones de información restringidas, el sistema verifica la validez de los datos de acceso. Si no son válidos se le informa al usuario, de lo contrario se inicia la sesión, finalizando así el caso de uso. Los datos requeridos para la autenticación son: usuario y contraseña.	
Referencias:	23
Precondiciones:	
Poscondiciones:	Se inicia la sesión del usuario.
Requisitos Especiales:	

Prototipo:

Caso de uso:	Cambiar contraseña
Actores	Usuario (Inicia).
Propósito:	Modificar la contraseña de un usuario registrado.
Resumen: El caso de uso se inicia cuando el usuario desea modificar su contraseña, el sistema chequea que el cambio sea correcto. Si el cambio no es correcto se le informa al usuario, de lo contrario registra la nueva contraseña, finalizando así el caso de uso. Los datos requeridos para el cambio de contraseña son: usuario y contraseña	
Referencias:	24
Precondiciones:	
Poscondiciones:	Se modifica el atributo contraseña.
Requisitos Especiales:	Tiene que estar registrado el usuario.
Prototipo:	

Caso de uso:	Actualizar datos de los comités de base.
Actores	Administrador (Inicia)
Propósito:	Mantener actualizado los datos de los comités de base
Resumen: El caso de uso se inicia cuando un usuario registrado como Administrador acceda al sistema. A través del menú escoge la opción de Actualizar Comité de Base, se visualiza una pantalla donde aparecen los datos que son necesarios para insertar un comité de base. Si lo que deseas hacer es insertar un nuevo comité de base, lo puedo hacer en esta ventana, si lo que deseas hacer es modificar o eliminar alguno, el la pantalla de insertar aparece el listado, en el cual puedes modificar o eliminar.	

Si existe en la base de datos muestra sus datos para que puedan ser modificados o eliminados respectivamente. El sistema devuelve un mensaje informando si se realizó la actualización satisfactoriamente o si por el contrario ocurriera algún problema, muestra un mensaje indicando el error. Al presionar los botones de Aceptar, Modificar o Eliminar respectivamente, el sistema registra la actualización correspondiente. Los datos que se recogen de los comités de base son: número de comité de base, facultad a la que pertenece, nombre y apellidos del secretario general, secretario ideológico, secretario funcionamiento, responsable de beca. El caso de uso finaliza cuando el sistema registra la actualización correspondiente.	
Referencias:	R1
Precondiciones:	
Poscondiciones:	
Requisitos Especiales:	Para modificar y eliminar tiene que estar registrado el comité de base.
Prototipo:	

Caso de uso:	Actualizar datos de facultad.
Actores	Administrador (Inicia).
Propósito:	Mantener actualizado los datos de las facultades.
Resumen: El caso de uso se inicia cuando un usuario registrado como administrador accede al sistema. A través del menú escoge la opción de Actualizar Facultad, se visualiza una pantalla donde aparecen los datos que son necesarios para insertar, modificar o eliminar una facultad. Una vez llenado todos los datos al presionar los botones de Aceptar, Modificar o Eliminar respectivamente, el sistema registra la actualización correspondiente. El sistema devuelve un mensaje informando si se realizó la actualización	

satisfactoriamente o si por el contrario ocurriera algún problema, muestra un mensaje indicando el error. Los datos que se recogen de las facultades son: nombre. El caso de uso finaliza cuando el sistema registra la actualización correspondiente.	
Referencias:	R10
Precondiciones:	
Poscondiciones:	
Requisitos Especiales:	Para modificar o eliminar tiene que estar registrada la facultad.
Prototipo:	

Caso de uso:	Actualizar datos de especialidad.
Actores	Administrador (Inicia).
Propósito:	Mantener actualizado los datos de las especialidades.
Resumen: El caso de uso se inicia cuando un usuario registrado como administrador accede al sistema. A través del menú escoge la opción de Actualizar Especialidad, se visualiza una pantalla donde aparecen los datos que son necesarios para insertar, modificar o eliminar una especialidad. Una vez llenado todos los datos al presionar los botones de Aceptar, Modificar, o Eliminar respectivamente, el sistema registra la actualización correspondiente. El sistema devuelve un mensaje informando si se realizó la actualización satisfactoriamente o si por el contrario ocurriera algún problema, muestra un mensaje indicando el error. Los datos que se recogen de las especialidades son: nombre de la especialidad y facultad a la que pertenece. El caso de uso finaliza cuando el sistema registra la actualización correspondiente.	

Referencias:	R11
Precondiciones:	Tiene que estar registrado en la base de datos de la facultad
Poscondiciones:	
Requisitos Especiales:	Para modificar o eliminar tiene que estar registrada la especialidad
Prototipo:	

Caso de uso:	Actualizar datos de un área.
Actores	Administrador (Inicia).
Propósito:	Mantener actualizado los datos de un Área.
Resumen: El caso de uso se inicia cuando un usuario registrado como administrador accede al sistema. A través del menú escoge la opción de Actualizar Área, se visualiza una pantalla donde aparecen los datos que son necesarios para insertar, modificar o eliminar un área. Una vez llenado todos los datos al presionar los botones de Aceptar, Modificar, o Eliminar respectivamente, el sistema actualiza la actualización correspondiente. El sistema devuelve un mensaje informando si se realizó la actualización satisfactoriamente o si por el contrario ocurriera algún problema, muestra un mensaje indicando el error. Los datos que se recoge de las áreas son: nombre. El caso de uso finalizar cuando el sistema registra la actualización correspondiente.	
Referencias:	R12
Precondiciones:	
Poscondiciones:	
Requisitos Especiales:	Para modificar o eliminar tiene que estar registrada el área.
Prototipo:	

Caso de uso:	Actualizar datos de un departamento.
Actores	Administrador (Inicia).
Propósito:	Mantener actualizado los datos de los departamentos.
Resumen: El caso de uso se inicia cuando un usuario registrado como administrador accede al sistema. A través del menú escoge la opción de Actualizar Departamento, se visualiza una pantalla donde aparecen los datos que son necesarios para insertar, modificar o eliminar un área. Una vez llenado todos los datos al presionar los botones de Aceptar, Modificar, o Eliminar respectivamente, el sistema registra la actualización correspondiente. El sistema devuelve un mensaje informando si se realizó la actualización satisfactoriamente o si por el contrario ocurriera algún problema, muestra un mensaje indicando el error. Los datos que se recogen de los departamentos son: nombre del departamento y área a la que pertenece. El caso de uso finaliza cuando el sistema registra la actualización correspondiente.	
Referencias:	R13
Precondiciones:	Tienen que estar registrada en la base de datos el área.
Poscondiciones:	
Requisitos Especiales:	Para modificar y eliminar tiene que estar registrado el departamento.
Prototipo:	

2.8 – Diagrama de clases del diseño Web.

El diagrama de clases es el diagrama principal del diseño donde la estructura de clases del sistema se especifica, con relaciones entre clases y estructuras de herencia.

Durante el análisis, el diagrama se desarrolla buscando una solución ideal. Durante el

diseño, se usa el mismo diagrama, y se modifica para satisfacer los detalles de las implementaciones.

En el diagrama de clases del diseño se modelan las páginas, los enlaces entre estas, todo el código que irá creando las páginas, así como el contenido dinámico de estas una vez que estén en el navegador del usuario; es muy importante pues estos son los artefactos que se necesitan modelar para que el desarrollador los implemente y obtener así el producto final con la calidad requerida.

A continuación se muestran los diagramas de clases del diseño Web del sistema, divididos en sus respectivos paquetes, Se representan los procesos más importantes, ocultamos los atributos y las operaciones de las clases de la capa intermedia para facilitar la comprensión de los mismos.

2.8.1- Módulo Actualizar Datos.

Los diagramas de clases Web del módulo actualizar muestra los enlaces de las páginas, de las operaciones más relevantes que se realizan. (Véase Anexos 2)

2.8.2- Módulo Seguridad.

Los diagramas de clases Web del módulo Seguridad representan como se lleva a cabo la seguridad del sistema a través de los enlaces de las páginas. (Véase Anexo 3)

2.8.3- Módulo Reporte.

Los diagramas de clase Web del módulo Reporte representan como funciona el sistema a partir de una solicitud pedida. (Véase Anexo 4)

2.9 – Diseño de la base de datos

2.9.1 Diagrama de clases persistentes.

Las clases persistentes son las clases que necesitan ser capaces de guardar su estado en medio permanente, esta necesidad esta dada por el almacenamiento físico permanente de la información de la clase, y para realizar la copia de seguridad en caso de fracaso del sistema, o solo para el intercambio de información. (Véase el Anexo 5, el cual esta dividido por partes para una mejor visibilidad)

El modelo de los datos describe la representación lógica y física de datos persistentes en el sistema. También incluye cualquier conducta definida en la base de datos, como los procedimientos almacenados, triggers y constraints. (Véase el Anexo 6)

2.10 – Diagrama de implementación

El modelo de implementación describe como los elementos del modelo de diseño, como las clases, se implementan en términos de componentes. Describe también como se organizan los componentes de acuerdo con los mecanismos de estructuración y modularización disponibles en el entorno de implementación y en el lenguaje o lenguajes de programación utilizados y como dependen los componentes unos de otros. **[28]**

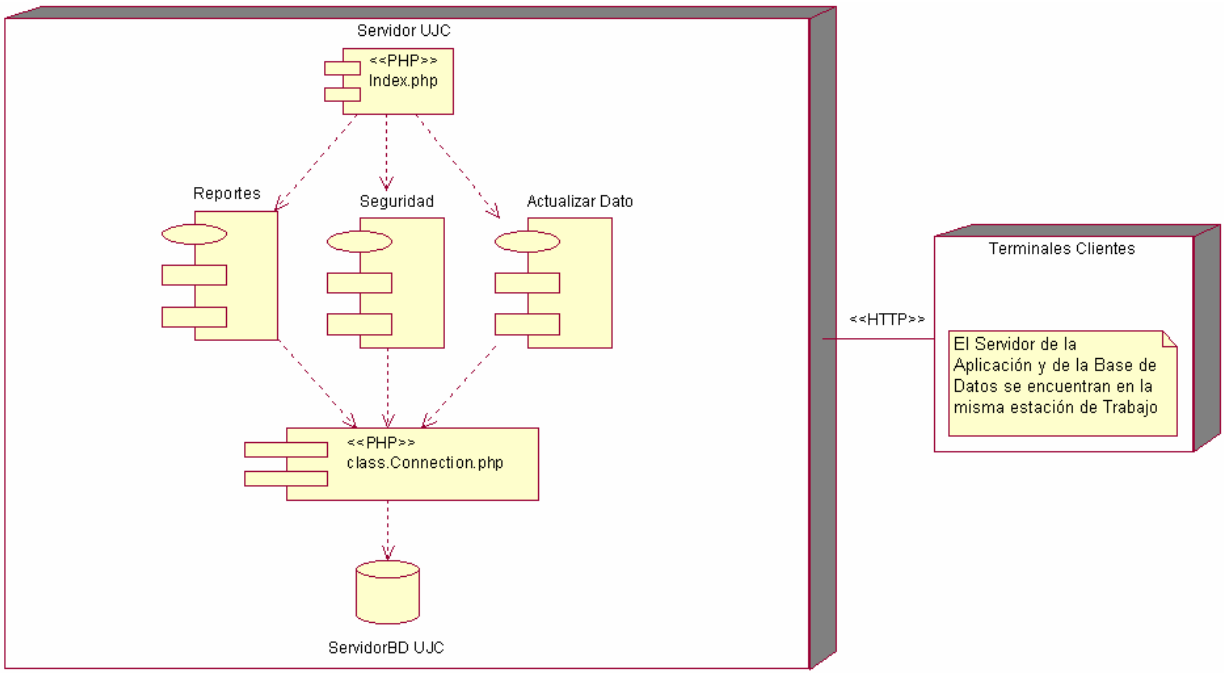


Figura 2.9 Diagrama de Implementación.

2.11 – Principios de diseño

2.11.1 – Estándares en la interfaz de la aplicación

El diseño de la interfaz es uno de los elementos "clave" en la realización del programa, se puede definir como: "el conjunto de trabajos y pasos que seguirá el usuario, durante todo el tiempo que se relacione con el programa, detallando lo que vera y escuchara en cada momento, y las acciones que realizara, así como las respuestas que el sistema le dará". **[30]**

La interfaz requiere, en si misma, un esfuerzo mental independiente del contenido que nos muestra. Todo esto supone una carga de memoria sumada por el usuario. Un buen sistema, por tanto, ha de requerir menos esfuerzos mentales de manejo del interfaz y concentrar la atención en el contenido que quiere transmitir. Como el objetivo fundamental en toda aplicación es el usuario, los Analistas de Sistemas, no solo deben considerar que debe hacer el sistema y cómo (computacionalmente) va a realizarlo, sino también considerar los principios de percepción y ergonomía que afectan al ser humane para lograr un producto que cumpla con los requerimientos y que se adapte a los usuarios del mismo. **[31]**

Para el desarrollo de nuestra interfaz tuvimos presente los siguientes aspectos:

1. Reducir la carga a la memoria
2. Diseño de diálogos que conducen a una conclusión.
3. Previsión de errores y manejo de errores simples.
4. Lograr deshacer acciones fácilmente.
5. Obtener información de retroalimentación.
6. Atajos a usuarios expertos.
7. Que el usuario sintiera la sensación de control.

Una de las premisas fundamentales de la aplicación es la ventaja que proporciona las interfaces Web sobre las interfaces de comandos. Ya que las interfaces Web:

- Proporcionan un ambiente amigable.
- Conducen a un aprendizaje mas natural
- Establecen un "sentimiento" (sobre todo en la uniformidad del ambiente) al usuario que enriquece su experiencia en el uso de la aplicación.

Además de estos principios, se tuvieron en cuenta las siguientes características:

- Utilizar una misma tipografía, forma y estilo en todas las páginas
- La facilidad del usuario de poder navegar desde cualquier punto a otro dentro de la aplicación.
- Se tuvo presente siempre el ancho de banda y por ello se utilizaron formato de imágenes de compresión favorables (gif y jpg)
- La simplicidad y consistencia, favoreciendo la usabilidad de la aplicación.
- Navegación simple en todas las páginas de la aplicación, de forma tal que siempre sea accesible por el usuario.
- Estabilidad y uniformidad del diseño, para así poder ubicar al usuario dentro del mismo y hacerlo sentir parte de él.

Para cumplir con todas las premisas anteriores se utilizaron tres plantillas: una para el computador, otra para el administrador y por ultimo para el usuario, estas se muestran en todas las páginas y varían de acuerdo a las opciones que le están permitidas a los actores, es decir, su diferencia radica en el menú contextual, logrando así un diseño homogéneo en la aplicación.

Para las páginas de salidas de datos se utilizó el mismo formato, una tabla que tiene la funcionalidad como encabezamiento y a continuación todos los datos, sin ninguna posibilidad de cambiarlos, tan solo de verlos. De la misma forma ocurre con las entradas de datos, solo que los campos si pueden ser cambiados; se mantuvo siempre la política de ahorrarle al máximo el trabajo al usuario, es decir, que tuviera que introducir la menor cantidad de información.

2.11.2 – Tratamiento de errores

En la aplicación existe una clase dedicada especialmente al tratamiento de los errores. Todas las páginas acceden a ella y muestran un mensaje pertinente en cada caso de error, garantizando en todo momento la estabilidad y funcionamiento. Esta clase esta preparada para recoger el número del error y presentar la pantalla con el error que le corresponde a ese código.

Se controlan además las páginas que muestran información que viene dada por algún link de la URL, en caso de que no este disponible se transfiere el control a pantallas anteriores de donde se escogió la opción.
Los errores en los datos para insertar, modificar, visualizar o eliminar se procesarán antes de enviar la página para evitar la carga de esta en vano. En este caso se validara y se presentara un mensaje de error.

2.11.3 – Concepción General de la ayuda

La ayuda constituye una parte importante dentro del mundo de las aplicaciones Web, esta no deben ser muy detallista o extensa, sino explicaciones y aclaraciones sencillas, además de las operaciones que puede realizar el usuario sobre el sistema.
En este trabajo se realizó una ayuda que facilita una mejor comprensión de las acciones que el usuario puede desarrollar.

2.12 – Conclusiones

En este capítulo se han dado los primeros pasos para la ejecución del proyecto según el análisis y diseño de los flujos de trabajo; lo cual propició definir los principales

requerimientos funcionales del sistema, y a la vez entender con claridad lo que realmente necesita el cliente, para que se logre un amplio dominio de la solución que se propone. Para el desarrollo del sistema se presentaron los diagramas de clases del diseño dividido por paquetes, utilizando las extensiones de UML para moderar los elementos relativos a las aplicaciones Web. Teniendo en cuenta los niveles de detalle y abstracción apropiados para diseñadores, desarrolladores y arquitectos de una aplicación Web. Se construyó el diagrama de clases persistentes de la aplicación con el objetivo de obtener el modelo lógico y físico de la base de datos y se definieron todos los elementos que describen el modelo de implementación. Además se muestra el diagrama de despliegue y los de componentes.

Capitulo 3 – Estudio de Factibilidad

Introducción

Hay en día el Software es el elemento más caro de la mayoría de los sistemas informáticos, un gran error en la estimaci6n del costa puede ser lo que marque la diferencia entre beneficios y pérdidas.

Existen diferentes modelos para calcular la estimación del costo de un sistema, actualmente el más difundido es el Modelo Constructivo de Costes (COCOMO). Es uno de los modelos más documentados en la actualidad y es muy fácil de utilizar, por todo lo anterior, lo utilizaremos en nuestro trabajo.

Fue desarrollado por B. W. Boehm a finales de los 70 y comienzos de los 80, pertenece a la categoría de modelos de estimación basados en estimaciones matemáticas. Está orientado a la magnitud del producto final, midiendo el tamaño del proyecto en líneas de código principalmente. **[32]**

Entre sus ventajas se encuentran:

- La utilizaci6n de unidades físicas, tales como las líneas de código para medir la magnitud del producto.
- La fácil realización, ya que las unidades de medida son tangibles
- La flexibilidad y la utilizaci6n en cualquier tipo de proyecto.
- Los resultados obtenidos son proporcionales a las tareas técnicas.

3.1 Planificación por puntos de función.

Entradas Externas	Ficheros	Elementos	Clasificación
Insertar datos de una reunión.	6	12	Alta
Modificar datos de una reunión.	6	12	Alta
Eliminar datos de una reunión.	6	12	Alta
Insertar datos del plan de trabajo.	1	9	Baja

Modificar datos del plan de trabajo.	1	9	Baja
Eliminar datos del plan de trabajo.	1	9	Baja
Insertar datos de cotización.	1	4	Baja
Modificar datos de cotización.	1	4	Baja
Eliminar datos de cotización.	1	4	Baja
Insertar datos de los comités de base.	1	6	Baja
Modificar datos de los comités de base.	1	6	Baja
Eliminar datos de los comités de base.	1	6	Baja
Insertar datos de un militante estudiante.	2	12	Media
Modificar datos de un militante estudiante.	2	12	Media
Insertar datos de un militante trabajador.	2	10	Media
Modificar datos de un militante trabajador.	2	10	Media
Insertar datos de no militante.	1	13	Baja
Modificar datos de no militante.	1	13	Baja
Eliminar datos de no militante.	1	13	Baja
Dar baja a un militante.	3	3	Media
Insertar militantes sancionados.	3	4	Media
Modificar militantes sancionados.	3	4	Media
Eliminar datos de militantes sancionados.	3	4	Media

Insertar datos de usuarios.	1	5	Baja
Modificar datos de usuarios.	1	5	Baja
Eliminar datos de usuarios.	1	5	Baja
Insertar datos de facultades.	1	1	Baja
Modificar datos de facultades.	1	1	Baja
Eliminar datos de facultades.	1	1	Baja
Insertar datos de especialidad.	2	2	Baja
Modificar datos de especialidad.	2	2	Baja
Eliminar datos de especialidad.	2	2	Baja
Insertar datos de áreas.	1	1	Baja
Modificar datos de áreas.	1	1	Baja
Eliminar datos de áreas.	1	1	Baja
Insertar datos de departamentos.	2	2	Baja
Modificar datos de departamentos.	2	2	Baja
Eliminar datos de departamentos.	2	2	Baja
Cambiar contraseña	1	2	Baja

Tabla 3.1 Entradas Externas.

Salidas Externas	Ficheros	Elementos de datos.	Clasificación
Consultar ausentes.	2	3	Baja
Consultar militantes no físicos.	3	6	Alta
Consultar militantes no incorporados.	3	6	Alta

Consultar actas fuera de fecha.	2	3	Baja
Consultar militantes sancionados.	5	3	Media
Consultar reuniones no visitadas.	3	5	Baja
Consultar C/B no cotizados.	2	4	Baja
Visualizar pago de cotización.	2	4	Baja
Visualizar datos de usuario.	1	3	Baja
Visualizar datos de facultad.	1	2	Baja
Visualizar datos de especialidad.	2	2	Baja
Visualizar datos de áreas.	1	2	Baja
Visualizar datos de departamentos.	2	3	Baja

Tabla 3.2 Salidas Externas.

Peticiones	Ficheros	Elementos de datos.	Clasificación
Buscar militantes por comité de base.	5	12	Alta
Buscar datos de reunión.	6	12	Alta
Buscar datos de militante.	2	10	Media
Buscar datos de no militante.	1	13	Baja
Buscar datos del plan de trabajo.	1	9	Baja
Buscar datos de sancionados	3	4	Media
Consultar reuniones por fecha.	2	4	Baja

Tabla 3.3 Peticiones

Ficheros Lógicos Internos	Records	Elementos de datos.	Clasificación
Usuario	1	5	Baja
No militante	1	14	Baja
Comité de base	1	6	Baja
Plan de trabajo	1	11	Baja
Cotizados	1	4	Baja
Sancionados	1	4	Baja
No físicos	1	2	Baja
No incorporados	1	1	Baja
Reunión	1	11	Baja
Ausentes	1	5	Baja
Facultades	1	2	Baja
Especialidades	1	3	Baja
Áreas	1	2	Baja
Departamentos	1	3	Baja
Altas_Bajas	1	4	Baja
Estudiante	1	4	Baja
Militante	1	10	Baja
Municipio	1	3	Baja
Provincia	1	2	Baja
Reunión extraordinaria	1	3	Baja
Reunión ordinaria	1	2	Baja

Sanciones	1	2	Baja
Trabajador	1	2	Baja

Tabla 3.4 Ficheros Lógicos Internos.

Elementos	Bajos		Medios		Altos		Subtotal de ptos de función
	No	X Peso	No	X Peso	No	X Peso	
Ficheros lógicos internos.	23	7	0	10	0	15	161
Ficheros de interfases externas.	0	5	0	7	0	10	0
Entradas externas.	28	3	8	4	3	6	134
Salidas externas.	10	4	1	5	2	7	63
Peticiones	3	3	2	4	2	6	29
TOTAL							**387**

Tabla 3.5 Puntos de Función sin ajustar.

Características	Valor	
Puntos de función desajustados	387	
Lenguajes.	PHP	MySql
Instrucciones fuentes por puntos de función	44	40
Por ciento de la aplicación en cuanto a requerimientos funcionales.	85 %	15 %
Instrucciones fuentes.	14473.8	2322
Total de Instrucciones fuentes	**16795.8**	

Tabla 3.6 Instrucciones fuentes.

Miles de instrucciones fuentes **(MF)= 16.7958**

3.2 – Determinación de los costos

Cálculo de:	Valor	Justificación
RCPX	1.00	Las exigencias de confiabilidad, documentación y volumen de datos son moderadas.
RUSE	1.00	Se implementa código reutilizable para el aprovechamiento de este en toda la aplicación.
PDIF	0.87	No tiene restricciones en cuanto al tiempo de ejecución y no tiene limitación de memoria. La plataforma de aplicación tiene gran estabilidad.
PERS	0.50	Analista y programadora con eficiencia media.
PREX	0.87	Se tiene experiencia en la plataforma y herramientas de desarrollo. Con una experiencia de aproximadamente un años.
FCIL	1.10	Se utilizan herramientas de programación como: Macromedia Dreamweaver 2004, así como la herramienta CASE Rational Rose para la documentación, empleando como notación UML,
SCED	1.00	Se requiere terminar el proyecto en tiempo estimado.
PREC	3.72	Se posee una comprensión considerable de los objetivos del producto, existen experiencias en la realización de software de este tipo. (Nominal)
FLEX	3.04	Algo de relajación en cuanto a la flexibilidad del desarrollo.
TEAM	1.10	El equipo que va a desarrollar el software es altamente cooperativo.
RESL	2.87	Arquitectura sólida.
PMAT	6.24	La madurez del proceso software es bajo.

Tabla 3.7 Costos: Factores de escalas

Multiplicador de esfuerzos

$EM = \prod_{i=1} Emi = RCPX*RUSE*PDIF*PERS*PREX*FCIL*SCED$

EM = 0.416295

Factores de escala

$SF = \sum SFi = PREC + FLEX + RESL + TEAM + PMAT$

SF = 16.97

Valores de los coeficientes

A, B, C, D coeficientes con valores: **A=** 2,94, **B=** 0,91, **C=** 3,67, **D=** 0,24

E = B + 0,01 * SF

E = 0.91+0.01*16.97

E = 1.0797

F = D + 0,2 * (E - B)

F = 0.24+0.2*(1.0797-0.91)

F = 0.27394

Esfuerzo

$PM = A * (MF)^{E} * EM$

$PM = 2.94* (16.7958)^{1.0797} * 0.416295$

PM = 25.739 ≈ 25.74

Cálculo del tiempo de desarrollo

$TDEV = C * PM^{F}$

$TDEV = 3.67*(25.74)^{0.27394}$

TDEV = 8.935 ≈ **9 meses**

Cálculo de la cantidad de hombres

CH = PM / TDEV

CH = 25.74/ 9

CH = 2.86 ≈ **3 hombres**

Ajustando para 1 hombre el proyecto:

CH* = 1 hombre.

TDEV = PM/CH* = 25.74

Costo

CHM = 1 * Salario Promedio

CHM = 1*225

CHM = $225/ Mes

Costo = CHM * PM

Costo = $225*25.74

Costo = $5791.5

Los costos en los que se incurriría de desarrollarse el sistema serían:

Cálculo de:	**Valor**
Esfuerzo(PM)	25.74
Tiempo de desarrollo	9 meses
Cantidad de hombres	3 hombres
Costo	$5791.5
Salario medio	$225
RCPX	1.00
RUSE	1.00
PDIF	0.87
PREX	0.87
FCIL	1.10
SCED	1.00

Tabla 3.8 Costos totales

3.4 Beneficios tangibles e inteligentes.

El sistema permite un mejor control de los procesos que tienen lugar en la UJC, además lo hace de una forma flexible, dinámica y de interfaz agradable. No solo facilita todos los procesos relacionados con la adquisición y uso de datos, sino que permite darle un mejor uso a las tecnologías de la información y ayudar en la toma de decisiones.

Ahora además de contar con una aplicación Web que permite registrar, actualizar y conocer de una forma precisa y en el menor tiempo posible datos de interés de los miembros de la organización, los datos se encuentran mejor organizados y controlados en un solo sistema confiable y seguro.

Los beneficios tangibles del sistema no están orientados a cifras en cuanto a materiales de oficina se refiere, sine más en concreto a las variables que definen el funcionamiento de la Unión de Jóvenes Comunistas y al aprovechamiento óptimo del tiempo, recurso imprescindible en los momentos actuales que esta viviendo nuestro país.

3.5 Análisis de costos y beneficios.

Este sistema, como producto del presente trabajo de diploma no implica costo alguno para la Unión de Jóvenes Comunistas, todo lo contrario, es un honor para nosotros poder aportar un granito de arena más a esta lucha que libra nuestro pueblo. La construcción de un sistema que sea capaz de gestionar la información de los procesos era evidente y solución única, ya que así se puede ayudar en la toma de decisiones.

A pesar de la importancia que confiere la aplicación no reporta grandes gastos. es decir si tomamos en cuenta que la cantidad de tiempo que ahorra puede ser empleado en otras tareas no menos importantes y de vital importancia, vale la pena el costo del proyecto El tiempo y la agilidad con que sepamos actuar es nuestra principal arma en estos tiempos.

3.6 Conclusiones.

No caben dudas de que una óptima planificación del proyecto contribuye a un desarrollo organizado y como consecuencia a una mejor calidad del producto obtenido. No solo se comprendió el ámbito del trabajo a realizar, los recursos que se necesitan y las tareas específicas a ejecutar, si no que además se tuvo en cuenta el costo y la factibilidad ya que esto puede influir grandemente en la aceptación y desarrollo del sistema deseado.

Además de realizar los cálculos manualmente, se utilizo el Demo Cocomo 11, que es un sistema para el cálculo de todas las variables planteadas por Boehm, y así se pudieron corroborar todos los datos obtenidos. **[33]**

Conclusiones

Para cumplir con el objetivo propuesto se realizó un estudio profundo de cómo funciona la organización de la UJC en la Universidad. Se obtiene la propuesta de un Sitio Web que informatizará el procesamiento manual de la información que actualmente resulta complejo, lo que constituyó el punto de partida para la realización del sistema, aquí se definieron los procesos más importantes a automatizar.

Para el diseño e implementación se estudió las tendencias, tecnologías y metodologías principales con el fin de tener una mayor confiabilidad en la información obtenida y facilitarle el trabajo al Secretario General, así como mejorar estas condiciones de trabajo. La herramienta desarrollada permite un menor tiempo de respuesta ante una solicitud de información y brinda facilidades al usuario para trabajar con la información de forma dinámica. Lográndose una seguridad y protección de los datos consecuente con el nivel de seguridad requerido.

Con el estudio y realización del Sitio Web se cumple con el objetivo propuesto de desarrollar un software que facilite la gestión de la actualización de los comités de base de la universidad y garantice el acceso a la información con la calidad, confiabilidad y rapidez requerida para el departamento u otros usuarios.

Recomendaciones

Después de realizar un análisis del presente trabajo, pueden ser planteadas las siguientes sugerencias:

Poner a prueba el sistema durante un período de tiempo significativo, para comprobar su desempeño y luego llevarlo como propuesta a la provincia ya que existe un amplio campo de desarrollo del Sitio Web en el futuro.

Continuar la implementación y desarrollo del sistema, partiendo de lo realizado hasta ahora.

Enriquecer el Sitio con nuevos enlaces, programas o servicios; que posibiliten el desarrollo de la UJC en nuestra universidad

Bibliografía

- Booch, G. Object-Oriented Analysis and Design with Applications. EBSCO, (10/02/07).
- Bosman, David. Crea tu sitio Web: Aprende Html y CSS. –PC cuadernos-Técnicos 2006. 200p
- García Alcázar, Enrique. Análisis de Requerimientos y Trazabilidad en el Modelo de Objetos. http://www.ati.es/gt/LATIGOO/OOp96/Ponen4/atio6p04.html, (2/05/06)
- Goodman, D. JavaScript Bible / D. Goodman. –[s.l]: [s.n], 2001. –34p.
- Jacobson, I. El Proceso Unificado de Desarrollo de software / Ivar Jacobson; G. Booch; J. Rumbaugh. --Addison-Wesley: [s.n], 2000. –356p.
- Leon, Mario. Diccionario de informática, telecomunicaciones y ciencias afines: ingles-español spanish-english. –ediciones Diaz de Santos, s.a., 2004.
- Lancker, luc van. Css 2 y css 2.1: hojas de estilo para enriquecer el código html. – ediciones eni 2007.
- Luke Welling, Thomson, Laura. PHP and MySQL Web Development / Laura Thomson, Luke Welling. –Third Edition: Sams, 2004. –984p
- Nieto, Victor. Fundamentos Físicos y Tecnológicos de la Informática. –Prentice Hall, 2006.
- Perez, Cesar. Oracle 9i: Servidor de Aplicaciones, Red y Programación. – Ra-Ma 2003.
- Pressman, R. Software Engineering. A Practitioner's Approach.--(E.U). McGraw – Hill, 1999.--[S.P].
- Stevens, Perdita. Utilizacion del UML en Ingenieria de Software. –Addison-Wesley Iberoamericana España, s.a., 2007
- Santana, Pedro. "Implementando servicios Web con PHP". Tomado De: http://www.pecesama.net/php/ws.php.

Referencias bibliográficas

[1] -Estatutos UJC. http://intranet.opr.edu.cu/ujc/pages/estatutos.htm (08/01/07)

[2] -Partido Comunista de Cuba. Unión de Jóvenes Comunistas. www.pcc.cu/pccweb/organizaciones/ujc.php (08/01/07)

[3] -Rodríguez ,1. La UJC se escribe con mayúscula. Archivo Central de la UJC Nacional.

[4] –Alejo, Oscar J. *Portal de Psicología de la Universidad de Cienfuegos.* Trabajo de Diploma para optar por el título de Ingeniería en Informática. Universidad de Cienfuegos "Carlos Rafael Rodríguez".Cienfuegos junio 2006.

[5] -Alcaz,y. *Sistema Automatizado para el control de la militancia UJC – CUJAE*. Trabajo de diploma para optar por el título de Ingeniero Informático, Instituto Superior Politécnico José Antonio Echeverría, Ciudad de la Habana, junio 2000.

[6] -Pérez, Y Jua, L. *Sistema para la gestión de Información Integral de la FEU (FEU Gest)*. Trabajo de diploma para optar por el título de Ingeniero Informático, Instituto Superior Politécnico José Antonio Echeverría. Ciudad de la Habana, junio 2004.

[7] -PERL—es.wikipedia.org/wiki/perl

[8] -Mula, f. Las siglas de ASP. http://www.talleresdelweb.com/index.php?tdw=tllarticulos&id =1&acc=65. (15/02/07)

[9]-García, M. Curso Comercio Electrónico. Segunda edición. http://geneura.org.es/~maribel/php/.(15/02/07)

[10]-JSP (Java Server Pages). http://html.rincondelrago.com./isp.html. (15/02/07)

[11] -Introducción a Java Script. http://www.depersonales.com/ecuador/Machala/jauascripts/jauaescript (15/02/07)

[12] -Introducción a Visual Basic Script. http://desarrolloweb/articulos/161.php?manual=114 (15/02/07)

[13] -Alvarez R.Lenguajes de lado servidor o cliente . http:239/.php?manual=7

[14] -Curso de Diseño con Dreamweaver MX 2004 - Cursos de Dreamweaver On-Line http://www.ciberaula.com/curso/dreamweaver/ (16/02/07)

[15] -Ferrá Grau Xavier *"Desarrollo orientado a objetos con UML"* Universidad Politécnica de Madrid, 2004 URL: http://www.clikear.com/manuales/uml/introduccion.asp, (16/02/07)

[16] - Letelier Torres Patricio"Desarrollo de Software Orientado a Objeto usando UML", Departamento Sistemas Informáticos y Computación, Universidad Politécnica de Valencia, España. URL: http://www.creangel.com/uml/intro.php, (20/02/07)

[17] Jacobson, Ivar; Booch, Grady y Rumbaugh, James. El Proceso Unificado de Desarrollo de software. Editorial Felix Varela, La Habana,2004. TI, p. 115-129

[18] "Lo Nuevo de Rational Rose 2000" Racional Corporation, 2000. http://www.abists.com.mf/Fabs/Rational/notasTK (21/02/07)

[19] - API. Tomado de http:es.wikipedia.org/wiki/API.

[20] - ¿Por qué elegir PHP? www.igelectronica.com/desarrollophp/modules.php?name=News&file=article&sid=53 (21/02/07)

[21] - Servidor http Apache. http://es.wikipedia.org/wiki/servidorHTTPApache (21/02/07)

[22] - Cárdenas L. Introducción a las base de datos. http://www.rinconprogametropoliglobal.com/CursosProg/BDatos/IntroBD/index.php?cap=1 (24/02/07)

[23] - My SQL. http://software_shop.com/Products/MySQL/mysql.html (24/02/07)

[24] - Aguilar V y Suan P.My SQL vs postgre SQL. http://www.mmlabx.va.es/mysql-postgre.html (24/02/07)

[25] - ¿Qué es Oracle? http://www.desarrolloweb.com/articulos/840.php?manual=15 (24/02/07)

[26] - Instalación y Configuración de Microsoft SQL Server 2000. http://www.usuario.lycos.es/outsusgbd/UD5.htm (26/02/07)

[27] – Álvarez, Hugandy. *Desarrollo de una Intranet para un Departamento Docente de un Centro de Educación Superior (CES)".* Trabajo de Diploma para optar por el título de Ingeniería en Informática. Universidad de Cienfuegos "Carlos Rafael Rodríguez".Cienfuegos junio 2006.

[28] - Jacobson, I.; Booch, G. y Rumbaugh, J.; "El Proceso Unificado de Desarrollo de software". Addison-Wesley,2000, 121 p

[29] - Idem, 125p.

[30] - Dorado C. El diseño de interfaz y la navegación. http://www.xtec.es/~odorado/cdora1/esp/disseny.htm (10/03/07)

[31] - Jimenez. Ing F. Tutorial ”Diseño de una Interfaz Gráfica”. Universidad Autónoma de Guadalajara.
http://www.universia.net/Catalogaxx/php/ir.asp?idURL=42071&iDP=100110&iDP=ES&iDi=1 (10/03/07)

[32] - Peralta M. Estimacion del esfuerzo basada en casos de usos. Reportes Tecnicos Ingenieria W Software.155N 1667_5002.Volumen 6, No 1, pag 1-16.2004.
http://www.itba.edu.ar/capis/webcapis/planma_esp.html (30/04/07)

[33] - COCOMO 11: Una Familia de Modelos de Estimacion. Universidad Simosn Bolivar, Ingenieria del Software 3, Enero-Marzo 2001.

Glosario de Términos

- API (Application Programming Interface)
- ASP (Active Server Pages)
- APL (General Public License)
- Case (Computer-Aide Software Engineering)
- COCOMO (Constructive Cost Model)
- PERL (Practical Extracting and Reporting Lenguage)
- RUP (Rational Unifield Process)
- UML (Unified Modeling Lenguage)

Anexos

Anexo 1 –Prototipos

A continuación se muestran los prototipos de los principales casos de uso del sistema.

Prototipos del Módulo Actualizar Datos

Anexo 1.1 -Actualizar estudiante no militante.

Anexo 1.2 -Actualizar datos del plan de trabajo.

Anexo 1.3 -Actualizar cotización.

Anexo 1.4 -Actualizar sancionados.

Anexo 1.5 -Actualizar datos de una reunión.

Anexo 1.6 -Actualizar militante.

Anexo 1.7 –Dar Baja a un militante.

Prototipos del Módulo Reportes

Anexo 1.8 –Consultar ausentes

Anexo 1.9 –Consultar militantes no físicos

Anexo 1.10 –Consultar militantes no incorporados

Anexo 1.11 –Consultar Listado por comité de base

Anexo 1.12 – Consultar Actas fuera de Fecha

Anexo 1.13 – Consultar militantes sancionados.

Anexo 1.14 – Consultar Reuniones por fecha

Anexo 1.15 – Consultar los comités de base no cotizados.

Prototipos del Módulo Seguridad

Anexo 1.16 –Actualizar usuario

Anexo 1.17 –Autenticar usuario

Anexo 1.18 –Cambiar contraseña

Anexo 1.19 –Actualizar datos de lo comités de base

Anexo 1.20 -Actualizar datos de facultad

Anexo 1.21 -Actualizar datos de especialidad

Anexo 1.22 -Actualizar datos de un Área.

Anexo 1.23 –Actualizar datos de un departamento.

Anexo 2 – Diagrama de clases del diseño Web.

Módulo Actualizar Datos

Anexo 2.1 -Actualizar estudiante no militante.

Anexo 2.2 -Actualizar datos del plan de trabajo.

Anexo 2.3 -Actualizar cotización.

Anexo 2.4 -Actualizar sancionados.

Anexo 2.5 -Actualizar datos de una reunión.

Anexo 2.6 -Actualizar militante.

Anexo 2.7 –Dar Baja a un militante.

Anexo 3 – Diagrama de clases del diseño Web.

Módulo Reportes

Anexo 3.1 Consultar ausentes

Anexo 3.2 Consultar militantes no físicos

Anexo 3.3 Consultar militantes no incorporados

Anexo 3.4 Consultar Listado por comité de base

Anexo 3.5 Consultar Actas fuera de Fecha

Anexo 3.6 Consultar militantes sancionados.

Anexo 3.7 Consultar Reuniones por fecha

Anexo 3.8 Consultar los comités de base no cotizados.

Anexo 4 – Diagrama de clases del diseño Web.

Módulo Seguridad

Anexo 4.1 Actualizar usuario

Anexo 4.2 Autenticar usuario

Anexo 4.3 Cambiar contraseña

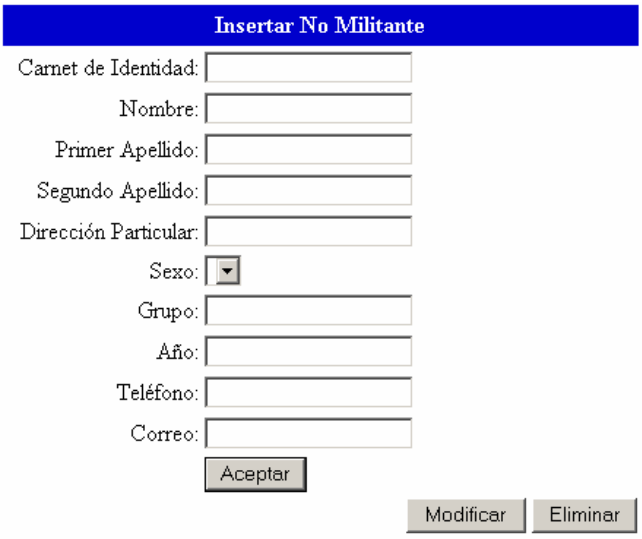
Anexo 4.4 Actualizar datos de lo comités de base

Anexo 4.5 Actualizar datos de facultad

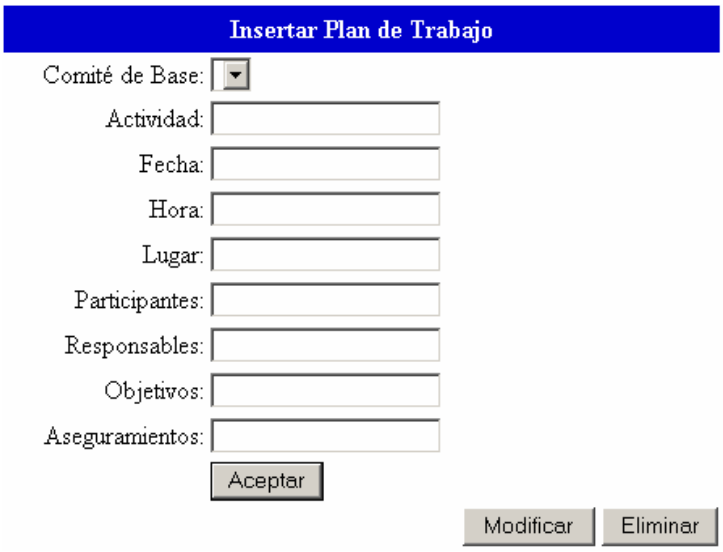
Anexo 4.6 Actualizar datos de especialidad

Anexo 4.7 Actualizar datos de un Área.

Anexo 4.8 Actualizar datos de un departamento.



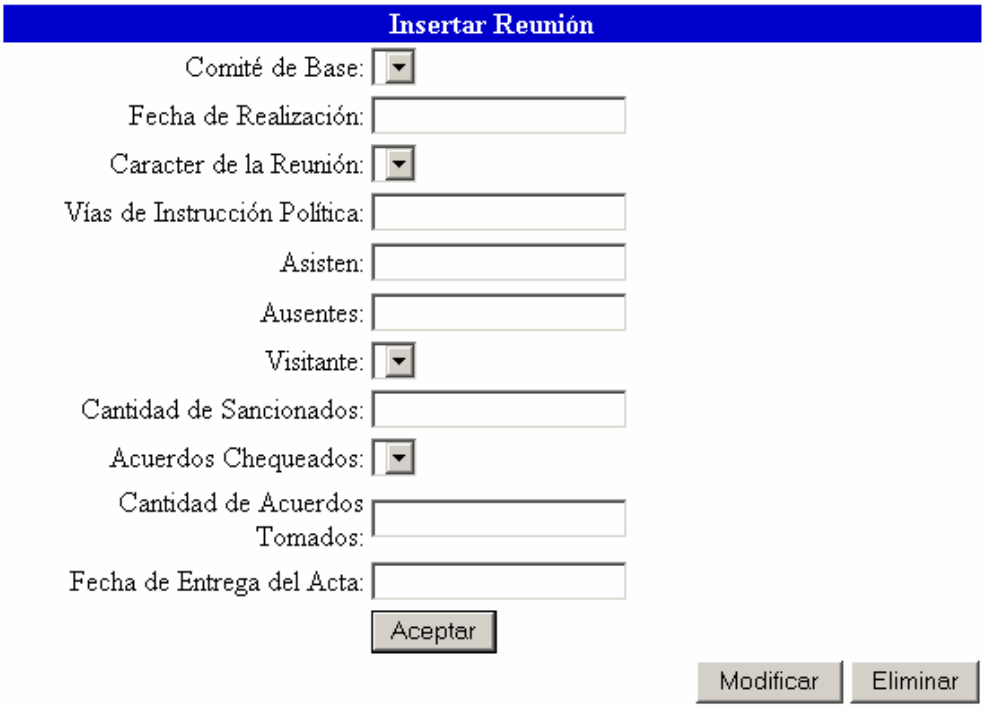
Anexo 1.1 -Actualizar estudiante no militante.



Anexo 1.2 -Actualizar datos del plan de trabajo.

Anexo 1.3 -Actualizar cotización.

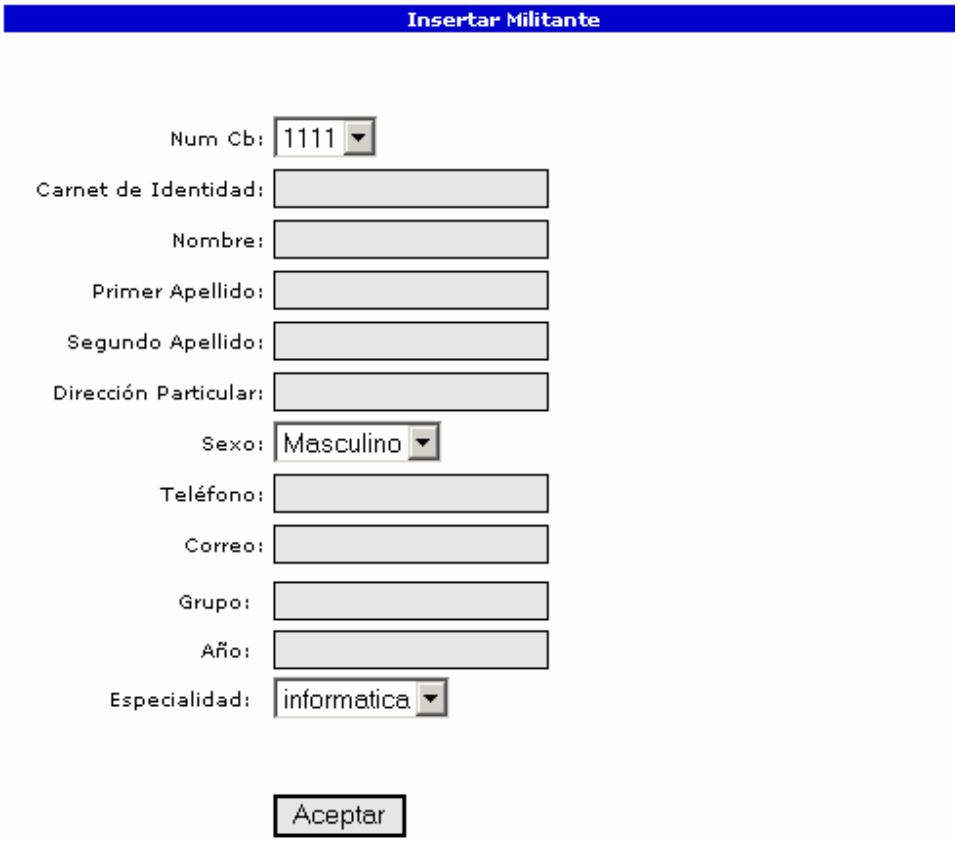
Anexo 1.4 -Actualizar sancionados.



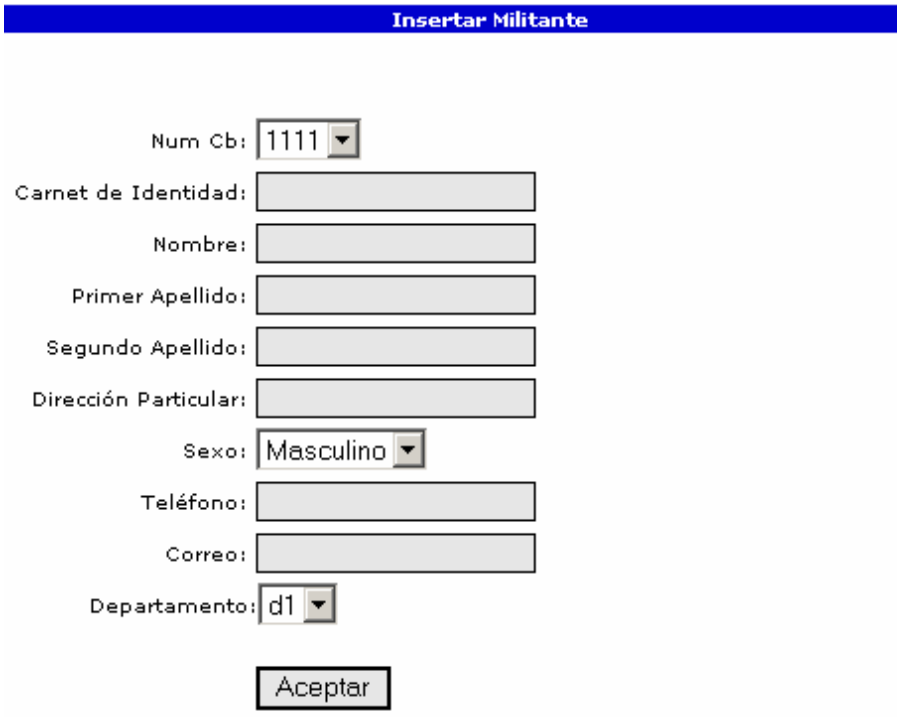
Anexo 1.5 -Actualizar datos de una reunión.

Escoger opción:

Si es Estudiante:

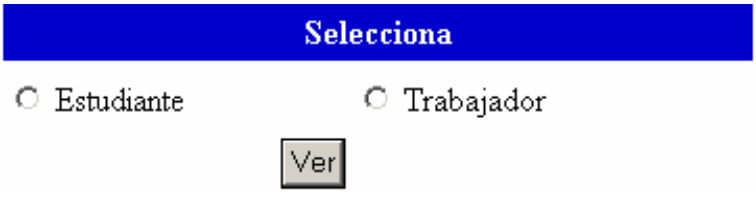


Si es un Trabajador:

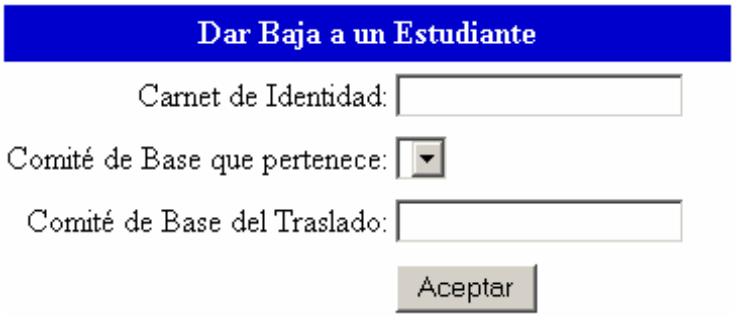


Anexo 1.6 -Actualizar militante.

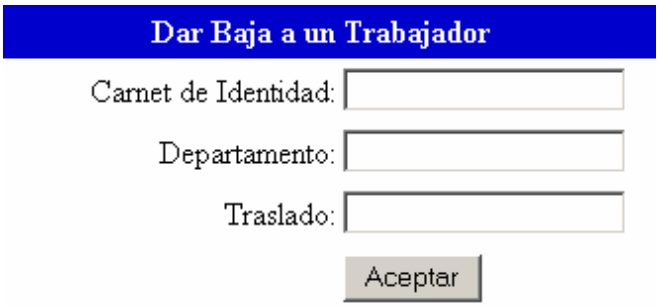
¿A quien le das baja?



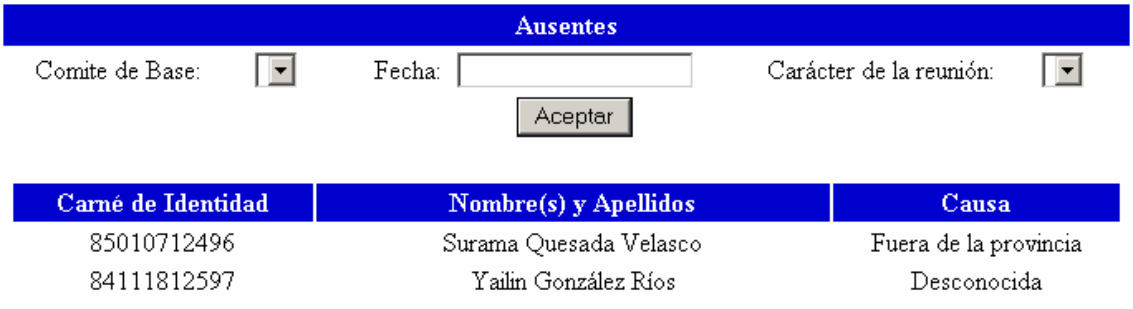
A un estudiante



A un Trabajador:



Anexo 1.7 –Dar Baja a un militante.



Anexo 1.8 –Consultar ausentes

Militantes NO Físicos	
Nombre(s) y Apellidos	**Número del Comité de Base**
Surama Quesada Velasco	1111
Mailen Albiza Albiza	1232

Anexo 1.9 –Consultar militantes no físicos

Militantes NO Incorporados	
Nombre(s) y Apellidos	**Número del Comité de Base**
Surama Quesada Velasco	1111
Mailen Albiza Albiza	1232

Anexo 1.10 –Consultar militantes no incorporados

Listar Militantes	
Número del CB:	
Nombre(s) y Apellidos	**Número del Comité de Base**
Surama Quesada Velasco	1111
Mailen Albiza Albiza	1232

Anexo 1.11 –Consultar Listado por comité de base

Actas Atrasadas	
Fecha Actual:	
Comité de Base	**Carácter de la Reunión**

Anexo 1.12 – Consultar Actas fuera de Fecha

Sancionados del Mes			
Nombres y Apellidos	**Comité de Base**	**Fecha de Inicio**	**Fecha de Culminación**
Surama Quesada Velasco	1111	2007-06-15	2007-09-15

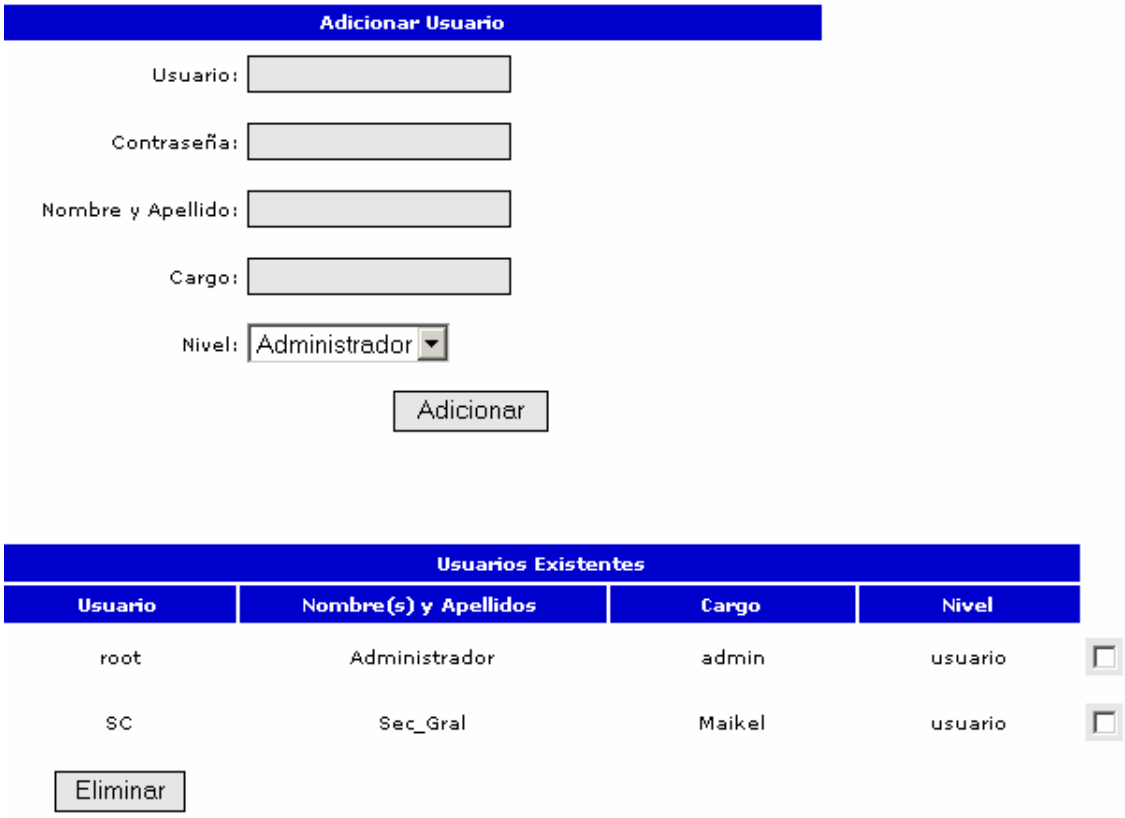
Anexo 1.13 – Consultar militantes sancionados.

Reuniones del Día...	
Fecha:	Mostrar
Número del Comité de Base	**Carácter de la Reunión**
1232	Extraordinaria
1111	Ordinaria

Anexo 1.14 – Consultar Reuniones por fecha

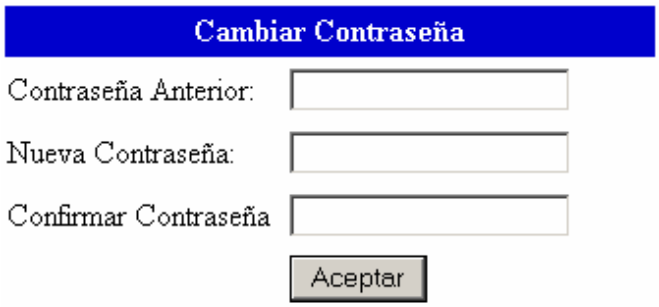
Comités de Base NO Cotizados	
Mes a Cotizar:	
Comité de Base	**Facultad**
1111	Mecánica

Anexo 1.15 – Consultar los comités de base no cotizados.

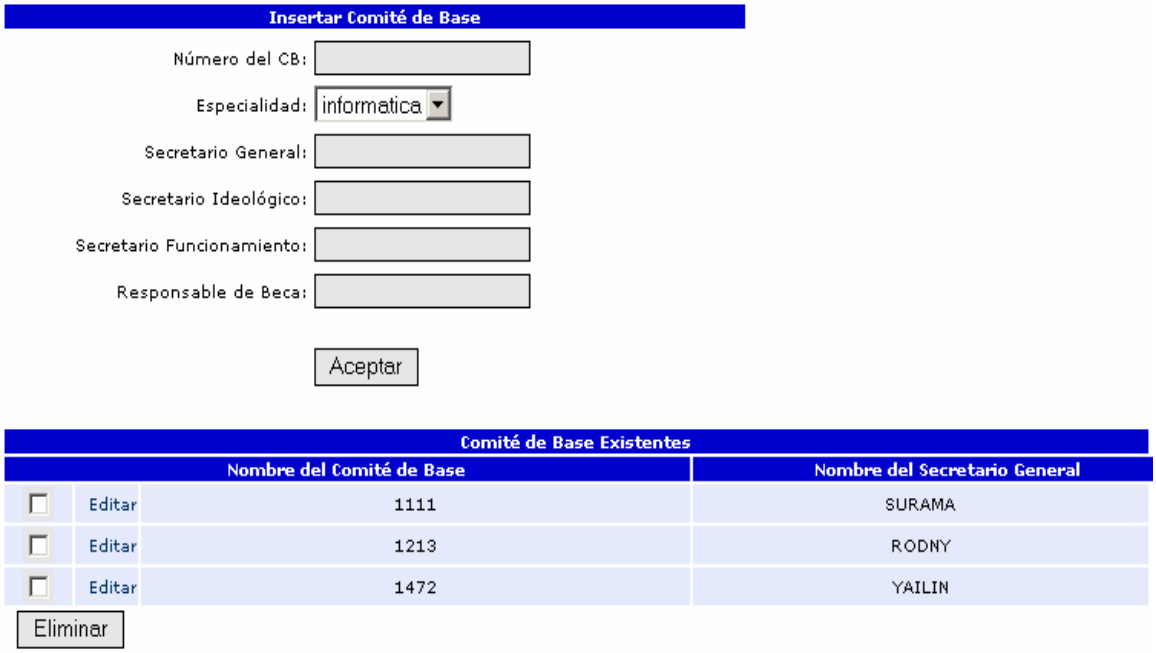


Anexo 1.16 –Actualizar usuario

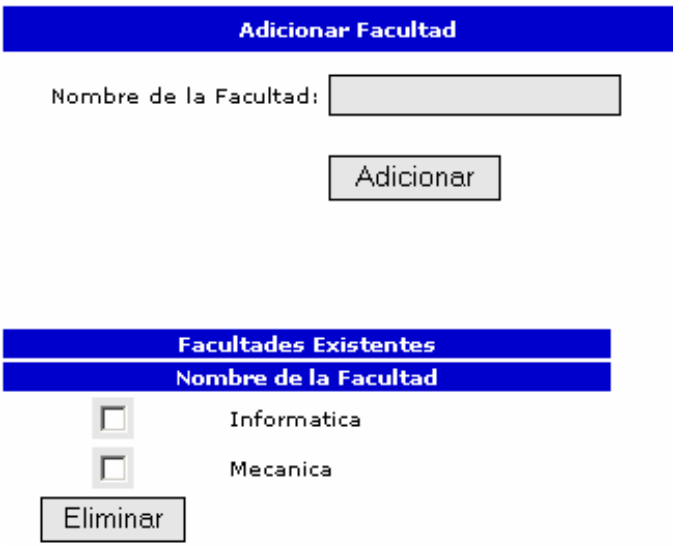
Anexo 1.17 –Autenticar usuario



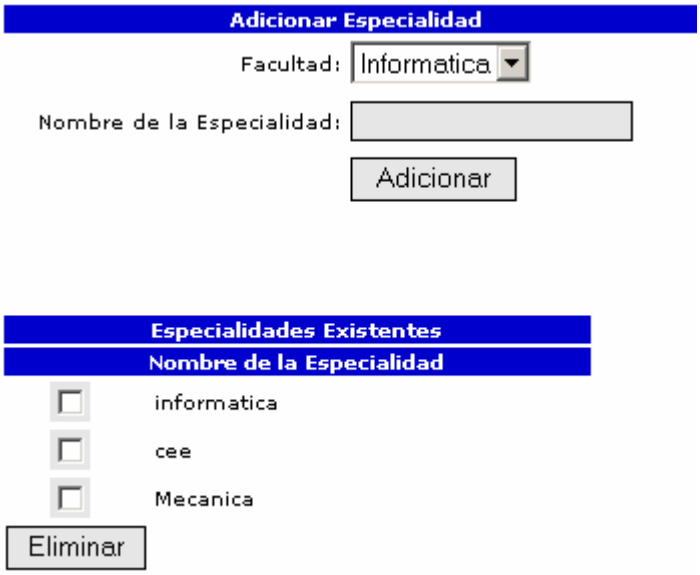
Anexo 1.18 –Cambiar contraseña



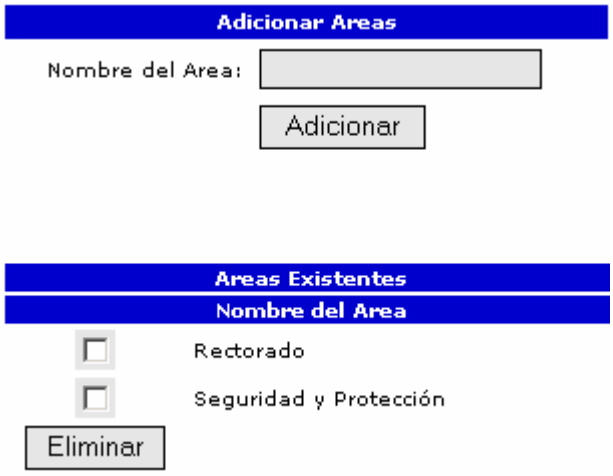
Anexo 1.19 –Actualizar datos de lo comités de base



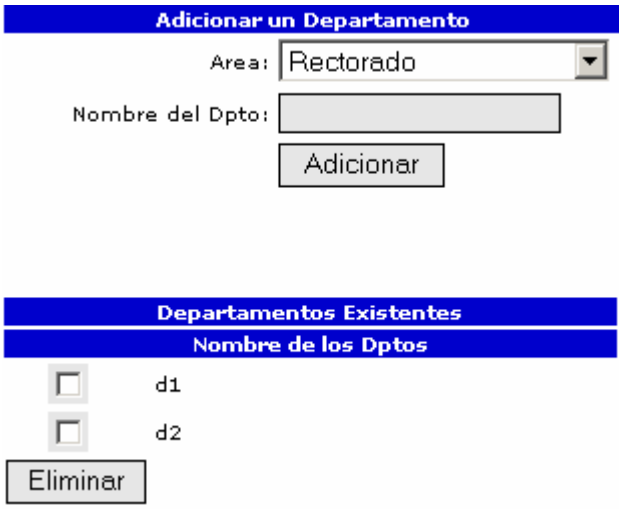
Anexo 1.20 -Actualizar datos de facultad



Anexo 1.21 -Actualizar datos de especialidad



Anexo 1.22 -Actualizar datos de un Área.

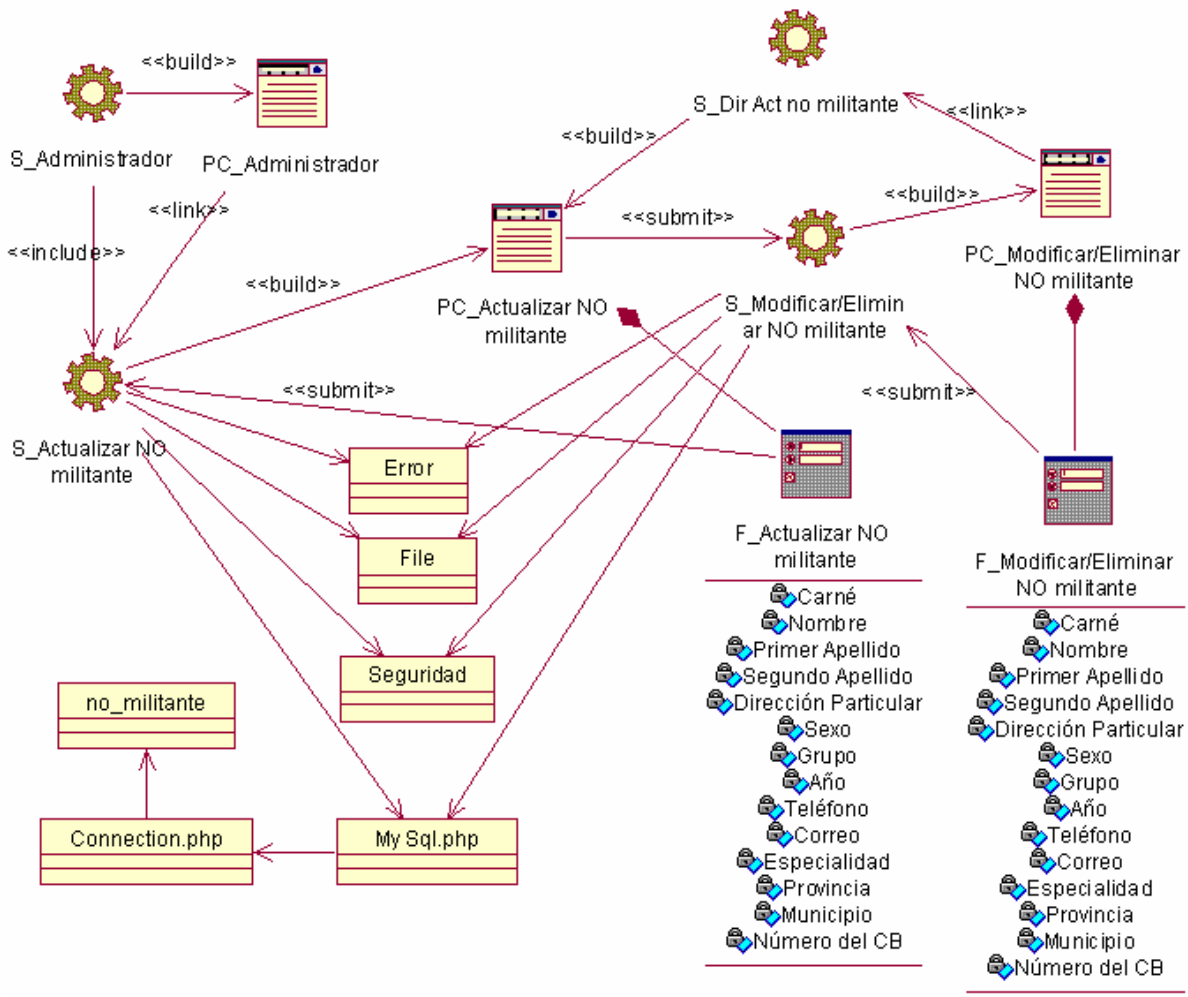


Anexo 1.23 –Actualizar datos de un departamento.

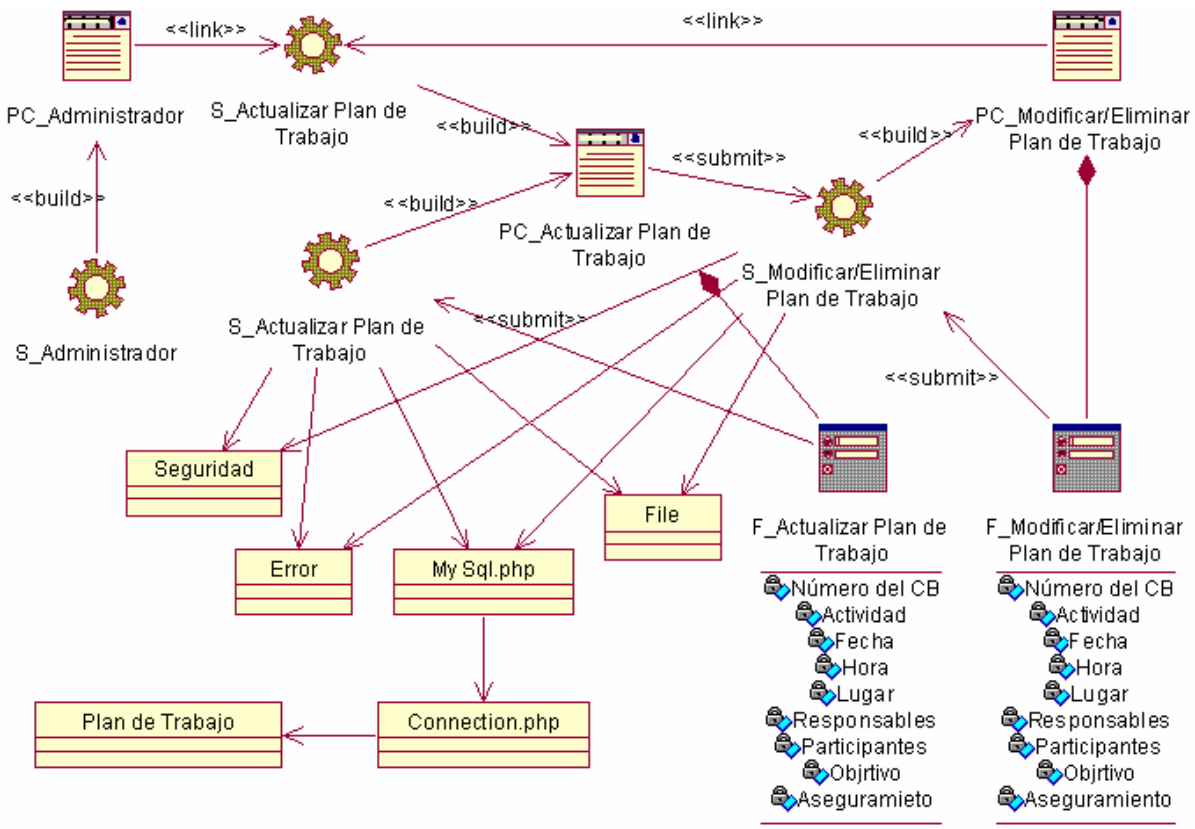
Anexo 2 – Diagrama de clases del diseño Web.

Módulo Actualizar Datos

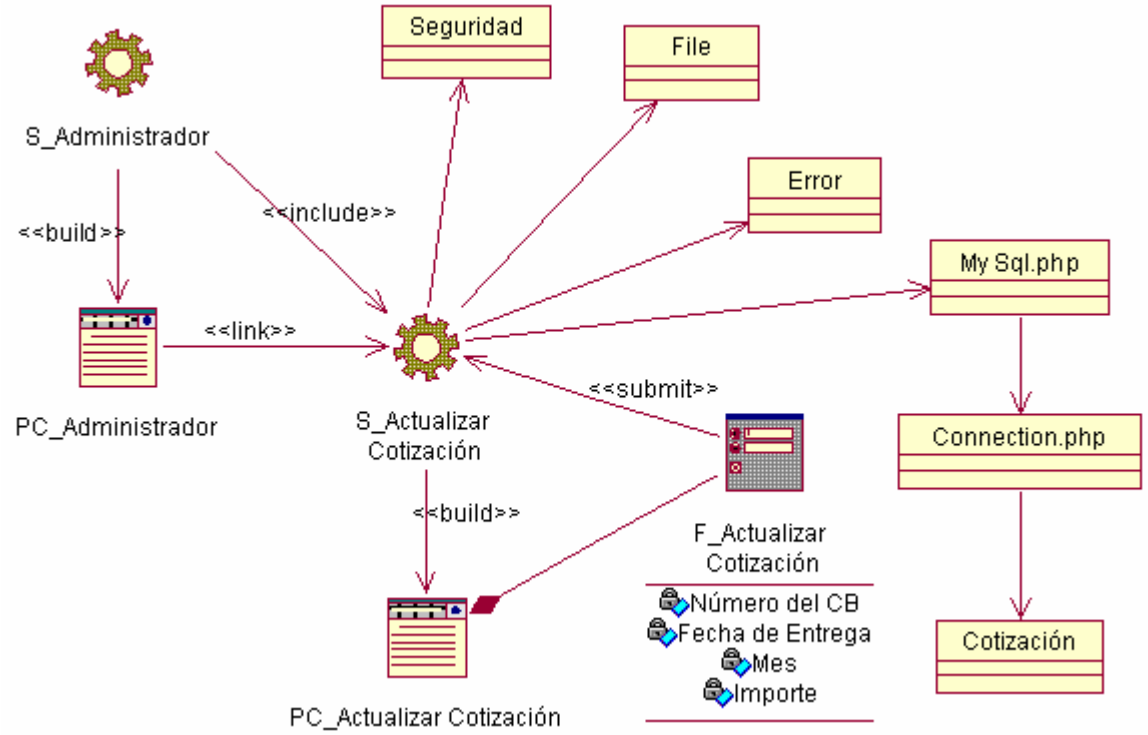
Anexo 2.1 -Actualizar estudiante no militante.



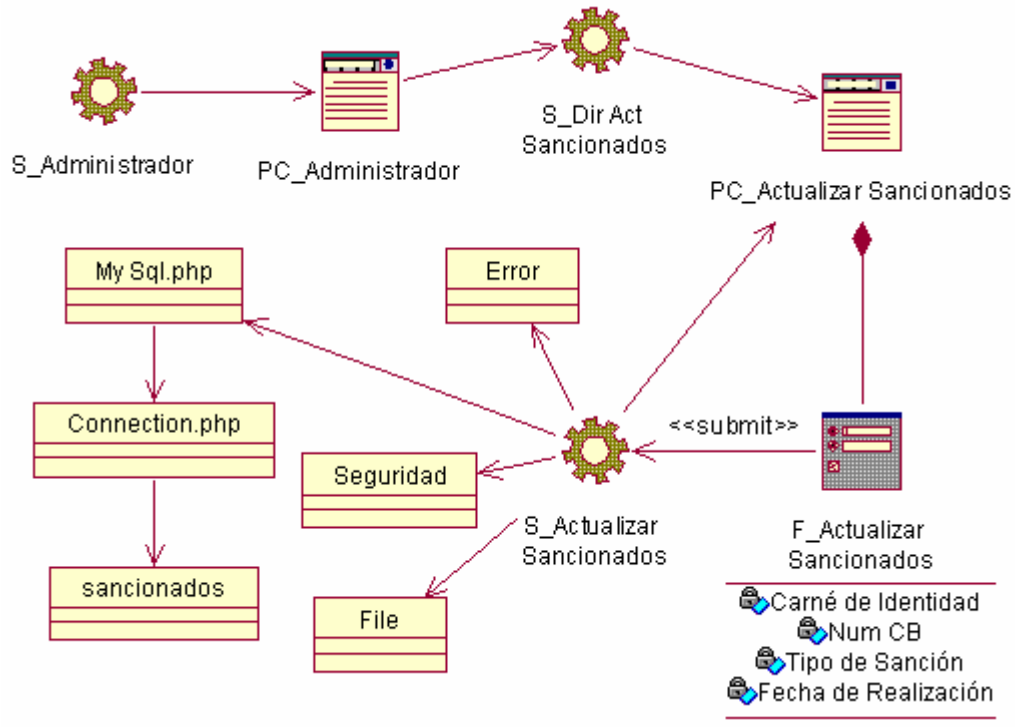
Anexo 2.2 -Actualizar datos del plan de trabajo.



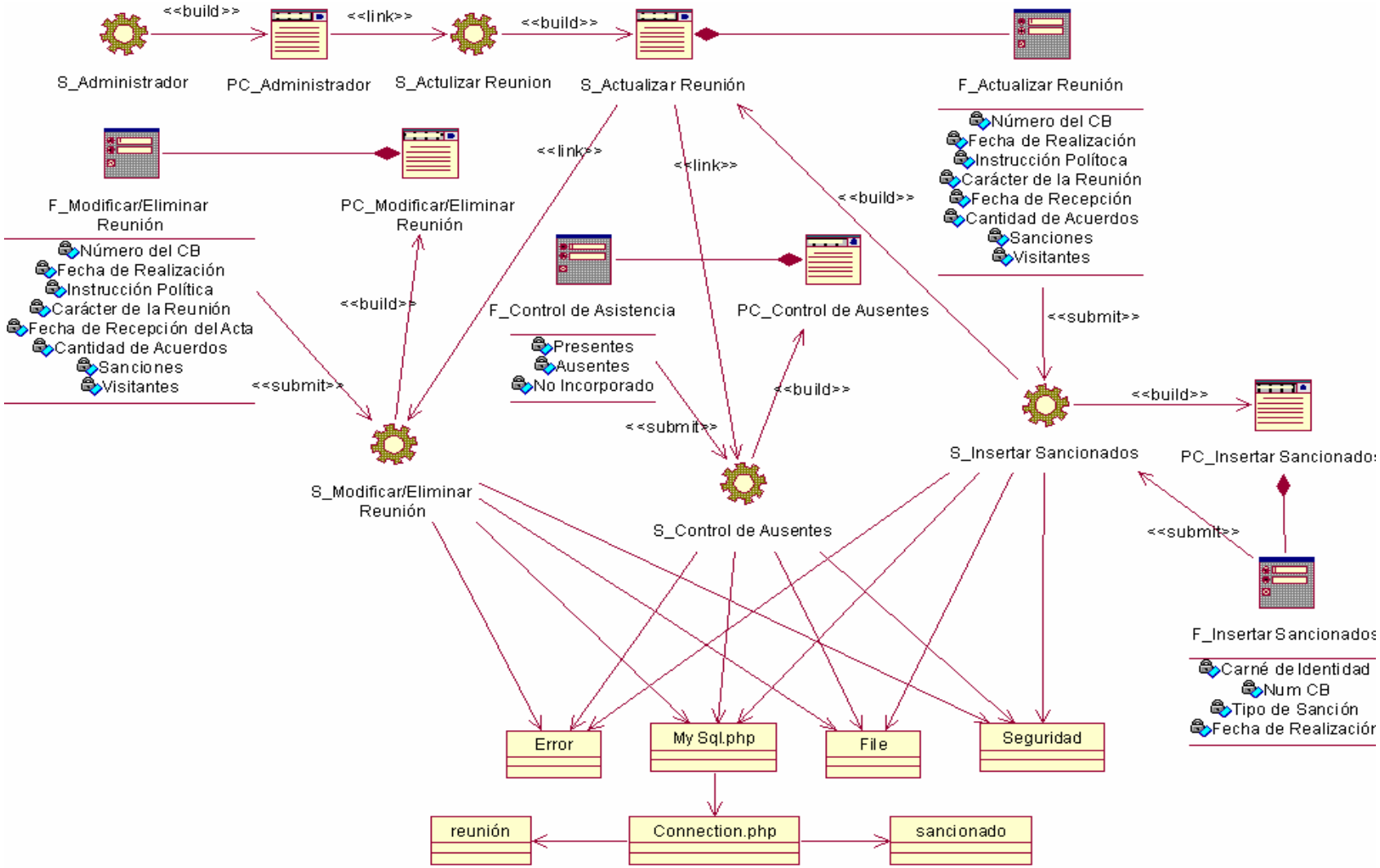
Anexo 2.3 -Actualizar cotización.



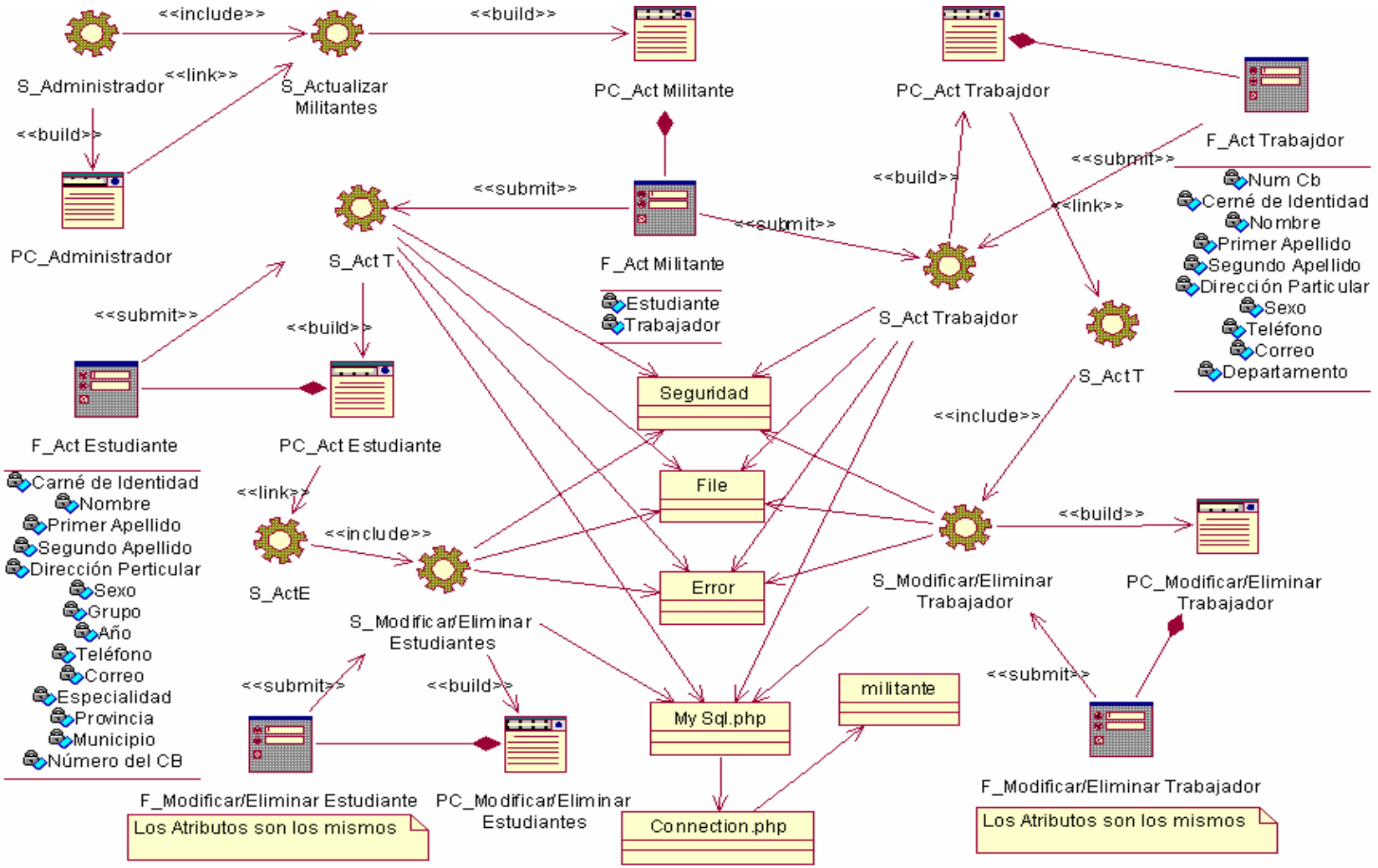
Anexo 2.4 -Actualizar sancionados.



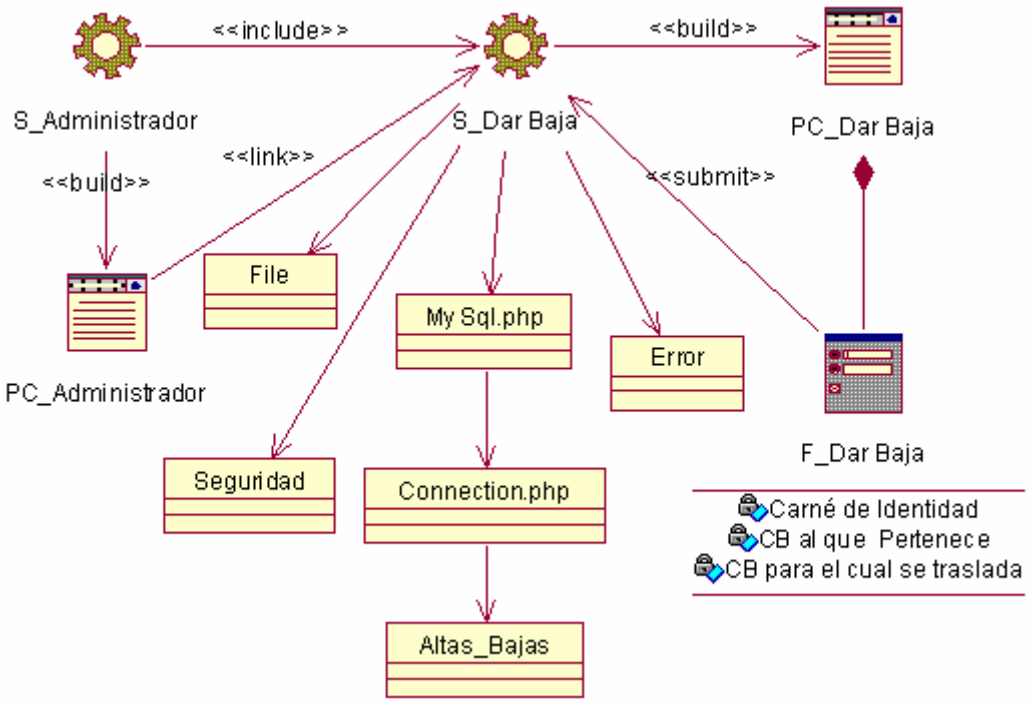
Anexo 2.5 -Actualizar datos de una reunión.



Anexo 2.6 -Actualizar militante.



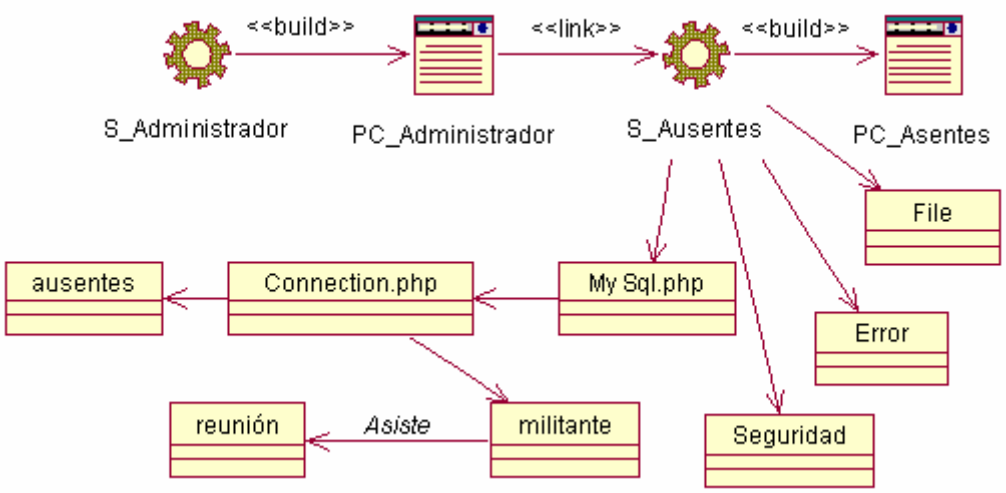
Anexo 2.7 –Dar Baja a un militante.



Anexo 3 – Diagrama de clases del diseño Web.

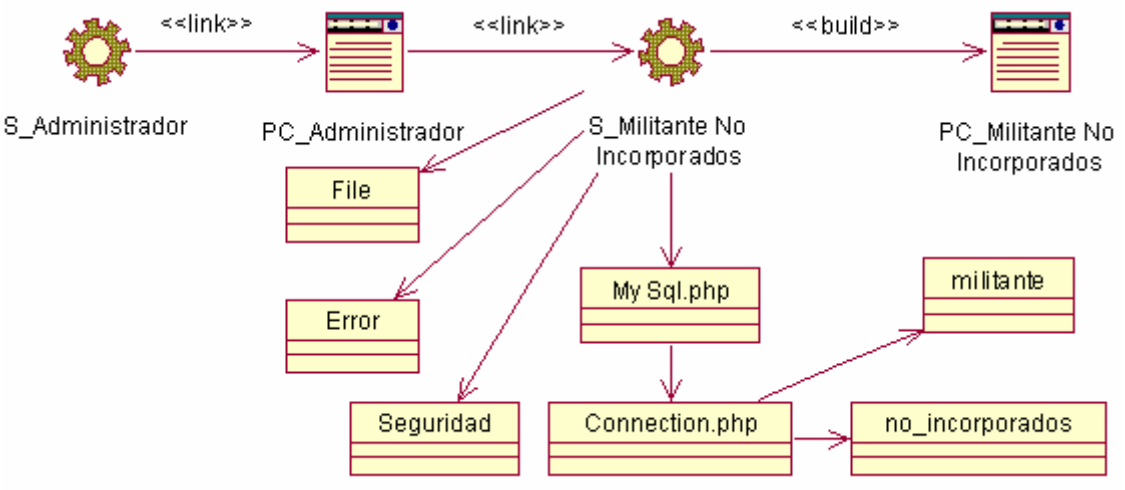
Módulo Reportes

Anexo 3.1 Consultar ausentes

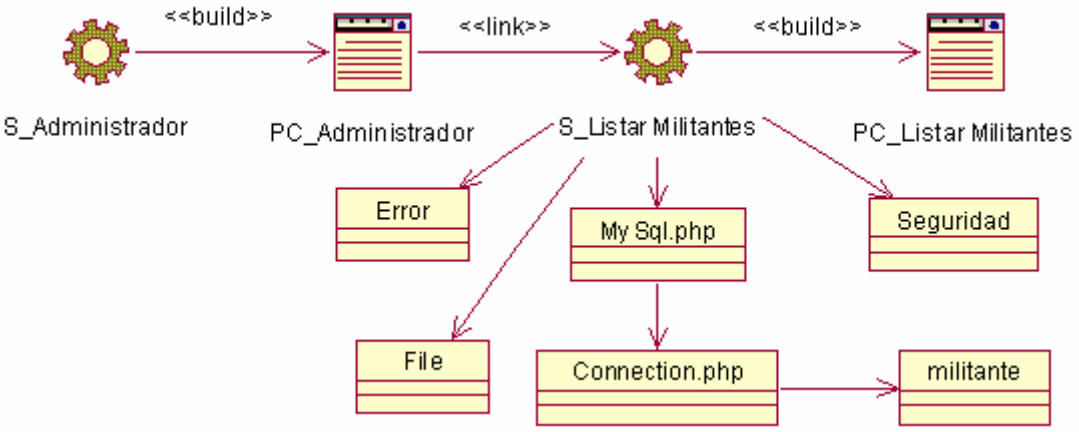


Anexo 3.2 Consultar militantes no físicos

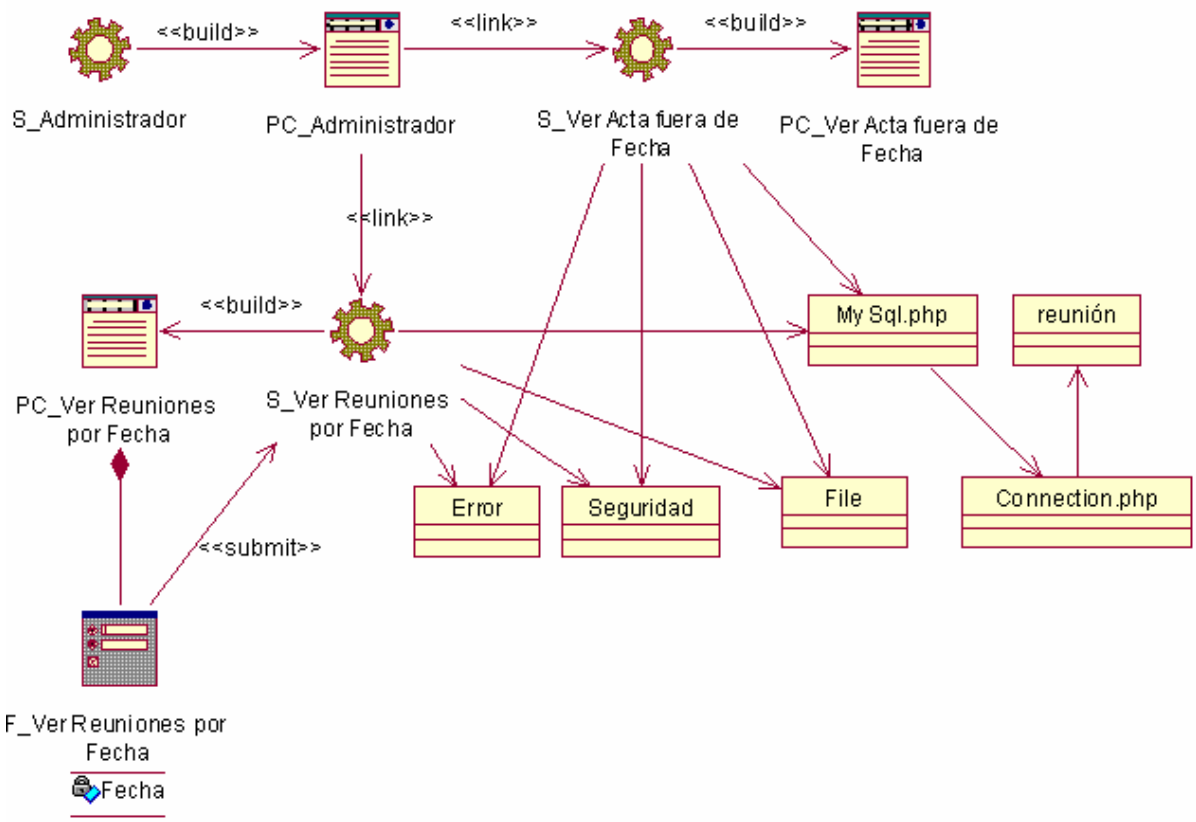
Anexo 3.3 Consultar militantes no incorporados



Anexo 3.4 Consultar Listado por comité de base

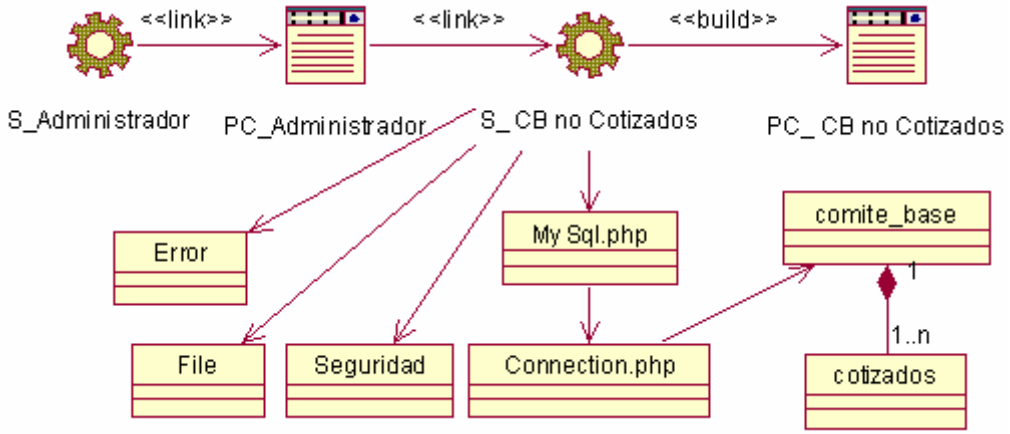


Anexo 3.5 Consultar Actas fuera de Fecha y reunión por fecha



Anexo 3.6 Consultar militantes sancionados.

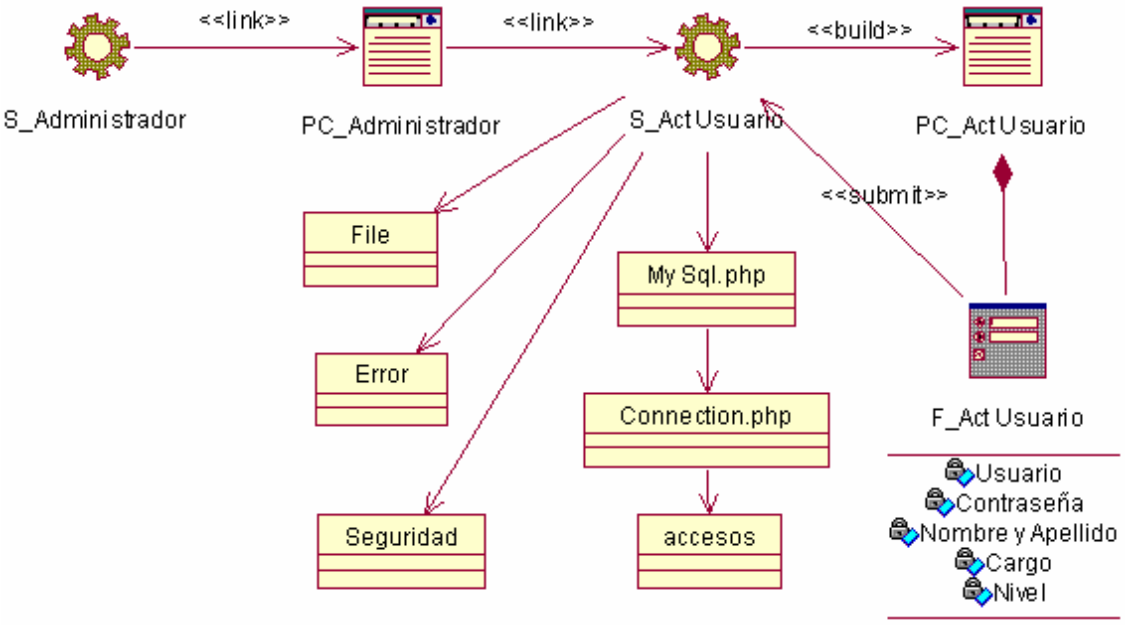
Anexo 3.7 Consultar los comités de base no cotizados.



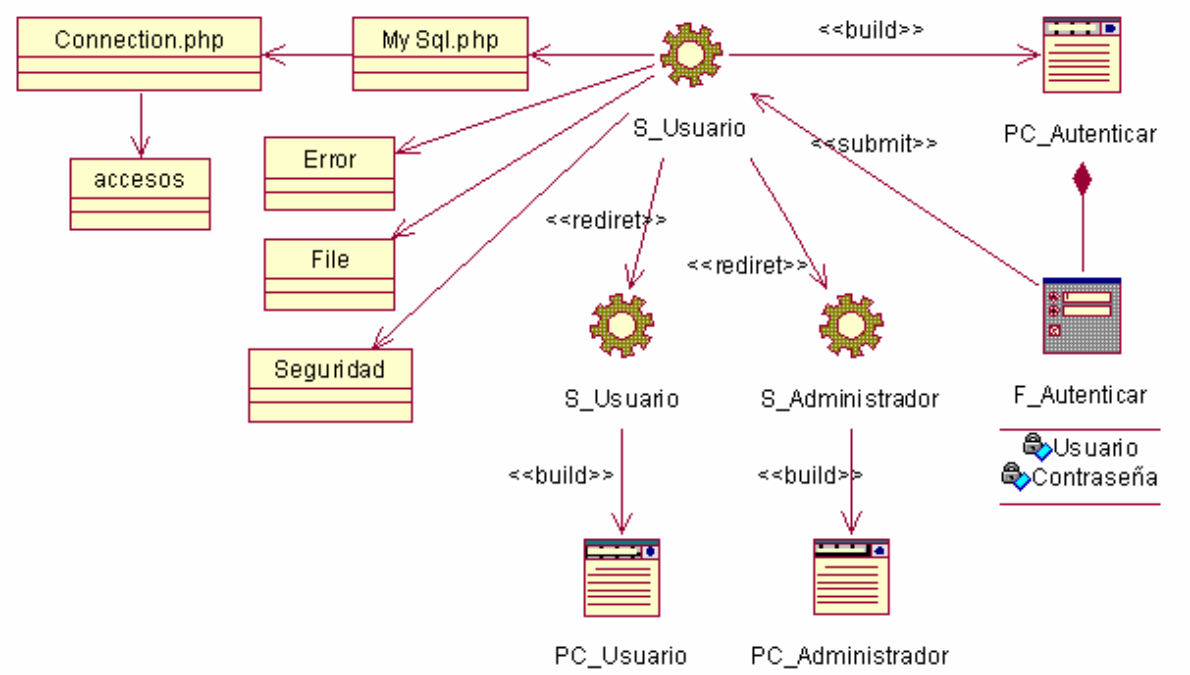
Anexo 4 – Diagrama de clases del diseño Web.

Módulo Seguridad

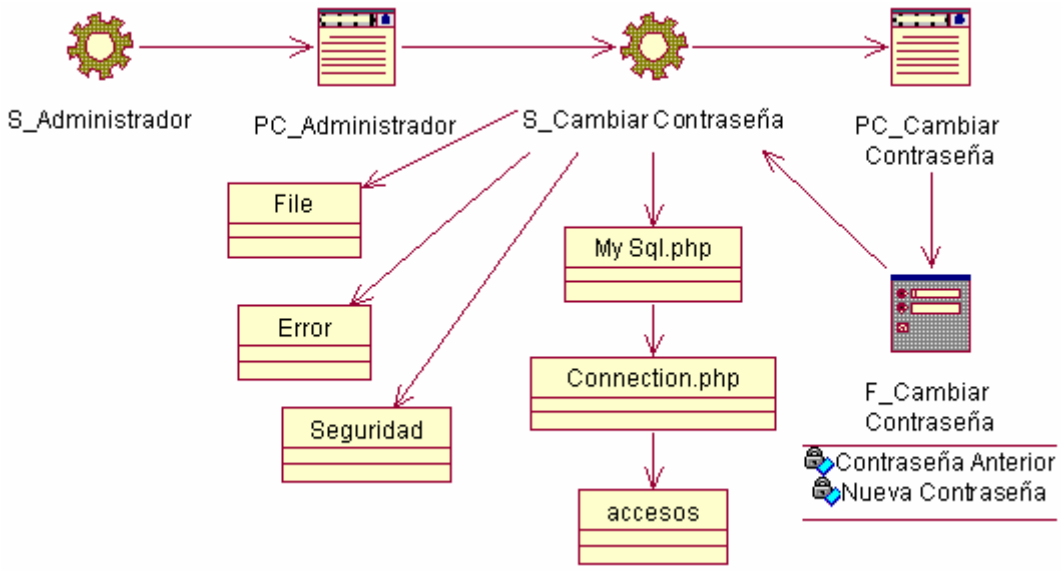
Anexo 4.1 Actualizar usuario



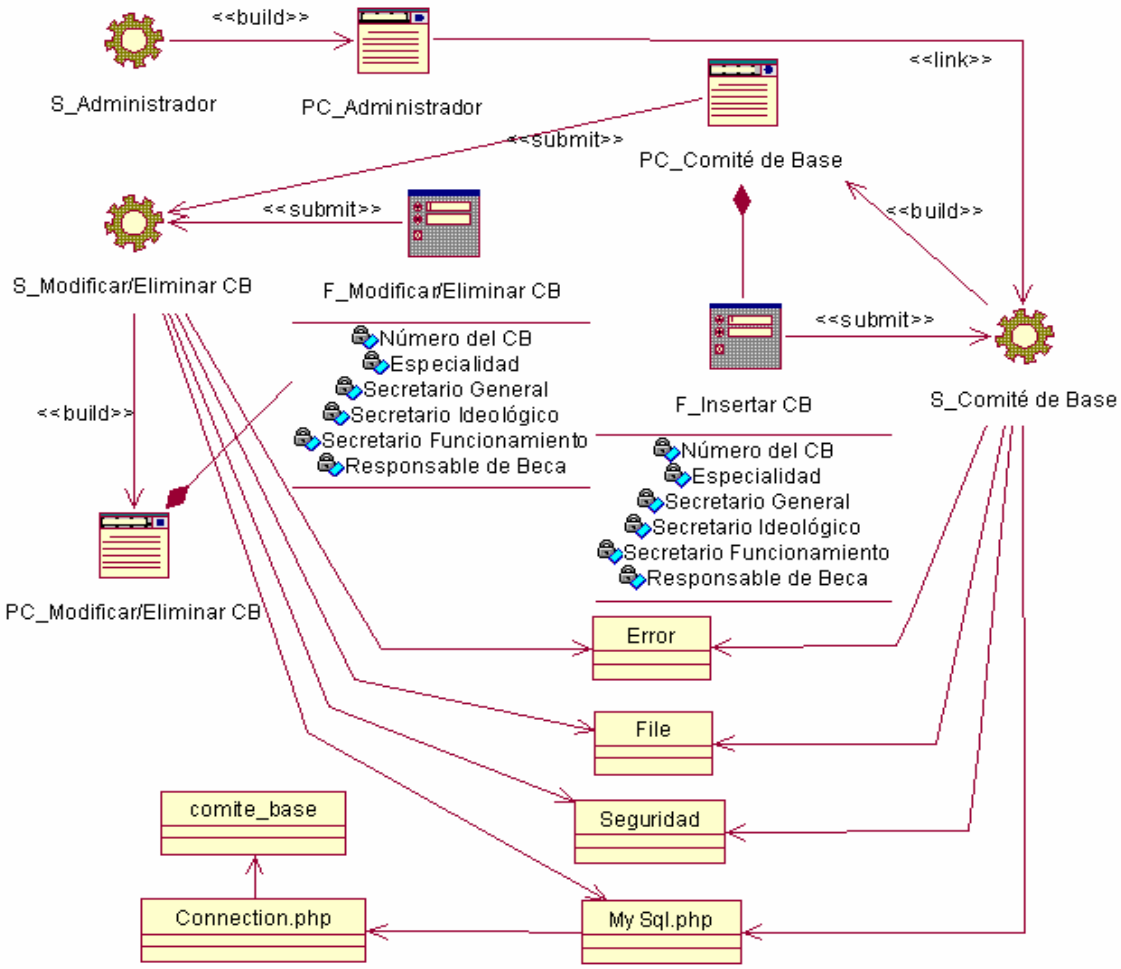
Anexo 4.2 Autenticar usuario



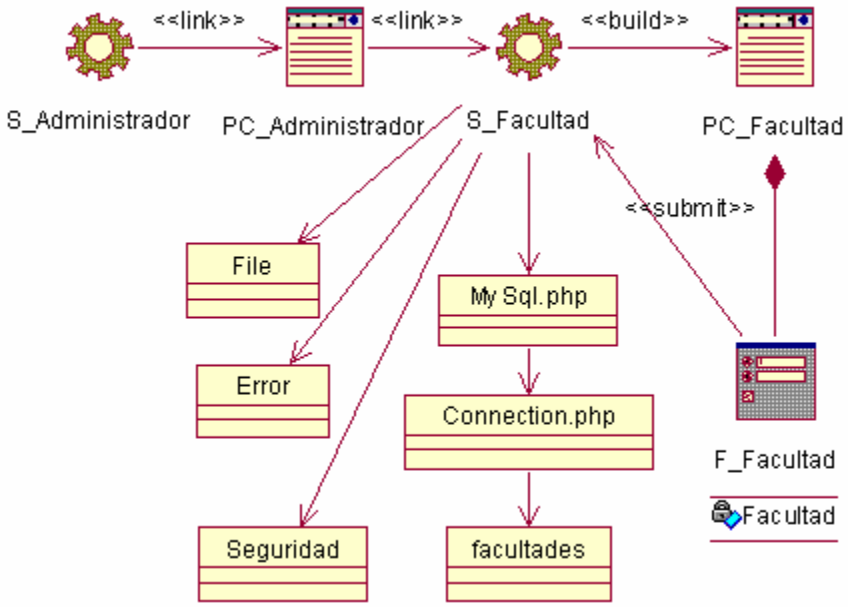
Anexo 4.3 Cambiar contraseña



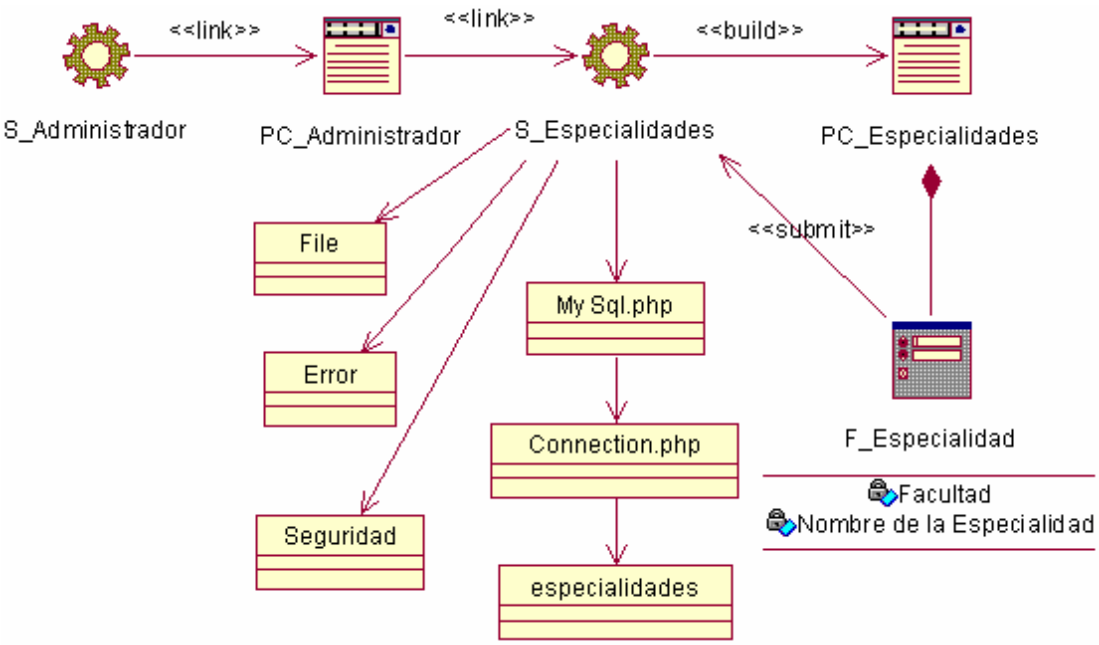
Anexo 4.4 Actualizar datos de lo comités de base



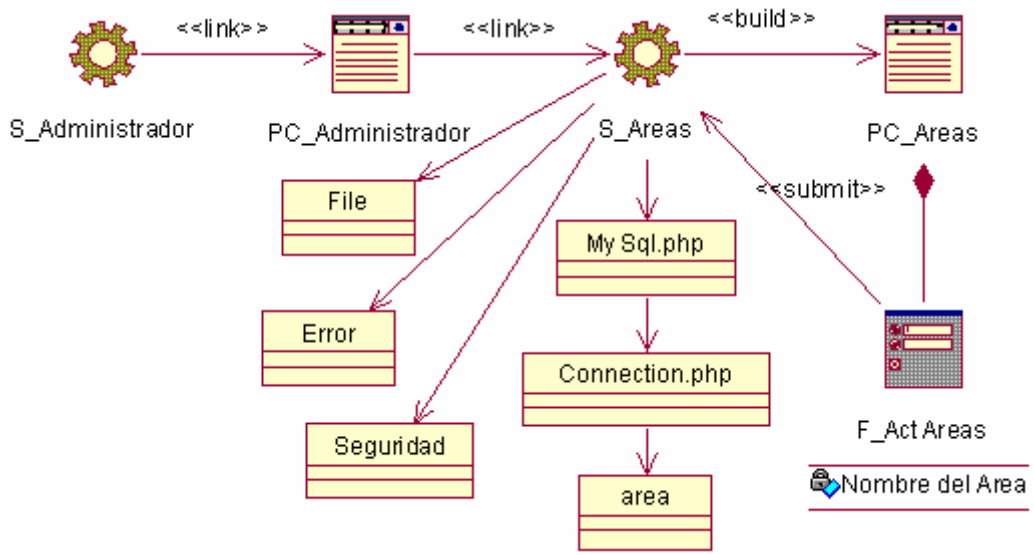
Anexo 4.5 Actualizar datos de facultad



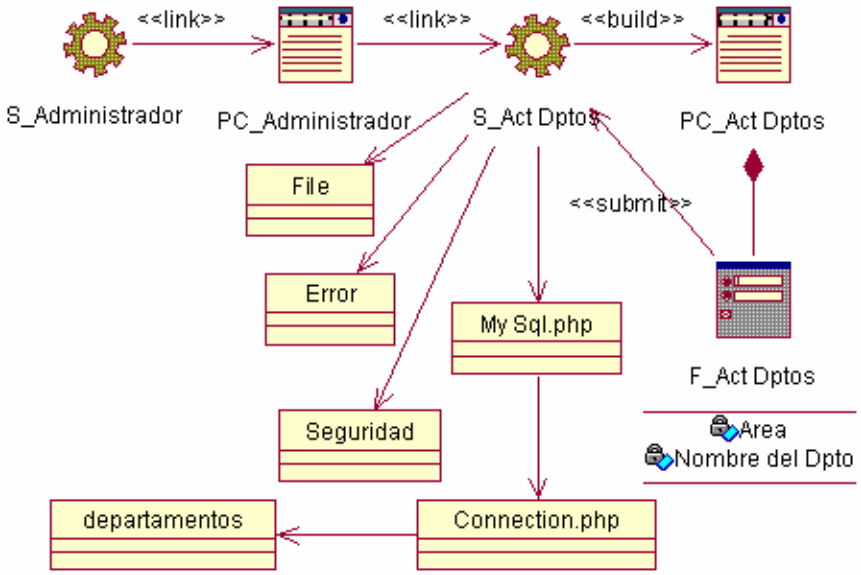
Anexo 4.6 Actualizar datos de especialidad



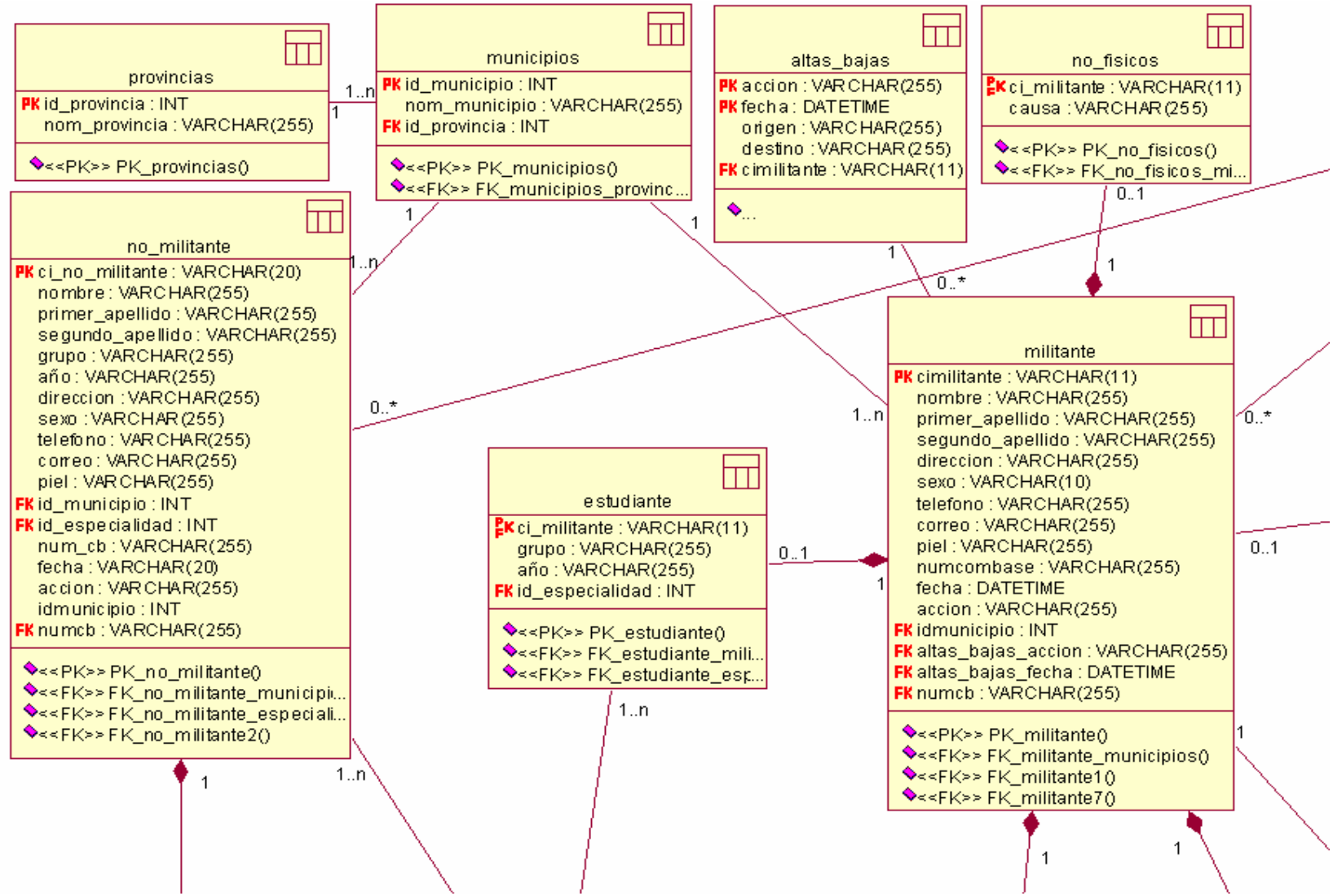
Anexo 4.7 Actualizar datos de un Área.

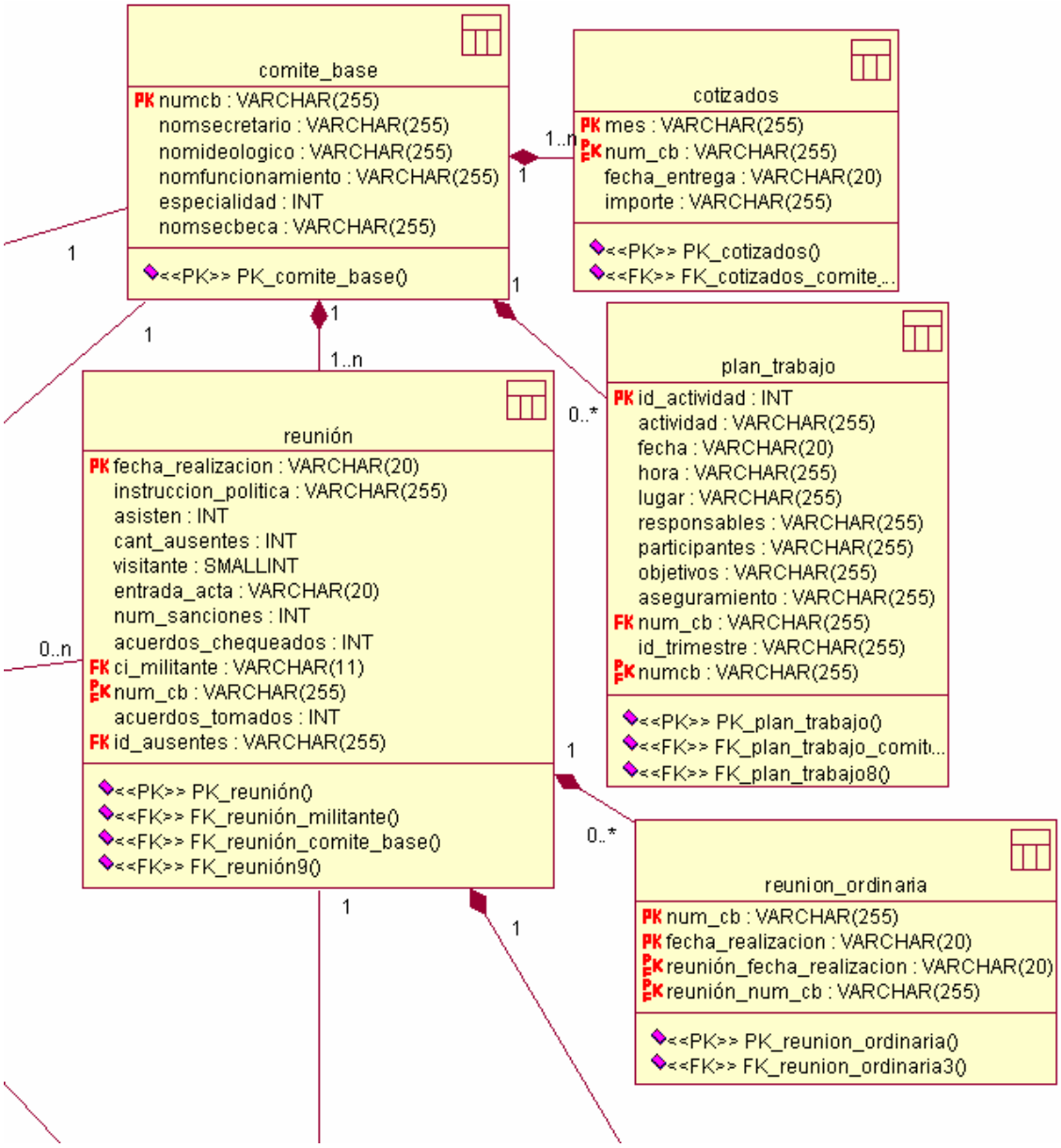


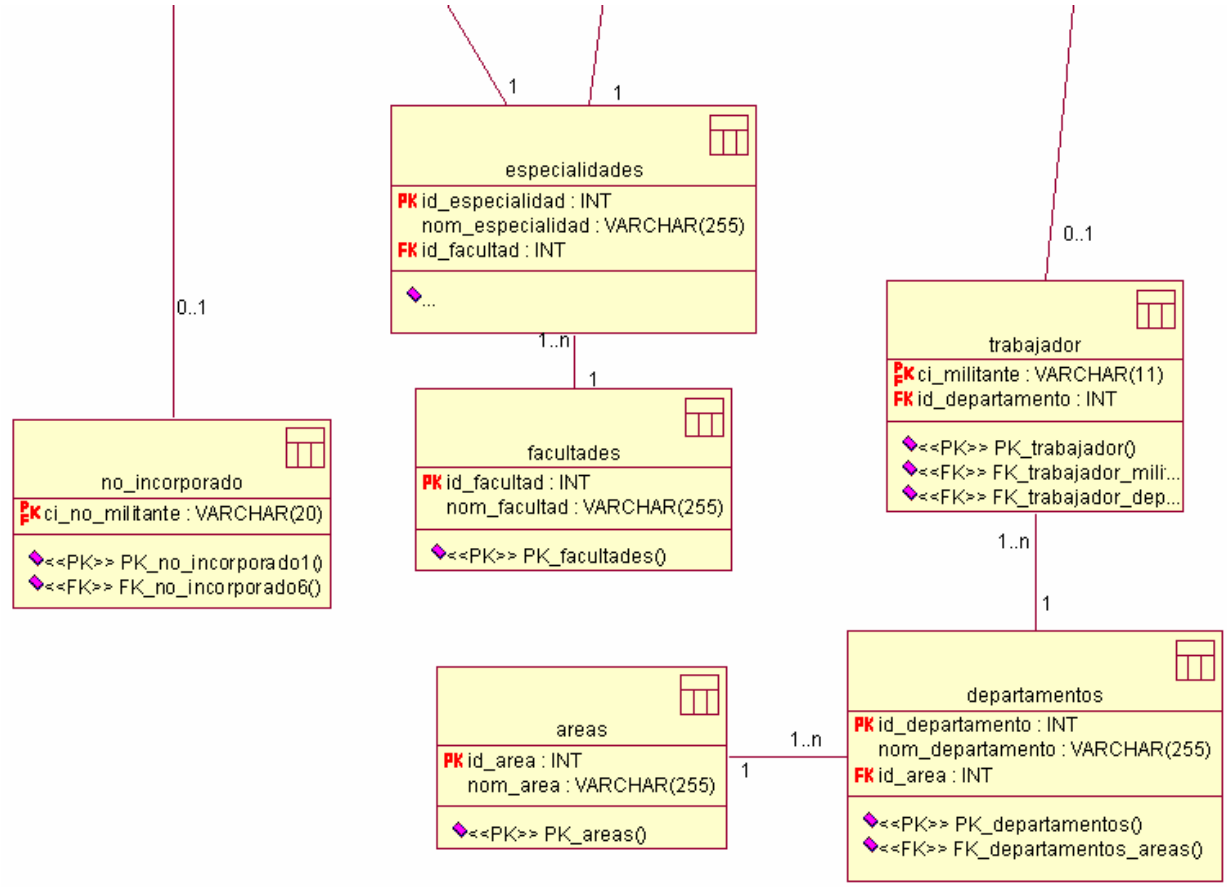
Anexo 4.8 Actualizar datos de un departamento.

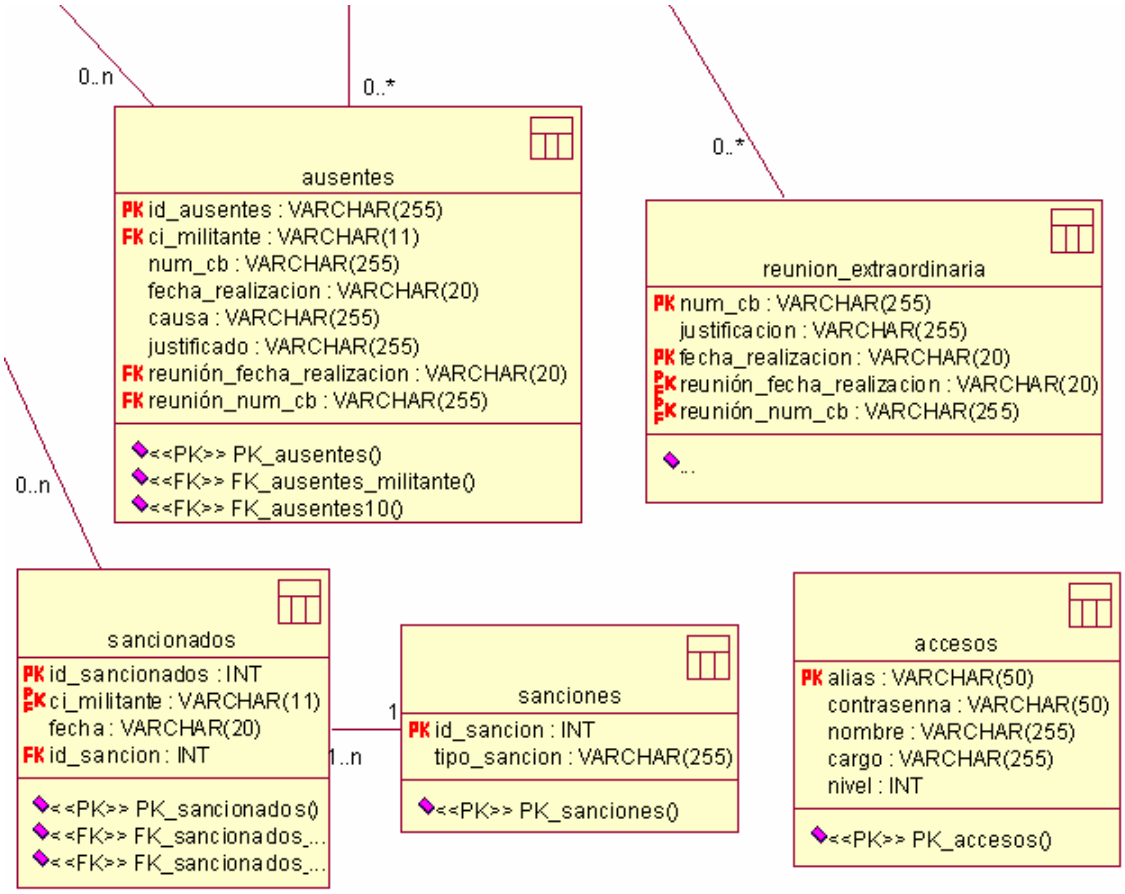
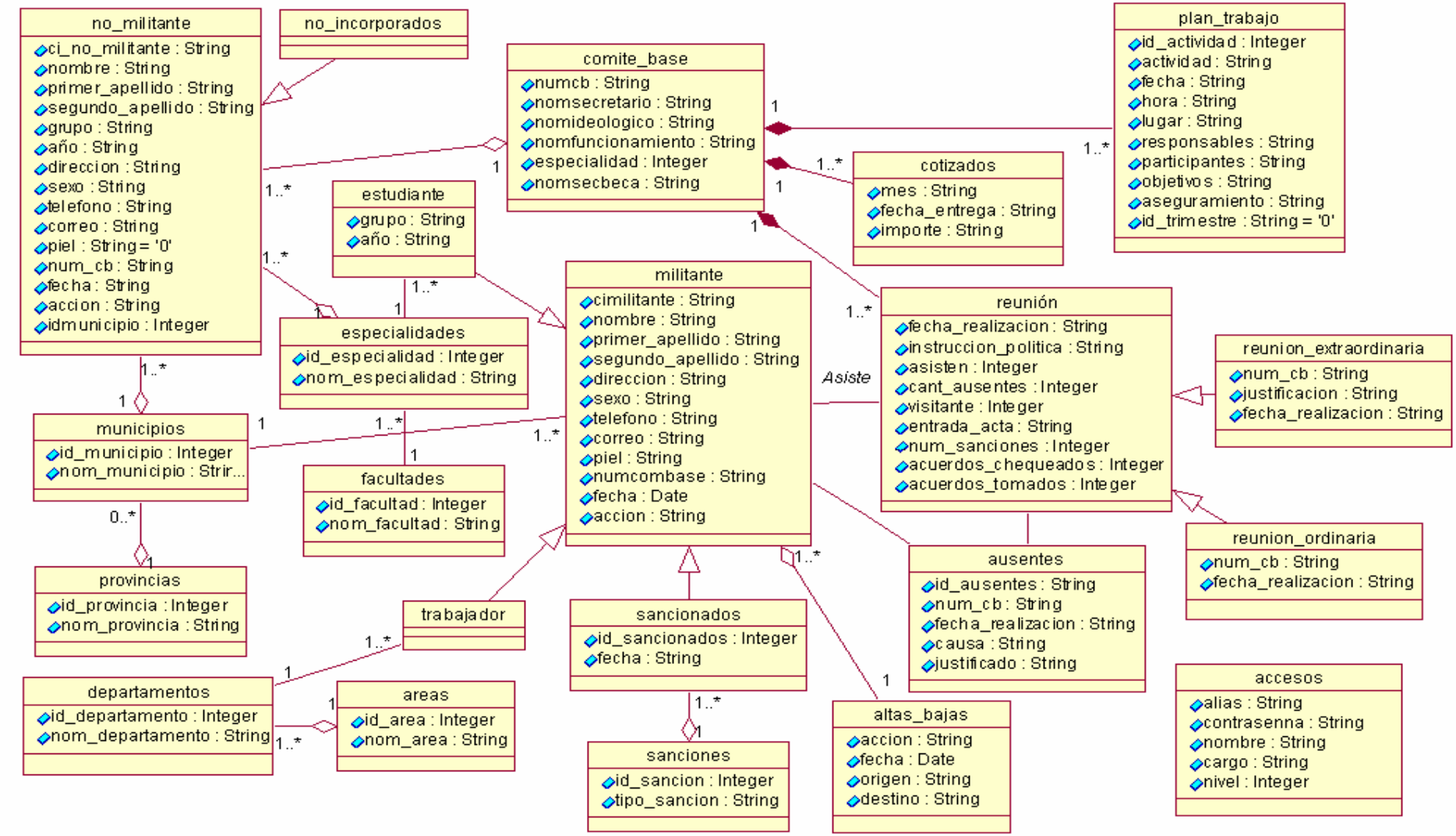


Anexo 5 –Modelo Físico de la Base de Datos.




comite_base
PK numcb : VARCHAR(255)
nomsecretario : VARCHAR(255)
nomideologico : VARCHAR(255)
nomfuncionamiento : VARCHAR(255)
especialidad : INT
nomsecbeca : VARCHAR(255)
<<PK>> PK_comite_base()
cotizados
PK mes : VARCHAR(255)
PK FK num_cb : VARCHAR(255)
fecha_entrega : VARCHAR(20)
importe : VARCHAR(255)
<<PK>> PK_cotizados()
<<FK>> FK_cotizados_comite_...
plan_trabajo
PK id_actividad : INT
actividad : VARCHAR(255)
fecha : VARCHAR(20)
hora : VARCHAR(255)
lugar : VARCHAR(255)
responsables : VARCHAR(255)
participantes : VARCHAR(255)
objetivos : VARCHAR(255)
aseguramiento : VARCHAR(255)
FK num_cb : VARCHAR(255)
id_trimestre : VARCHAR(255)
PK FK numcb : VARCHAR(255)
<<PK>> PK_plan_trabajo()
<<FK>> FK_plan_trabajo_comit...
<<FK>> FK_plan_trabajo8()
reunión
PK fecha_realizacion : VARCHAR(20)
instruccion_politica : VARCHAR(255)
asisten : INT
cant_ausentes : INT
visitante : SMALLINT
entrada_acta : VARCHAR(20)
num_sanciones : INT
acuerdos_chequeados : INT
FK ci_militante : VARCHAR(11)
PK FK num_cb : VARCHAR(255)
acuerdos_tomados : INT
FK id_ausentes : VARCHAR(255)
<<PK>> PK_reunión()
<<FK>> FK_reunión_militante()
<<FK>> FK_reunión_comite_base()
<<FK>> FK_reunión9()
reunion_ordinaria
PK num_cb : VARCHAR(255)
PK fecha_realizacion : VARCHAR(20)
PK FK reunión_fecha_realizacion : VARCHAR(20)
PK FK reunión_num_cb : VARCHAR(255)
<<PK>> PK_reunion_ordinaria()
<<FK>> FK_reunion_ordinaria3()
1
1..n
0..*
0..n


especialidades
PK id_especialidad : INT
nom_especialidad : VARCHAR(255)
FK id_facultad : INT
...
facultades
PK id_facultad : INT
nom_facultad : VARCHAR(255)
<<PK>> PK_facultades()
trabajador
PFK ci_militante : VARCHAR(11)
FK id_departamento : INT
<<PK>> PK_trabajador()
<<FK>> FK_trabajador_mili...
<<FK>> FK_trabajador_dep...
no_incorporado
PFK ci_no_militante : VARCHAR(20)
<<PK>> PK_no_incorporado1()
<<FK>> FK_no_incorporado6()
areas
PK id_area : INT
nom_area : VARCHAR(255)
<<PK>> PK_areas()
departamentos
PK id_departamento : INT
nom_departamento : VARCHAR(255)
FK id_area : INT
<<PK>> PK_departamentos()
<<FK>> FK_departamentos_areas()
1
1
0..1
1..n
1
1..n
1
0..1
1..n
1


0..n
0..*
0..*
ausentes
PK id_ausentes : VARCHAR(255)
FK ci_militante : VARCHAR(11)
num_cb : VARCHAR(255)
fecha_realizacion : VARCHAR(20)
causa : VARCHAR(255)
justificado : VARCHAR(255)
FK reunión_fecha_realizacion : VARCHAR(20)
FK reunión_num_cb : VARCHAR(255)
<<PK>> PK_ausentes()
<<FK>> FK_ausentes_militante()
<<FK>> FK_ausentes10()
reunion_extraordinaria
PK num_cb : VARCHAR(255)
justificacion : VARCHAR(255)
PK fecha_realizacion : VARCHAR(20)
PK FK reunión_fecha_realizacion : VARCHAR(20)
PK FK reunión_num_cb : VARCHAR(255)
0..n
sancionados
PK id_sancionados : INT
PK FK ci_militante : VARCHAR(11)
fecha : VARCHAR(20)
FK id_sancion : INT
<<PK>> PK_sancionados()
<<FK>> FK_sancionados_...
<<FK>> FK_sancionados_...
1
1..n
sanciones
PK id_sancion : INT
tipo_sancion : VARCHAR(255)
<<PK>> PK_sanciones()
accesos
PK alias : VARCHAR(50)
contrasenna : VARCHAR(50)
nombre : VARCHAR(255)
cargo : VARCHAR(255)
nivel : INT
<<PK>> PK_accesos()

Anexo 6 –Modelo Lógico de la Base de Datos.