

UNIVERSIDAD DE SANCTI SPÍRITUS “JOSÉ MARTÍ PÉREZ”

FACULTAD DE INGENIERÍA

CARRERA INGENIERÍA INFORMÁTICA



Título: “Aplicación Web para facilitar la gestión de la información del Buró Universitario Sindical de la Universidad de Sancti Spíritus José Martí Pérez”.

Autor: Orlando Valle Martínez.

Tutor: Dra. C. Lydia Rosa Ríos Rodríguez.

Consultantes: Ing. Irina Machado Mutis.

Sancti Spíritus

Junio de 2013

Resumen

Este trabajo está dirigido al Buró Universitario Sindical (BUS) de la Universidad de Sancti Spíritus José Martí Pérez. La necesidad de perfeccionar sus métodos para el tratamiento de la información y ofrecer mayor visibilidad a la misma, ha demandado la utilización de aplicaciones informáticas que permitan mejores resultados en el trabajo de la organización sindical. El objetivo general del trabajo es desarrollar una aplicación web que facilite la gestión de la información en este buró sindical.

Las etapas llevadas a cabo en la elaboración del software fueron las planteadas por el Proceso Unificado de Desarrollo (RUP), utilizándose como lenguaje al “Lenguaje Unificado de Modelado” (UML) y como herramienta para el análisis y diseño: el Rational Rose Enterprise Edition. La aplicación está escrita en PHP, utilizando el *framework*, para aplicaciones web Symfony 2, Twig como motor de plantilla y Doctrine2 para la gestión con la Base de Datos. Para la persistencia de los mismos se utilizó MySQL versión 5.5.8 y como herramienta de programación el NETBEANS versión 7.2.1 para PHP.

Abstract

This work is created for the University Labour Bureau (BUS in Spanish) of the University of Sancti Spíritus " José Martí Pérez". The necessity to better the methods for the treatment of the information to offer a bigger visibility to it, have demanded the use of computer applications that allows better results. The general objective of this work is to develop an application web that facilitates the management and bigger the information for and from the "BUS".

The stages carried out in the elaboration of the software were those by the Unified Process of Development (RUP), using as language the "Unified Language of Modeling" (UML) and as tools for the analysis and design: the Rational Rose Enterprise Edition. The application is written in PHP, using the framework, for applications web Symfony 2, Twig like insole motor and Doctrine2 for the administration with the Database. For the persistence of the same ones it was used MySQL version 5.5.8 and as a tool programming the NETBEANS version 7.2.1 for PHP. The final result allows offering a solution that will contribute to the execution of the goals of the University Labour Bureau.

Índice

INTRODUCCIÓN.....	1
CAPITULO I: Fundamentación teórica y metodológica que sustentan la construcción una aplicación web para facilitar la gestión de la información en el Buró Universitario Sindical de la Uniss.....	6
Introducción.....	6
1.1 Información.....	6
1.2 La Gestión de la información.....	7
1.3 La gestión de la información mediada por computadora.....	7
1.4 Las Tecnologías de la información y las comunicaciones (TIC) en la gestión de la información.....	8
1.5 Gestión de la información sindical en la Uniss.....	12
1.5.1 La Universidad “José Martí Pérez” de Sancti Spíritus. Caracterización	12
1.5.2 Organización sindical en la Uniss.....	14
1.5.3 Gestión de la información sindical en la Uniss.....	15
1.6 Metodologías utilizadas para el diseño.....	16
1.6.1 Metodología RUP.....	16
1.6.2 Lenguaje Unificado de Modelado (UML).....	17
1.7Tecnologías y Herramientas actuales para la elaboración de aplicaciones web.....	17
1.7.1 Arquitectura Cliente/Servidor.....	17
1.7.2 Patrón de diseño Modelo-Vista-Controlador.....	18
1.7.3 Frameworks.....	18
1.7.4 Tecnologías del lado del cliente.....	20
1.7.5Tecnologías del lado del servidor.....	24
Doctrine.....	24
1.7.6Herramientas de desarrollo.....	27
1.7.7 Sistemas gestores de base de datos (SGBD).....	29
1.7.8 Servidores Web.....	31
1.7.9 ¿Por qué utilizar Apache-PHP-MySQL?.....	32
Conclusiones.....	32

<i>CAPÍTULO II: Descripción de la Aplicación Web para facilitar la gestión de la información en el Buró Universitario Sindical de la Uniss.</i>	33
Introducción.....	33
2.1 Modelo del negocio.....	33
2.1.1 Identificación de los procesos del negocio	33
2.2 Reglas del negocio	35
2.3 Modelo de casos de uso del negocio.....	35
2.3.1 Actores del Negocio.....	35
2.3.2 Trabajadores del Negocio.....	36
2.3.3 Diagramas de casos de uso del Negocio.....	36
2.3.4 Descripción de los casos de uso del negocio.....	37
2.3.5 Diagramas de actividades del negocio.	40
2.4 Modelo de objetos del negocio	43
2.5 Requisitos funcionales.....	45
2.6 Requisitos no funcionales.....	47
2.7 Modelo de casos de uso del sistema.....	49
2.7.1 Actores del sistema.....	50
2.7.2 Casos de uso del sistema.....	51
2.7.3 Diagrama de casos de uso del sistema.....	52
Conclusiones.....	56
Capítulo III: Construcción de la aplicación propuesta para la gestión de la información para el BUS de la Universidad de Sancti Spiritus “José Martí Pérez”.	
.....	57
Introducción.....	57
3.1 Diagrama de clases Web.....	57
3.2 Diseño de la Base de Datos.....	74
3.3 Modelo de Implementación	76
3.4 Principios de diseño.....	79
Conclusiones.....	81
Conclusiones Generales:	82
Recomendaciones:.....	83
Bibliografía	84
Anexos:	87

Índice de Figuras

Figura 1: Diagrama de Caso de Uso del Negocio	36
Figura 2: Diagrama de Actividad: Caso de uso Obtener Información.....	41
Figura 3: Diagrama de Actividad: Caso de uso Pagar Cotización.....	42
Figura 4: Diagrama de Actividad: Caso de uso Pagar Día Haber.	43
Figura 5: Modelo Objeto Caso de Uso Obtener Información.....	44
Figura 6: Modelo Objeto Caso de Uso Pago de Cotización.	44
Figura 7: Modelo Objeto Caso de Uso Pago de Día Haber.....	45
Figura 8: Casos de uso por paquetes.	53
Figura 9: Diagrama de casos de uso del sistema Paquete de Seguridad	54
Figura 10: Diagrama casos de uso del Sistema Paquete Nomencladores.....	55
Figura 11: Diagrama casos de uso del Sistema Paquete Reporte.....	56
Figura 12: Diagrama de clases Web Caso de Uso Cerrar Sesión.....	58
Figura 13: Diagrama de clases Web Caso de Uso Autenticación	59
Figura 14: Diagrama de clases Web Caso de Uso Cambiar Contraseña.....	60
Figura 15: Diagrama de clases Web Caso de Uso Gestionar Usuario.....	61
Figura 16: Diagrama de clases Web Caso de Uso Gestionar Noticia	62
Figura 17: Diagrama de clases Web Caso de Uso Subir Documento	63
Figura 18: Diagrama de clases Web Caso de Uso Gestionar Integrantes	64
Figura 19: Diagrama de clases Web Caso de Uso Gestionar Sección	65
Figura 20: Diagrama de clases Web Caso de Uso Gestionar Cotización	66
Figura 21: Diagrama de clases Web Caso de Uso Gestionar Día de Haber.....	67
Figura 22: Diagrama de clases Web Caso de Uso Gestionar Área de trabajo.....	68
Figura 23: Diagrama de clases Web Caso de Uso Gestionar Efemérides	69
Figura 24: Diagrama de clases Web Caso de Uso Gestionar Destacados	70
Figura 25: Diagrama de clases Web Caso de Uso Descargar Documento	71
Figura 26: Diagrama de clases Web Caso de Uso Reportar Sección	72
Figura 27: Diagrama de clases Web Caso de Uso Reportar Integrantes por Sección	73
Figura 28: Diagrama de clases Web Caso de Uso Reportar Usuarios.....	74
Figura 29: Diagrama de clases persistentes	75
Figura 30: Diagrama de Modelo físico de datos	76
Figura 31: Diagrama de Despliegue.....	77
Figura 32: Diagrama de Componentes	78

Índice de Tablas

Tabla 1: Actor del negocio.....	35
Tabla 2: Trabajadores del negocio	36
Tabla 3: Descripción del Caso de Uso Obtener Información.....	38
Tabla 4: Descripción del Caso de Uso Pagar de cotización.....	39
Tabla 5: Descripción del Caso de Uso Pagar de día haber.....	40
Tabla 6: Actores del Sistema.....	50
Tabla 7: Descripción de los paquetes	53

INTRODUCCIÓN.

Con el triunfo de la Revolución cubana en 1959, se creó la base política y social que ha favorecido grandes transformaciones sociales, entre las que se encuentran las que estimulan el desarrollo institucional de las universidades y sus colectivos de trabajadores. Estos tienen la misión de mantener los logros educacionales y las conquistas del proyecto de desarrollo social que realiza el país.

En Cuba la Informatización de la Sociedad es un imperativo para el desarrollo en las condiciones del mundo actual. Esto influye de manera creciente en todos los aspectos de la vida de las instituciones y los ciudadanos, por lo que los Gobiernos locales y las entidades laborales deben potenciar las acciones para aprovechar las capacidades de cada lugar, a partir de una correcta estrategia de informatización que se identifique con las particularidades de cada lugar. (Partido Comunista de Cuba [PCC] ,2000)

En atención a lo indicado en la Resolución Económica del VI Congreso del Partido Comunista de Cuba y de forma específica en los Lineamientos para la política económica y social (PCC, 2011). “El país debe encaminarse a la modernización informática mediante un programa integral que involucre a las organizaciones que deben proveer los recursos materiales, financieros e intelectuales y a las entidades económicas, políticas y sociales que deben traducirlos en más y mejores servicios. La industria de los servicios informáticos deberá asegurar la modernidad de su base técnica y organizativa y la elevación constante del nivel científico técnico de sus especialistas con vistas a garantizar esos propósitos. Las telecomunicaciones (...) deberán continuar aumentando la calidad de sus servicios y garantizar el mejoramiento progresivo de su eficiencia en beneficio de la economía y la población”.

La Universidad Cubana tiene como estrategia, el desarrollo de sus procesos; es estimulada por la voluntad política que propicia llevar la cultura a toda la sociedad, manteniendo altos los principios de equidad y justicia social. Las transformaciones que se realizan deben tener en cuenta los retos y desafíos que plantea la crisis

capitalista actual y las propias necesidades y debilidades que se manifiestan en el trabajo cotidiano de las organizaciones y entidades.

El Sindicato Nacional de la Educación, la Ciencia y el Deporte (SNTECD) es la organización que agrupa a los trabajadores de la educación y de las universidades cubanas. La Central de Trabajadores de Cuba (CTC) ha definido que las misiones de los sindicatos en nuestra sociedad socialista están vinculadas con la necesidad de organizar a los trabajadores, educarlos y prepararlos para movilizarlos en el cumplimiento de la actividad laboral y social del país, con el propósito de construir el socialismo en el país. Además representan a los trabajadores ante el Estado y la sociedad. (SNTECD, 2011)

En las universidades de Cuba el SNTECD se estructura en secciones sindicales vinculadas a un buró universitario sindical (BUS). En el cumplimiento de su misión los sindicatos deben encontrar vías que permitan el contacto y la comunicación con sus afiliados, facilitando la visibilidad del trabajo sindical.

Encontrar las mejores vías de comunicación e interacción entre sus miembros es un imperativo para las organizaciones. El mundo de hoy está cada vez más globalizado y el desarrollo económico y social está condicionado por factores insoslayables, como los vinculados con el desarrollo del conocimiento, la investigación y el progreso de la ciencia y la tecnología, lo que obliga a organizaciones como el SNTECD a utilizar las mejores aplicaciones informáticas posibles para mejorar su trabajo.

Las tecnologías vinculadas a la informática son un buen precedente para este fin. Su uso ascendente y su ampliación a toda la sociedad han contribuido de manera notable en los resultados e indicadores económicos y sociales. En este sentido destacan los avances en las investigaciones vinculadas con la informática, a partir de los primeros equipos considerados enormes, costosas y hoy arcaicas computadoras de la década de los años 1960 – 1970, seguidas por las mini de los años 70 – 80 y desde los años 90 y hasta hoy, con las computadoras (PC); en este último período destaca además el desarrollo alcanzado en la Internet.

La existencia y desarrollo de Internet y luego de las redes sociales ha constituido un catalizador del desarrollo global. Las relaciones interpersonales, de grupos y

organizacionales se han beneficiado, incrementándose las posibilidades de comunicación a niveles nunca vistos hasta este momento.

Los antecedentes históricos muestran que en 1971 se envió el primer e-mail entre dos ordenadores; en 1978 Ward Christensen y Randy Suess crean el Bulletin Board Systems (BBS) para informar a sus amigos sobre reuniones y compartir información (Olmedo Ramos, 2001); luego en 1994 se lanza GeoCities, un servicio que permite a los usuarios crear sus propios sitios web y alojarlos en determinados lugares según su contenido. (Harasim, 1995)

Sobre la Web se expresa por, (De la Iglesia, 2012) que alcanzó el millón de sitios web en 1995; en el año 2000 se llegó a la cifra de setenta millones de ordenadores conectados en Red; para el 2010 los usuarios de Internet se estimaron en 1,97 billones, casi el 30% de la población mundial. En el año 2012 más de 800 millones de personas utilizaron servicios de intercambio de información.

La expresión cubana de estas transformaciones es consecuente con lo expresado por (Castro R, 2004) cuando señaló “(...) Por lo anterior las universidades y por supuesto, los sujetos que en ella intervienen, tienen la responsabilidad de propiciar transformaciones esenciales en su gestión, para hacer de la labor intelectual, del trabajo formativo, la investigación, la extensión y la calidad de vida de los trabajadores en la institución, la base del cambio hacia un nuevo modelo de desarrollo social.”

La universidad no puede verse sólo como el centro formador de hombres, sino también de cultura, de producción científica e innovación tecnológica, vinculada siempre con la sociedad. (Valle Fasco, 2007). El sindicato universitario desempeña un importante rol en esta dirección, logrando por lo que requiere de una mayor aplicación de la informática en función del perfeccionamiento de su gestión.

Estos aspectos se evidencian en los resultados obtenidos mediante el diagnóstico constituyendo las insuficiencias fundamentales las siguientes:

- No están cubiertas todas las expectativas y necesidades de información de los afiliados al sindicato.
- Aun no se logra una legítima y genuina incorporación plena de los factores en acciones que permitan la comunicación permanente entre los afiliados y de ellos con sus directivos.
- Las condiciones de trabajo en las universidades no facilita el interactuar físico entre los miembros
- No existe una completa identificación con las posibilidades del trabajo sindical en la comunidad universitaria.

Las necesidades del movimiento obrero, los actuales retos políticos y sociales y las insuficiencias anteriores tienen necesariamente que contar con propuestas que contribuyan a su mejoría y perfección, por lo que la investigación, luego de valorada la información referente al Buró Universitario Sindical, identifica el siguiente **problema**: ¿Cómo facilitar el proceso de gestión de la información en el Buró Universitario Sindical de la Universidad de Sancti Spíritus (Uniss)? El diagnóstico causal del problema mencionado ofrece criterios vinculados con lo siguiente:

- Poca comprensión por parte de los líderes sindicales de la institución sobre las aplicaciones informáticas que pudieran permitir una mejor gestión de la organización.
- No se ha desarrollado adecuadamente una cultura informática en el seno de la organización sindical
- Las estructuras sindicales no utilizan la intranet de la Uniss como una vía de gestión de la labor sindical
- Los afiliados no poseen información en la red informática referida al trabajo del sindicato, lo que no contribuye a una mejor comunicación.

En consecuencia la investigación se plantea como **objeto de estudio** el proceso de gestión de la información vinculado al estudio de las tecnologías actuales e identifica el **campo de acción** como el proceso de gestión de la información

relacionada con el Buró Universitario Sindical. Se propone como **objetivo general** desarrollar una aplicación web para facilitar la gestión de información en el Buró Universitario Sindical de la Uniss.

Para cumplir el objetivo general se realizan las siguientes **Preguntas Científicas**:

- ¿Qué fundamentos teóricos y metodológicos sustentan la construcción de una aplicación web para facilitar la gestión de información?
- ¿Cómo diseñar una aplicación web para facilitar la gestión de información en el Buró Universitario Sindical de la Uniss?
- ¿Cómo implementar una aplicación web para facilitar la gestión de información en el Buró Universitario Sindical de la Uniss?

Para dar cumplimiento a las preguntas científicas se plantean las siguientes **Tareas de Investigación**:

- Determinar los fundamentos teóricos y metodológicos que sustentan la construcción una aplicación web para facilitar la gestión de información.
- Diseñar una aplicación web para facilitar la gestión de información en el Buró Universitario Sindical de la Uniss.
- Implementar una aplicación web para facilitar la gestión de información en el Buró Universitario Sindical de la Uniss.

El documento que respalda la investigación está compuesto por tres capítulos. Incluye un análisis del diseño y de la propuesta que se realiza. Contiene además la introducción, un primer capítulo referido a la fundamentación teórica, un segundo capítulo con el estudio de la propuesta de la aplicación web y un tercero vinculado con los resultados que se esperan y las conclusiones. Además el documento cuenta con un apéndice para anexos con imágenes, gráficas y tablas que le ayudarán a un mejor entendimiento del problema y de la propuesta que se realiza.

CAPITULO I: Fundamentación teórica y metodológica que sustentan la construcción una aplicación web para facilitar la gestión de la información en el Buró Universitario Sindical de la Uniss.

Introducción

La gestión de la información constituye una necesidad y prioridad para las empresas, instituciones y organizaciones. En el plano individual representa la forma fundamental para el desarrollo del conocimiento. Los avances tecnológicos y el desarrollo cultural alcanzado permiten utilizar nuevas técnicas y procedimientos, lo que facilita la utilización de la cultura acumulada por la humanidad y colocarla en función de las necesidades propias del investigador, el estudiante o el lector.

1.1 Información.

Existen varios conceptos referidos a la información expresados por diferentes autores. Para (Deleuze, 2012) la información es el sistema de control, en tanto que es la propagación de consignas que deberíamos de creer o hacer que creemos. En tal sentido la información es un conjunto organizado de datos capaz de cambiar el estado de conocimiento en el sentido de las consignas transmitidas.

Desde el punto de vista de la ciencia de la computación la información es, según (Montuschi, 2012) un conocimiento explícito extraído por seres vivos o sistemas expertos como resultado de la interacción con el entorno o percepciones sensibles del mismo. En el mismo se expresa que en principio, la información a diferencia de los datos o las percepciones sensibles, tienen estructura útil que modificará las sucesivas interacciones del ente que posee dicha información con su entorno. Se añade que la información puede ser conceptualizada como un conjunto organizado de datos procesados que ofrecen un mensaje con un nuevo conocimiento para la persona o sistema que lo recibe.

Para cualquier organización, entidad o empresa la organización de la información representa una prioridad en su trabajo. La introducción de la computación en estas áreas significó un importante ascenso en su nivel de desarrollo. En un breve período y con la llegada de Internet se cambió la forma en que se procesaba y utilizaba la información a través de la historia. Una de las ventajas que se alcanza

en esta etapa es la posibilidad de que muchas personas tengan acceso a ella y que además puedan vincularse entre sí de una forma segura, y ágil.

En la universidad el conocimiento y la información son básicos para su desarrollo. Una correcta organización de la información permite a sus miembros interactuar entre sí; la estructura de mando de la organización podrá coordinar mejor sus decisiones facilitándose el proceso de información y la gestión estará privilegiada con el uso de una potente herramienta de trabajo.

1.2 La Gestión de la información.

La Gestión de la Información no tiene una diferenciación clara con la gestión documental o con la llamada archivística. La primera surge a mediados de la década de los años setenta del siglo XX, momento este en que los sistemas informáticos comienzan a ser comunes en las organizaciones según la opinión de (Pulgarín Guerrero & Muñoz Cañavate, 2009). Este término se utiliza cuando se quiere hacer énfasis en un modelo de gestión documental que, además de los elementos tradicionales, involucra las tecnologías de la información y las comunicaciones. En este sentido se tienen en cuenta además la organización, almacenamiento, y recuperación de información.

A manera de resumen se puede decir que la Gestión de la Información, (GI), es un conjunto de procesos que permiten conducir o direccionar el ciclo de vida de la información; este se inicia con la obtención de la información - por creación o captura, hasta su disposición final - archivada o eliminada.

Los procesos de la GI también comprenden la extracción, combinación, depuración y distribución de la información a los interesados, mientras que sus objetivos expresan la necesidad de garantizar la integridad, disponibilidad y confidencialidad de la información.

1.3 La gestión de la información mediada por computadora.

El proceso de gestión de la información puede ser más rápido, y de mayor alcance utilizando las actuales tecnologías de la información. Mediante las computadoras, según expresa (Rojas Mesa, 2006), es posible utilizar la información existente en

bases de datos computarizadas a gran distancia, compartir criterios con otros usuarios informáticos y someter la información a un proceso de comparación y validación a partir del criterio de diferentes autores.

El uso de la computación puede permitir utilizar las posibilidades de la navegación en redes informáticas, internet o el intercambio de ideas a través de diferentes vías, todo lo cual permite a su vez compartir la información y precisarla.

En las universidades el saber o el conocimiento acumulado se colocan en función de la formación, la investigación y la extensión universitaria. Estas instituciones deben trabajar en función de la pertinencia social, o sea respondiendo a las necesidades y demandas de la sociedad a nivel local, de país y global. El uso adecuado de la información permitirá alcanzar los resultados esperados.

Gestionar la información de forma adecuada mediante computadoras permitirá a los profesores y estudiantes utilizar información actualizada, con alto rigor científico, compartir criterios, obtener conocimientos y a su vez divulgar los mismos.

En la actualidad todas las organizaciones realizan inversiones que le permitan mejorar la gestión de la información mediante la informática, lo que las hace más competitivas en un mundo cada vez más globalizado.

1.4 Las Tecnologías de la información y las comunicaciones (TIC) en la gestión de la información.

Las tecnologías de la información y la comunicación (TIC) agrupan los elementos y las técnicas usadas en el tratamiento y la transmisión de las informaciones, principalmente vinculadas a la informática, internet y telecomunicaciones.

En la ONU (Annan, 2003) expresó: "Las tecnologías de la información y la comunicación no son ninguna panacea ni fórmula mágica, pero pueden mejorar la vida de todos los habitantes del planeta".

Se denominan tecnologías de la información y la comunicación al conjunto de tecnologías que permiten la adquisición, producción, almacenamiento, tratamiento, comunicación, registro y presentación de informaciones, en forma de voz,

imágenes y datos contenidos en señales de naturaleza acústica, óptica o electromagnética. Las TIC incluyen la electrónica como tecnología base que soporta el desarrollo de las telecomunicaciones, la informática y el audiovisual. (Guitert i Catasús, Romeu, & Pérez-Mateo, 2007)

Las redes de computadoras:

El método más elemental para realizar una conexión a internet es el uso de un modem en un acceso telefónico básico. De acuerdo a lo expresado (Stallings, 2012) esta forma a pesar que no tiene todas las ventajas características de la banda ancha, ha sido el punto de inicio para muchos internautas, y es una alternativa básica para zonas de menor poder adquisitivo.

En sus inicios la banda ancha hacía referencia a la capacidad de acceso a internet superior a los de un acceso analógico (56 Kbps en un acceso telefónico básico o 128 kbps en un acceso básico RDS). Este concepto varió con el tiempo en paralelo a la evolución tecnológica. En la Comisión Federal de Comunicaciones de los EEUU se considera la banda ancha como el acceso que ocurre a una velocidad igual o superior a los 200 kilobyte por segundos (Kbps), como mínimo en un sentido. Para la Unión Internacional de telecomunicaciones el umbral se sitúa en los 2 megabyte por segundo.

Atendiendo a los países existen diferentes tecnologías: la llamada FTTH (fibra óptica hasta el hogar), cable (introducido en principio por distribución de TV), el satélite, el RDSL (soportado en la red telefónica tradicional) y otros en fase de desarrollo. El modelo de desarrollo de la conectividad en cada país ha sido diferente, y las decisiones de los reguladores de cada país han dado lugar a diferentes estructuras de mercado.

Internet está evolucionando muy rápidamente con un gran incremento de contenidos pesados (videos, música...) Por este motivo, los operadores se están encontrando en muchas ocasiones que las redes tradicionales no tienen suficiente capacidad para soportar con niveles de calidad adecuada el tránsito que se comienza a generar, y además el problema crecerá con el tiempo, dado las actuales proporciones de crecimiento.

Internet.

Internet es un conjunto descentralizado de redes de comunicación interconectadas que utilizan la familia de protocolos TCP/IP, garantizando que las redes físicas heterogéneas que la componen funcionen como una red lógica única de alcance mundial, (Castells, 2009). Su origen se remonta a 1969, momento en el que se estableció la primera conexión de computadoras, conocida como ARPANET, entre tres universidades en California y una en Utah en Estados Unidos

Uno de los servicios que más éxito ha tenido en Internet ha sido la World Wide web (WWW, o "la Web"), hasta tal punto que es habitual la confusión entre ambos términos. La WWW es un conjunto de protocolos que permite, de forma sencilla, la consulta remota de archivos de hipertexto. Ésta fue un desarrollo posterior 1990 y utiliza Internet como medio de transmisión. (Carral, 1998)

Existen otros servicios y protocolos en Internet: el envío de correo electrónico (SMTP), la transmisión de archivos (FTP y P2P), las conversaciones en línea (IRC), la mensajería instantánea y presencia, la transmisión de contenido y comunicación multimedia -telefonía (VoIP), televisión (IPTV)-, los boletines electrónicos (NNTP), el acceso remoto a otros dispositivos (SSH y Telnet) o los juegos en línea. (Noreña, 2013)

Las exigencias de investigadores y laboratorios propiciaron el nacimiento de ARPANet (Advanced Research Projects Agency Network o Red de la Agencia para los Proyectos de Investigación Avanzada) de los Estados Unidos; legó el trazado de una red inicial de comunicaciones de alta velocidad a la cual fueron integrándose otras instituciones gubernamentales y redes académicas durante los años 70. (Carral, 1998)

Investigadores, científicos, profesores y estudiantes se beneficiaron de la comunicación con otras instituciones y colegas. De igual manera, disfrutaron de la nueva habilidad para publicar y hacer disponible a otros la información generada en sus actividades. La primera red interconectada nació el 21 de noviembre de 1969, cuando se crea el primer enlace entre La Universidad de California, Los Ángeles (UCLA) y Stanford por medio de la línea telefónica conmutada, y gracias a los trabajos y estudios anteriores de varios científicos y organizaciones.

En 1972 se realizó la primera demostración pública de ARPANET, una red de comunicaciones que funcionaba de forma distribuida sobre la red telefónica conmutada. El éxito de esta nueva arquitectura sirvió para que, en 1973, se iniciara un programa de investigación sobre posibles técnicas para interconectar redes (orientadas al tráfico de paquetes) de distintas clases. Para este fin, desarrollaron nuevos protocolos de comunicación que permitiesen este intercambio de información entre las computadoras conectadas. De la filosofía del proyecto surgió el nombre de "Internet", que se aplicó al sistema de redes interconectadas mediante los protocolos TCP e IP.

A inicios de los 90, con la introducción de nuevas facilidades de interconexión y herramientas gráficas simples para el uso de la red, se inició el auge que actualmente le conocemos al Internet. Este crecimiento masivo trajo consigo el surgimiento de un nuevo perfil de usuarios, en su mayoría de personas comunes no ligadas a los sectores académicos, científicos y gubernamentales.

Internet alcanzó los mil cien millones de usuarios en el 2006 y se prevé que en el 2013 puedan existir 2000 millones de navegantes en la Red. El resultado de todo esto es lo que experimentamos hoy en día: la transformación de lo que fue una enorme red de comunicaciones para uso gubernamental, planificada y construida con fondos estatales, que ha evolucionado en una diversa red de redes privadas interconectadas entre sí. Actualmente la red experimenta cada día la integración de nuevas redes y usuarios, extendiendo su amplitud y dominio, al tiempo que surgen nuevos mercados, tecnologías, instituciones y empresas que aprovechan este nuevo medio, cuyo potencial apenas comenzamos a descubrir.

Aplicación web

En la ingeniería de software se denomina aplicación web a aquellas herramientas que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una intranet mediante un navegador. En otras palabras, es una aplicación software que se codifica en un lenguaje soportado por los navegadores web en la que se confía la ejecución al navegador. (Mora, 2002)

Las aplicaciones web son populares debido a lo práctico del navegador web como cliente ligero, a la independencia del sistema operativo, así como a la facilidad

para actualizar y mantener aplicaciones web sin distribuir e instalar software a miles de usuarios potenciales. Existen aplicaciones como los webmails, wikis, weblogs, tiendas en línea y la propia Wikipedia que son ejemplos bien conocidos de aplicaciones web. (Mora, 2002)

1.5 Gestión de la información sindical en la Uniss.

1.5.1 La Universidad “José Martí Pérez” de Sancti Spíritus. Caracterización

La provincia de Sancti Spíritus está situada geográficamente en el centro este del archipiélago cubano. Los estudios de nivel superior en el territorio tienen como antecedentes particulares sus inicios a partir de la Filial Universitaria, que como parte de la Universidad Marta Abreu de Las Villas (UCLV) propició a partir de 1976 la formación en cursos para trabajadores de licenciados en la rama económica e ingenieros agrónomos. En una segunda etapa y a partir de 1994, la Sede Universitaria de Sancti Spíritus (SUSS) continuó siendo parte de la estructura de la UCLV.

En el año 2003 la SUSS se transforma en el Centro Universitario de Sancti Spíritus, (CUSS) ya desvinculado de la estructura orgánica de la UCLV y con personalidad jurídica propia. Este período se extendió hasta el 2010 cuando se constituyó la Universidad de Sancti Spíritus “José Martí Pérez” (Uniss).

Esta institución desarrolla quince carreras universitarias en tres tipos de cursos. Entre ellas destaca en el curso diurno la Ingeniería Informática, iniciada en el año 2005. Respondía esta decisión a una necesidad y demanda de la provincia de Sancti Spíritus y constituyó a su vez una respuesta al avance de las tecnologías de la información en función del desarrollo económico-social del territorio.

Este centro de educación manifestaba un limitado desarrollo informático durante la década de los ochenta. En 1989 la institución poseía 6 computadoras personales, siendo la más moderna de ellas una 486. En 1990 se conformó el primer laboratorio de computación con los nuevos equipos recibidos y en el año 1997 se montó el segundo mucho más moderno y conectado en red a través de servidores. Cerca de quince computadoras ofrecieron servicios a profesores y estudiantes en un local climatizado (actual laboratorio de idiomas).

A partir de ese momento se ha desarrollado un progresivo ascenso del uso de la informática en la institución. En estos momentos la red Uniss posee 704 computadoras distribuidas en sus cinco facultades, siete filiales universitarias municipales y en otras áreas docentes, investigativas y de apoyo a los procesos universitarios.

La universalización de los estudios universitarios constituyó a partir del 2001 un redimensionamiento de la misión de esta institución universitaria. La estructura actual para la gestión de la red informática Uniss se caracteriza por la existencia de un grupo de administración. Los diez servidores que se utilizan en estos momentos son computadoras de mesa con prestaciones que les permiten dar respuesta a la conexión con Internet, ftp, mensajería electrónica, la conexión con los municipios y otras.

En los momentos actuales persisten y surgen nuevas necesidades vinculadas con el desarrollo de la comunicación entre los actores de los procesos universitarios. Existe una nueva etapa en la Educación Superior, en la institución y en el territorio; en este sentido las principales dificultades quedan resumidas a continuación:

- Necesidad de una política de ampliación y mejoramiento de la red y de sus posibilidades técnicas internas.
- Lograr un mejor ancho de banda e interconectividad con toda la infraestructura física de la Uniss.
- La comunidad universitaria debe desarrollar una cultura organizacional que se ajuste a las exigencias de una sociedad informatizada e identificada plenamente con los principios y retos del proyecto social cubano. Debe ser reflejo de su compromiso e identidad con la universidad.
- .La red Uniss se encuentra mal diseñada y su explotación no está al nivel de las demandas de su comunidad.
- La relativamente joven provincia de Sancti Spíritus heredó una insuficiente infraestructura tecnológica, arquitectónica y de recursos humanos, lo que se refleja en las posibilidades para promover el desarrollo informático, lo que

implica en lo cultural, tecnológico, económico y ambiental hasta la diversidad del municipio.

- Existe además una demanda potencial no satisfecha por las organizaciones y grupos, acerca de información ágil, dinámica y segura que no siempre se le logra dar respuesta.
- La gestión que desarrolla el BUS en la Uniss debe ser extendida a todos sus miembros a través de nuevas formas y vías que lo faciliten.
- No existe conocimiento exacto de las necesidades y posibilidades que puede aportar el uso de la informatización en los resultados de la actividad sindical en la Uniss. Esto provoca el no uso correcto de las plataformas informáticas en este sentido
- El desconocimiento de las potencialidades informáticas que posee la universidad y el BUS desfavorece el compromiso de los propios actores involucrados en la búsqueda del cambio, lo que lleva a la poca participación en tales acciones y al no enriquecimiento de la cultura de mejoramiento mediante el uso de la informática.
- El desarrollo universitario y de la carrera de ingeniería informática debe contemplar y asumir de forma activa acciones que den respuesta a las necesidades de los diferentes niveles organizativos en la institución.
- El BUS de la universidad en Sancti Spíritus necesita revalorar su actual estrategia de comunicación con sus afiliados, a partir de las nuevas concepciones vinculadas con la informática, sus necesidades y potencialidades.

1.5.2 Organización sindical en la Uniss.

El Sindicato de los trabajadores de la educación y en particular los profesores y trabajadores universitarios están agrupados en la universidad en secciones sindicales. Su afiliación y trabajo permite la unidad de la clase obrera.

Desde sus inicios en 1976, la Uniss y su entonces filial universitaria contó con una sesión de base sindical con un número muy reducido de trabajados. Esta

estructura se subordinaba al Sindicato municipal de Sancti Spíritus.No es hasta el año 2001 que se crea el Buró Universitario Sindical, BUS, en la universidad. En estos momentos y desde el año 1989 existen en esta institución dos organizaciones de masas: la federación estudiantil universitaria (FEU) y el sindicato, agrupando cada una de ellas a los estudiantes y trabajadores.

Como parte de la estructura sindical, el buró universitario sindical, BUS, representa a la organización dentro de la estructura central de la organización a nivel de la universidad. Se constituye a partir de la selección de representantes elegidos en asambleas de trabajadores y en su membrecía se identifican a líderes y profesores destacados, los que han laborado por la unidad de la clase trabajadora, los derechos laborales, la búsqueda de espacios que faciliten la atención a los trabajadores, la satisfacción de sus necesidades y la defensa de los principios de la revolución.

El BUS tiene entre sus objetivos más importantes organizar, conducir y controlar el trabajo de las trece secciones sindicales que existan en la universidad. Debe trabajar en la diversidad que representa cada área de la institución, integrando cada una de ellas en función de los principios establecidos.

En su labor de búsqueda de la unidad entre los trabajadores ha sido necesario encontrar nuevos métodos y estilos de trabajo. Durante muchos años el BUS ha utilizado los murales, mítines, matutinos, la radio, la prensa, el teléfono, la propaganda gráfica y otros medios en un entorno que se limitaba a solo los dos edificios que constituyeron la filial, la sede y el centro universitario.

En la actualidad la descentralización de la Uniss en cinco edificios localizados en la ciudad capital de la provincia de Sancti Spíritus exige encontrar técnicas que garanticen la comunicación entre sus miembros o afiliados, los que suman setecientos tres en estos momentos.

1.5.3 Gestión de la información sindical en la Uniss.

En la Uniss el trabajo sindical se desarrolla en la sede central mediante el BUS y sus trece sesiones sindicales, las que agrupan a los trabajadores del municipio cabecera. Se debe destacar que las secciones sindicales de los municipios y la correspondiente a la Facultad de Montaña del Escambray en Topes de Collantes

no tienen subordinación directa a la estructura sindical mencionada, por lo que se vinculan a los niveles de dirección de esta organización en sus respectivos municipios.

La organización del trabajo sindical tiene sus propias particularidades en la Uniss a partir de la descentralización de las secciones sindicales. Estas últimas generan información que es necesaria para sus propios afiliados, los que se identifican por tareas y perfiles laborales comunes. Del mismo modo deben recibir o emitir información desde o hacia el resto de la comunidad universitaria.

La información generada por los integrantes de la organización de masas debe tener visibilidad, por lo que murales, matutinos y reuniones constituyen los espacios más utilizados. La Uniss posee un estimado de setecientas computadoras y una parte de ellas están interconectadas. Esta oportunidad puede ser aprovechada en mayor escala, ya que solo el correo electrónico es utilizado en alguna medida para el trabajo sindical o la comunicación entre afiliados.

En atención a la cultura de trabajo informática de la institución el BUS se ha propuesto encontrar otras vías que sean interesantes y amenas para los trabajadores; esto facilitará utilizar de forma paralela los recursos informáticos de la institución, ampliar el espectro para la búsqueda de información, identificar bases de datos y ofrecer visibilidad a los resultados.

1.6 Metodologías utilizadas para el diseño.

A fin de lograr un mejor desempeño y calidad para desarrollar un producto informático, se hace necesario la consulta de las nuevas tendencias y el uso de una metodología que indique los pasos correctos a seguir. El desarrollo del software propuesto, exige búsqueda, estudio y análisis de diversas tecnologías, lenguajes, gestores de base de datos y herramientas de desarrollo que hagan posible su realización y el cumplimiento de todas sus funcionalidades.

1.6.1 Metodología RUP.

El Proceso Unificado de Rational, (Rational Unified Process, de ahí las siglas RUP), es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas de software, con diferentes áreas de aplicación, de

organizaciones, niveles de aptitud y tamaños de proyectos. (Rumbaugh, Booch, & Jacobson, 2006)

Es un proceso basado en componentes y utiliza el Lenguaje Unificado de Modelado (UML) para preparar todos los esquemas de un sistema software. No obstante, los verdaderos aspectos definitorios de RUP se resumen en tres frases clave: está dirigido por casos de uso, centrado en la arquitectura y es iterativo e incremental. (Rumbaugh, Booch, & Jacobson, 2006)

1.6.2 Lenguaje Unificado de Modelado (UML).

El Lenguaje Unificado de Modelado (Unified Modeling Language, UML) es un lenguaje estándar para escribir planos de software, incluyendo aspectos conceptuales tales como procesos de negocio y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables. Puede utilizarse para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra una gran cantidad de software. (Rumbaugh, Booch, & Jacobson, 2006)

1.7 Tecnologías y Herramientas actuales para la elaboración de aplicaciones web.

1.7.1 Arquitectura Cliente/Servidor.

IBM define al modelo Cliente/Servidor de la siguiente manera "Es la tecnología que proporciona al usuario final el acceso transparente a las aplicaciones, datos, servicios de cómputo o cualquier otro recurso del grupo de trabajo y/o, a través de la organización, en múltiples plataformas. El modelo soporta un medio ambiente distribuido en el cual los requerimientos de servicio hechos por estaciones de trabajo inteligentes o "clientes", resultan en un trabajo realizado por otros computadores llamados servidores" La arquitectura tradicional de cliente/servidor también es conocida como arquitectura de dos capas. La cual facilita la integración entre sistemas diferentes y comparte información. Además de permitir un rápido mantenimiento y desarrollo de aplicaciones, contribuye a proporcionar, a los diferentes departamentos de una organización, soluciones locales, pero permitiendo la integración de la información relevante a nivel global.

1.7.2 Patrón de diseño Modelo-Vista-Controlador

En el diseño de aplicaciones con sofisticadas interfaces se utiliza el patrón de diseño Modelo-Vista-Controlador (MVC). Pues la lógica de una interfaz de usuario cambia con más frecuencia que los almacenes de datos y la lógica del negocio. Por lo que si se realiza un diseño que mezcle los componentes de interfaz y de negocio, entonces la consecuencia será que, cuando se necesite cambiar la interfaz, se tendrán que modificar los componentes de negocio, siendo esto trabajoso y engorroso, además de provocar posibles errores en el sistema. Por lo que se trata de realizar un diseño que desacople la vista del modelo, con la finalidad de mejorar la reusabilidad. De esta forma las modificaciones en las vistas impactan en menor medida en la lógica de negocio. (Lago, 2007)

Elementos del patrón:

Vista (View): es lo que utilizan los usuarios para interactuar con la aplicación (los gestores de plantillas pertenecen a esta capa).

Modelado (Model): La capa del modelo define la lógica de negocio (la base de datos pertenece a esta capa).

Controlador (Controller): es un bloque de código que realiza llamadas al modelo para obtener los datos y se los pasa a la vista para que los muestre al usuario.

1.7.3 Frameworks

Un framework, en el desarrollo de software es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado. Provee una estructura y una metodología de trabajo. (Gutiérrez, 2009)

Un framework representa una arquitectura de software que modela las relaciones generales de las entidades del dominio. Provee una estructura y una metodología de trabajo la cual extiende o utiliza las aplicaciones del dominio. Es decir, un framework está específicamente orientado a resolver o extender una solución general de un dominio de problemas. (Tedei, 2009)

Symfony 2

Inicialmente la idea de Fabien Potencier era juntar varios frameworks existentes y empezó incluyendo Propel como ORM y Ruby on Rails para las plantillas. Symfony 2 es la segunda versión de Symfony.

Desarrollado en PHP(siglas que originalmente significaban Personal Home Page, pero actualmente significan Hypertext Preprocessor) 5.3, es compatible con Windows, Unix o Linux. Acepta Gestores de base de datos, como MySQL, PostgreSQL, Oracle o Microsoft SQL Server. Hoy día también admite e integra herramientas para usar Propel o Doctrine como ORMs aunque cada vez más enfocado a Doctrine 2

Características básicas (Eguiluz, 2012):

- Fácil de instalar y configurar en la mayoría de plataformas, aunque es preferible para el desarrollo de grandes aplicaciones Web que para pequeños proyectos. No puedo más que asentir pero como explico anteriormente la curva de aprendizaje del uso del framework no es pronunciada pero una vez aprendidas cosas puede ser una herramienta muy potente y agilizar mucho el desarrollo.
- Es independiente del SGBD, se pueden usar varios distintos como MySQL, PostgreSQL, Oracle o Microsoft SQL Server.
- Sigue el MVC y por lo tanto separa las lógicas del sistema consecuentemente a lo explicado en el apartado 4.2.
- Basado en la premisa convenir y no configurar. (CoC - Convención sobre Configuración). Se acuerda una configuración pre-establecida que permite al programador ponerse a trabajar sin tener que preocuparse por ello.
- Usa PHP con OOP (Programación Orientada a Objetos) y requiere PHP 5 aunque es altamente recomendable que sea la versión 5.3 o superior.
- Sigue la mayoría de las mejores prácticas y patrones de diseño para la web.
- Versión estable.
- Fácil de extender permitiendo la integración con librerías de terceros.

- Características de automatización, Symfony automatiza la mayoría de elementos comunes de los proyectos web, como por ejemplo:
- La capa de internacionalización que incluye Symfony permite la traducción de los datos y de la interfaz, así como la adaptación local de los contenidos.
- La capa de presentación utiliza plantillas y layouts que pueden ser creados por diseñadores HTML sin ningún tipo de conocimiento del framework. Los helpers incluidos permiten minimizar el código utilizado en la presentación, ya que encapsulan grandes bloques de código en llamadas simples a funciones.
- Los formularios incluyen validación y relleno automático de datos, lo que asegura la obtención de datos correctos y mejora la experiencia de usuario.
- Los datos incluyen mecanismos de escape que permiten una mejor protección contra los ataques producidos por datos corruptos.
- La gestión de la caché reduce el ancho de banda utilizado y la carga del servidor.
- La autenticación y la gestión de credenciales simplifican la creación de secciones restringidas y la gestión de la seguridad de usuario.
- El sistema de enrutamiento y las URL limpias permiten considerar a las direcciones de las páginas como parte de la interfaz, además de estar optimizadas para los buscadores.
- Los listados son más fáciles de utilizar debido a la paginación automatizada, el filtrado y la ordenación de datos.

Symfony 2 además tiene una potente línea de comandos que facilitan generación de código, lo cual contribuye a ahorrar tiempo de trabajo (Eguiluz, 2012).

1.7.4 Tecnologías del lado del cliente

Las tecnologías del lado del cliente están insertadas en la página HTML del cliente y son interpretadas y ejecutadas por el navegador. Es decir, que su correcta funcionalidad depende del soporte de la versión del navegador a ser utilizado por el usuario visitante. (Rodríguez & Alejo, 2006)

Twig

Twig es un sistema de plantillas para la programación PHP que viene incluido en Symfony2, es open source.

En realidad Symfony2 también soporta plantillas PHP pero yo me decante por usar las plantillas twig porque son altamente recomendables al ser más sencillas, seguras y flexibles que las plantillas PHP:

- Symfony2 Traduce las plantillas twig a código PHP optimizado reduciendo en mucho la cantidad de código que generaría usando plantillas PHP.
- Twig tiene un sistema de evaluación de código que permite evitar que los usuarios puedan modificar el diseño de las plantillas.
- Twig permite definir etiquetas, funciones y servicios.

Una de las características más interesantes de Twig es que permite herencia múltiple y eso me ha permitido desarrollar un sistema de templates de tal forma que cada pantallazo de la aplicación puede tener hasta 6 plantillas twig detrás (Eguiluz, 2012).

HTML

El HTML (HyperText Markup Language) es un formato de datos para crear documentos de hipertexto que puedan ser entendidos desde computadoras con diferentes sistemas operativos. Es un sencillo sistema de marcas, que insertadas en un texto, permiten el enlace con otras páginas o recursos (imágenes, sonidos, etc.) y establecen ciertos atributos del documento (dimensiones, alineamiento del texto, estilo, etc.). La flexibilidad que posee es tal que sus características pueden aprovecharse para generar documentos multimedia, aunque no vayan a colocarse en un servidor. (Vidal, 2006) HTML es un subconjunto de la especificación SGML (Standard Generalized Markup Language) que es mucho más amplia.

Los documentos HTML son SGML con una semántica determinada que es apropiada para representar información de diversos tipos en un único documento. En definitiva, HTML es un lenguaje de marcas que sigue una definición formal de acuerdo con un estándar. Su potencia se basa en su independencia del sistema operativo y que identifica cada parte de un documento por la finalidad que tiene.

(Vidal, 2006) HTML no tiene la complejidad de un lenguaje de programación, pues es bastante sencillo de utilizar y no precisa de ningún compilador. Más bien, es semejante a un documento de texto. Todo documento HTML es almacenado como texto normal en formato ASCII, de manera que cualquier procesador o editor de textos puede ser utilizado para su creación. El navegador es quien se encarga de leer ese documento de texto e interpretar las etiquetas HTML que aparecen en él. (Vidal, 2006)

Ventajas: (Cadenas, 2010)

- Sencillo, que permite describir hipertexto.
- Texto presentado de forma estructurada y agradable.
- No necesita de grandes conocimientos cuando se cuenta con un editor de páginas Web o WYSIWYG.
- Despliegue rápido.
- Lenguaje de fácil aprendizaje.
- Lo admiten todos los exploradores.

Desventajas: (Cadenas, 2010)

- Lenguaje estático.
- La interpretación de cada navegador puede ser diferente.
- Guarda muchas etiquetas que pueden convertirse en “basura” y dificultan la corrección.
- El diseño es más lento.
- Las etiquetas son muy limitadas.

JavaScript

Permite ejecutar secuencias de comandos en el mismo navegador del usuario. Con JavaScript se pueden realizar cálculos rápidos y complejos y hasta controlar la mayor parte de los elementos del navegador. También se pueden ejecutar acciones como abrir ventanas, verificar formularios antes de enviarlos, convertir divisas y crear calendarios. (Ruiz, 2006)

(Álvarez, 2009) afirma que es un lenguaje interpretado, no requiere compilación. Utilizado principalmente en páginas web. Es similar a Java, aunque no es un lenguaje orientado a objetos, el mismo no dispone de herencias. La mayoría de los navegadores en sus últimas versiones interpretan código Java Script. El código JavaScript puede ser integrado dentro de las páginas web. Los script tienen capacidades limitadas, por razones de seguridad, el código visible por cualquier usuario y el código deben descargarse completamente.

Se utiliza embebido en el código HTML, entre las etiquetas <script> y </script>, de manera que solo sirve para incluirse en documentos HTML y fuera de ellos no tiene ninguna vigencia, o sea que no crea aplicaciones autónomas. - Puede ejecutarse en el cliente evitando la necesidad de realizar peticiones continuamente al servidor, por lo que descarga el trabajo de este y hace que la navegación sea más rápida.

Ventajas: (Cadenas, 2010)

- Lenguaje de scripting seguro y fiable.
- Los script tienen capacidades limitadas, por razones de seguridad.
- El código Java Script se ejecuta en el cliente.

Desventajas: (Cadenas, 2010)

- Código visible por cualquier usuario.
- El código debe descargarse completamente.

CSS

CSS (Cascade Style Sheets) es una tecnología que permite controlar la presentación de los documentos en la Web. Las Hojas de Estilo en Cascada o CSS constituyen un lenguaje sencillo que complementa el de HTML, suponiendo un apoyo fundamental a la hora de diseñar páginas Web; permiten una mayor precisión en el ajuste de los elementos de diseño. (Vedora, 2006)

Esta técnica consiste en separar el diseño del contenido; las indicaciones para conformar el diseño se agrupan en una hoja de estilo o archivo fuera del contenido del documento de la página HTML. Lo que hace fundamentalmente el código de

las hojas de estilos es transformar las etiquetas del lenguaje HTML y conformarlas a las características que se quiera darle; con este código se pueden crear etiquetas nuevas, que se introducen dentro del documento. Una de las ventajas de las hojas de estilos es que se pueden modificar algunas características de todos los documentos de un sitio Web desde un archivo, sin tener que modificarlas en cada uno de los documentos. (Vedora, 2006)

Ventajas. Varios autores se refieren a las mismas. (Cadenas, 2010) señala las siguientes:

- Es la mejor forma de separar los contenidos y su presentación, es imprescindible para crear páginas Web complejas.
- Reduce la complejidad de su mantenimiento.
- Permite visualizar el mismo documento en infinidad de dispositivos diferentes.
- Permite especificar la apariencia de todo el sitio Web en un único lugar, los estilos para fuentes, tamaños de la fuente, espaciado de letras, colores, y márgenes. Un cambio en un fichero genera cambios en todos los ficheros.

Desventajas: (Cadenas, 2010)

- No todos los navegadores soportan CSS.

1.7.5Tecnologías del lado del servidor

Las tecnologías del lado del servidor pueden o no estar insertadas dentro de la página HTML. A diferencia de las tecnologías del lado del cliente, estas no dependen del navegador ya que son interpretadas y ejecutadas por el servidor. Por ejemplo, si se utiliza PHP en un sitio determinado se necesita que el servidor donde esté alojado el mismo, tenga instalado PHP. (Rodríguez & Alejo, 2006)

Doctrine

Doctrine es un mapeador de objetos-relacional (ORM) escrito en PHP que proporciona una capa de persistencia para objetos PHP. Es una capa de abstracción que se sitúa justo encima de un Sistema Gestor de Base de Datos. (Project Team, 2012)

Una característica de Doctrine es el bajo nivel de configuración que necesita para empezar un proyecto. Doctrine puede generar clases a partir de una base de datos existente y después el programador puede especificar relaciones y añadir funcionalidad extra a las clases autogeneradas. No es necesario generar o mantener complejos esquemas XML de base de datos como en otros frameworks.

Otra característica de Doctrine es la posibilidad de escribir consultas de base de datos utilizando un dialecto de SQL denominado **DQL** (Doctrine Query Language) que está inspirado en Hibernate (Java). También se añaden las siguientes: (Project Team, 2012)

- Soporte para datos jerárquicos;
- Soporte para hooks (métodos que pueden validar o modificar las escrituras y lecturas de la base de datos) y eventos para manejar la lógica de negocio relacionada;
- Herencia;
- Un framework de caché que utiliza diversos motores como memcached, SQLite o APC;
- Transacciones ACID;
- Diversos comportamientos del modelo (conjuntos anidados, internacionalización, log, índice de búsqueda);
- Una función "compilar" que combina varios archivos PHP del framework en uno solo para evitar el descenso de rendimiento que provoca incluir varios archivos PHP.

PHP

Luego de valorar las diferentes tecnologías se decide utilizar el PHP ya que en la UNISS se asumiría mejor por ser un lenguaje impartido en clases y por tanto se tienen conocimientos previos. Es rápido, gratuito y predomina en la mayoría de las aplicaciones existentes actualmente en la red porque está soportado en la mayoría de las plataformas de Sistemas Operativos. Existe suficiente documentación sobre el mismo. El PHP no tiene costo oculto, o sea que cuando se adquiere incluye un

sin número de bibliotecas que proporcionan el soporte para la mayoría de las aplicaciones Web, por ejemplo e-mail, generación de ficheros PDF y otros. En caso de que no se tengan las bibliotecas estas se pueden adquirir gratis en Internet. Existen muchos lenguajes parecidos al PHP en cuanto a la forma de utilización, pero este es más rápido, gratuito y multiplataforma.

PHP es un lenguaje de programación el cual se ejecuta en los servidores Web y que permite crear contenido dinámico en las páginas HTML, con un lenguaje propietario derivado del Perl. Dispone de múltiples herramientas que permiten acceder a bases de datos de forma sencilla, por lo que es ideal para crear aplicaciones para Internet. Es multiplataforma, funciona tanto para Linux (con Apache) como para Windows (con Microsoft Internet Information Server) de forma que el código que se haya creado para una de ellas no tiene por qué modificarse al pasar a la otra. PHP es un lenguaje de script ejecutado del lado del servidor específicamente para programación web.

PHP un lenguaje que se ejecuta en el servidor no es necesario que el navegador lo soporte, es independiente del navegador, pero sin embargo para que sus páginas PHP funcionen, el servidor donde están alojadas debe soportar PHP.

Ventajas: (Cadenas, 2010)

- Muy fácil de aprender.
- Se caracteriza por ser un lenguaje muy rápido.
- Soporta en cierta medida la orientación a objeto. Clases y herencia.
- Es un lenguaje multiplataforma: Linux, Windows, entre otros.
- Capacidad de conexión con la mayoría de los manejadores de base de datos: MySQL, PostgreSQL, Oracle, MS SQL Server, entre otras.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Incluye gran cantidad de funciones.
- No requiere definición de tipos de variables ni manejo detallado del bajo nivel.

Desventajas: (Cadenas, 2010)

- Se necesita instalar un servidor Web.
- 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.

1.7.6 Herramientas de desarrollo.

NetBeans

NetBeans es un proyecto de código abierto de gran éxito con una gran base de usuarios, una comunidad en constante crecimiento, y con cerca de 100 socios en todo el mundo. Sun Microsystems fundó el proyecto de código abierto NetBeans en junio de 2000 y continúa siendo el patrocinador principal de los proyectos. (Netbeans, 2009)

El IDE NetBeans es un producto libre y gratuito sin restricciones de uso y de amplias comodidades para el programador. (Netbeans, 2009) Para la realización de este proyecto se utilizó el IDE 7.2.1 de NetBeans, una herramienta para programadores pensada para escribir, compilar, depurar y ejecutar programas.

Symfony 2 inicialmente no estaba integrado en ningún IDE pero debido a que en Netbeans era el IDE que había hecho esfuerzos por integrar la versión anterior, parecía que sería en un futuro el IDE que apostaría por integrar la versión 2.0.

Por ello se ha decidido usar Netbeans para la programación sobre Windows sin tener ningún tipo de integración ya que no existía ninguna aplicación que tuviese integración con Symfony2.

Macromedia Dreamweaver.

(Álvarez, 2009) se refiere a esta herramienta como un editor HTML profesional para diseñar, codificar y desarrollar sitios, páginas y aplicaciones web. Tanto si desea controlar manualmente el código HTML como si prefiere trabajar en un entorno de edición visual, Dreamweaver le proporciona útiles herramientas que mejorarán su experiencia de creación Web. Es la herramienta de diseño de páginas web más avanzada, tal como se ha afirmado en muchos medios. Aunque sea un experto programador de HTML el usuario que lo maneje, siempre se

encontrará en este programa razones para utilizarlo, sobre todo en lo que a productividad se refiere.

EMS SQL Manager for MySQL

Es una herramienta de alto desempeño para administración y desarrollo en Servidor de Base de Datos MySQL. SQL Manager para MySQL trabaja con cualquier versión de MySQL desde 3.23 hasta 5.2 y soporta todas las últimas características del MySQL, incluyendo vistas, procedimientos almacenados y funciones, claves foráneas InnoDB y más. Ofrece una gran cantidad de herramientas poderosas a usuarios experimentados para satisfacer todas sus necesidades, además de tener una nueva interfaz gráfica de usuario de avanzada con un sistema asistente bastante descriptivo.

Rational Rose

Rational Rose Interprise Edition es una herramienta CASE desarrollada por Rational Corporation basada en el Lenguaje Unificado de Modelado (UML), que permite crear los diagramas que se van generando durante el proceso de Ingeniería en el Desarrollo del Software. Esta herramienta 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. Es la herramienta CASE que comercializan los desarrolladores de UML y que soporta de forma completa la especificación del UML. 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. (Alonso, 2006)

1.7.7 Sistemas gestores de base de datos (SGBD).

Una Base de Datos (BD) es un conjunto de datos interrelacionados, almacenados con carácter más o menos permanente en la computadora, puede ser considerada una colección de datos variables en el tiempo. (Mato García, 2006)

Un Sistema de Gestión de Base de Datos (SGBD) es el software que permite la utilización y la actualización de los datos almacenados en una o varias bases de datos desde diferentes puntos de vista y a la vez. Su objetivo fundamental es suministrar al usuario las herramientas que le permitan manipular, en términos abstractos, los datos, de forma que no le sea necesario conocer el modo de almacenamiento de los datos en la computadora, ni el método de acceso empleado.

Un SGBD cumple los objetivos de garantizar independencia de los datos y los programas de aplicación, minimizar las redundancias, integrar y sincronizar las bases de datos y los datos que estas contienen, al tiempo que los protege y facilita la manipulación de la información, mediante un control centralizado. La información es representada a través de tuplas, las cuales describen al fenómeno, proceso o ente de la realidad objetiva que se está analizando y se representan a través de tablas. (Alonso & Hernández, 2006)

En la actualidad existe un sinnúmero de SGBD, muchos de ellos muy potentes, entre los que se encuentran Microsoft SQL Server, Oracle, PostgreSQL, MySQL, entre otros. En esta investigación se hace especial alusión al SGBD MySQL por las características que exponen a continuación y que demuestran que puede resultar una buena elección a la hora de concebir la capa de datos de cualquier arquitectura que implemente el modelo de N capas.

MySQL

MySQL es el sistema gestor de bases de datos Open Source más popular en la comunidad de programadores. Puede ser descargado de Internet y usado de forma gratuita. Su código se puede estudiar y adecuar a cualquier requerimiento establecido. Es rápido, fiable, fácil de usar y surge para manipular bases de datos muy grandes con gran eficiencia. Es un sistema multiplataforma de base de datos, aspecto que le aporta la característica de ser fiable y veloz. Cuenta con un sistema

de privilegios y contraseñas muy seguro que permite la autenticación básica para el acceso al servidor; elemento muy importante que contribuye a garantizar en gran medida la seguridad de la aplicación. El lenguaje PHP posee un amplio conjunto de funciones definidas para el tratamiento de este gestor. Este SGBD es considerado (en su propia documentación así lo reseña) como la más rápida y robusta herramienta para la administración y gestión de bases de datos, tanto para volúmenes de datos grandes como pequeños.

Sus principales características son: (Vedora, 2006)

- Cuenta con tecnología InnoDB Engine, que proporciona transacciones, claves externas, actualización y borrado en cascada y bloqueo a nivel de fila, más rápido, con caché de consultas, mejoras en inserciones, búsqueda en índices compuestos y creación de índices sobre texto completo, un servidor embebido y compatibilidad con otras bases de datos.
- Es posible definir diversos tipos de columnas como enteros de 1, 2, 3, 4, y 8 bytes, coma flotante, doble precisión, carácter, fechas, enumerados, etc.
- El servidor soporta mensajes de error en distintas lenguas.
- Todas las columnas pueden tener valores por defecto. Admite registros de longitud fija y variable.
- Todas las claves viajan cifradas en la red. Cuenta con un sistema de contraseñas y privilegios muy flexibles y seguros.
- Puede trabajar en distintas plataformas y sistemas operativos. Implementa el multiprocesamiento, o sea, puede usar varias CPU si estas están disponibles.

Ventajas: (Cadenas, 2010)

- Está disponible para la mayoría de las plataformas de sistemas operativos.
- Su bajo consumo lo hace apto para ser ejecutado en una máquina con escasos recursos sin ningún problema.
- Gran velocidad y flexibilidad. Es muy rápido. Alto rendimiento. (Servidores de Base de Datos citado en Raphie Morgan, 2006)
- Elevada velocidad tanto al conectar con el servidor como al servir consultas y demás funciones.

- Excelentes utilidades de administración (backup, recuperación de errores, etc).
- Aunque se bloquee, no suele perder información ni corromper los datos.
- Excelente integración con PHP. No tiene límites en el tamaño de los registros.
- Significativo control de acceso, es decir, qué usuarios tienen acceso a que tablas y con qué permisos.
- MySQL se comporta mejor que otros motores de Base de Datos como SQL Server y Oracle a la hora de modificar o añadir campos a una tabla.
- Es multi-hilo, o sea, maneja muchas conexiones al mismo tiempo, donde cada una tiene su propio hilo, de modo que ningún hilo tiene que esperar por otro, a menos que uno esté modificando una tabla que otro quiera acceder.

Desventajas: (Cadenas, 2010)

- Un gran porcentaje de las utilidades de MySQL no están documentadas.
- No es intuitivo, como otros programas (ACCESS).
- No tiene integridad referencial, por lo que necesita ser programada.
- Al no utilizar la integridad referencial la velocidad de ejecución aumenta exponencialmente.

1.7.8 Servidores Web

Los servidores web son aquellos cuya tarea es alojar sitios y/o aplicaciones, las cuales son accedidas por los clientes utilizando un navegador que se comunica con el servidor utilizando el protocolo HTTP.

Servidor Apache

Apache: servidor web utilizado mundialmente. Por defecto lo traen instalado en todas las distribuciones Linux. También existe para otras plataformas incluso Windows. Su funcionamiento básico es ejecutando un proceso padre y tantos procesos hijos como peticiones reciba para atender a cada cliente. Hoy en día, el servidor Web Apache es el servidor más usado de Internet, con una utilización del 65% aproximadamente. Este servidor se sigue desarrollando “en Internet” como un proyecto de Software libre. Las principales metas de su diseño son: velocidad,

simplicidad, multiplataforma y facilidad del desarrollo distribuido. Es un servidor basado en procesos.

1.7.9 ¿Por qué utilizar Apache-PHP-MySQL?

La selección de estas tecnologías se basó principalmente en los criterios de rapidez, sencillez y rendimiento, unidos a otros dos mucho más importantes: la tecnología Open Source y la característica de integrarse para funcionar en cualquier plataforma. Estas características aportan un gran beneficio gracias al entorno de software libre de la actualidad y cumplen con una política trazada por el país de proyectarse con visión al futuro y comenzar a desarrollar este tipo de aplicaciones. Como Sistema Gestor de Base de Datos se seleccionó MySQL, por su velocidad y sencillez. La selección estuvo motivada además por la utilización de PHP, ya que esta tecnología exhibe un magnífico soporte para el trabajo con MySQL. Finalmente como servidor Web se determinó que la mejor opción era el Apache, por mostrar un excelente comportamiento y estabilidad. Este servidor es el más utilizado en el mundo por su magnífico desempeño, seguridad y las posibilidades de gestionar de una forma eficiente muchos de los aspectos que definen en gran medida la calidad de los servidores Web. La selección de Apache añade una integración perfecta con las tecnologías antes mencionadas. Apache-PHP-MySQL forman un en su conjunto una vía muy eficaz a la hora de tener en cuenta la implementación de Sitios Web de corto, mediano o gran alcance.

Conclusiones

En el capítulo que concluye se abordaron los fundamentos teóricos necesarios para la comprensión del tema que se trata; se describieron todos los conceptos asociados al problema. Se analizó además el objeto de estudio y la situación problemática. Se han justificado las razones por las cuales ha sido seleccionada como metodología a seguir en la documentación del software propuesto y en su proceso de desarrollo a RUP y como notación el Lenguaje de Modelado Unificado (UML), siendo el lenguaje de programación PHP y utilizando como gestor de base de datos MySQL.

CAPÍTULO II: Descripción de la Aplicación Web para facilitar la gestión de la información en el Buró Universitario Sindical de la Uniss.

Introducción

En este capítulo se realiza un estudio del modelo del negocio, este proceso permite una mejor comprensión de la problemática al conocer cuales son los actores del negocio, los trabajadores, los casos de uso; así como los diagramas de actividades y el modelo de objetos. Además en este capítulo se dejan plasmados los casos de uso del sistema, sus actores, diagramas de casos de uso del sistema así como la descripción de cada uno según lo propuesto por la metodología RUP.

2.1 Modelo del negocio.

El modelado de negocio es una técnica para comprender los procesos de negocio de la organización. Además de identificar los casos de uso y las entidades del negocio relevantes que el software debe soportar, de forma que se puede modelar solo lo necesario para que se comprenda el contexto. (Jacobson, Booch, & Rumbaugh, 2000)

El modelo del negocio está soportado por dos tipos de modelos de UML: modelo de casos de uso y modelo de objetos. (Jacobson, Booch, & Rumbaugh, 2000) Seguidamente se describe el proceso de negocio que se lleva a cabo en el BUS de la Universidad de Sancti Spíritus “José Martí Pérez”, mediante los artefactos propuestos por la metodología RUP y modelados por el lenguaje UML.

2.1.1 Identificación de los procesos del negocio

El Buró Universitario Sindical de la Universidad de Sancti Spíritus representa el máximo nivel jerárquico en la estructura de esta importante organización de masas que agrupa a los trabajadores docentes y no docentes de la institución.

El hecho de existir trece secciones sindicales en la Uniss permite que existan las condiciones reglamentadas por la organización para constituir una nueva estructura denominada Buró Universitario Sindical.

La misma se elige en las elecciones que convoca la organización con la participación de todos los trabajadores, quienes con su voto deciden la composición de su membresía.

El BUS define y ofrece al colectivo sindical de la universidad los principios, objetivos, metas y lineamientos fundamentales que rigen el desarrollo para el trabajo de las secciones sindicales, cumpliendo además con las políticas y las reglamentaciones establecidas por la organización y sus niveles de mando en el municipio, provincia y nación.

Para cumplir con los preceptos anteriores elabora la planificación del trabajo a partir de planes que contienen las tareas fundamentales del año y las correspondientes a cada mes. En ellas se trabaja por la unidad de los trabajadores, la defensa de sus intereses, y la creación de un clima laboral que estimule los mejores resultados y compulse hacia el perfeccionamiento de las misiones de los trabajadores en una sociedad socialista donde la justicia social ocupa un lugar importante.

Realiza encuentros mensuales ordinarios con la participación de todos sus miembros; están presentes además invitados que rinden cuentas de su gestión de trabajo, tales como secretarios de secciones sindicales, dirigentes administrativos y otros.

Del mismo modo moviliza a los trabajadores hacia tareas priorizadas por la organización, destacando la emulación, cotización, donaciones de sangre, trabajos voluntarios o productivos, actividades culturales, deportivas y recreativas, debates políticos, capacitación obrera y atención en general del trabajo de las secciones sindicales.

El BUS representa a todos los trabajadores universitarios y está presente en el consejo de dirección de la institución como miembro permanente. Allí trasmite las inquietudes y quejas en la búsqueda de la solución a los problemas o dificultades expresadas por los trabajadores.

2.2 Reglas del negocio

Las reglas del negocio describen políticas que deben cumplirse o condiciones que deben satisfacerse, por lo que regulan algún aspecto de negocio (Rumbaugh, Booch, & Jacobson, 2006).

Partiendo de lo planteado anteriormente se identificaron las siguientes reglas:

- El Personal del Sindicato y el usuario no pueden tener los mismos privilegios.
- Cada área de trabajo debe estar asociada a una Sección Sindical.
- Generar mensualmente una tabla de los destacados del mes.
- El integrante del Bus debe ser un trabajador de la Uniss.

2.3 Modelo de casos de uso del negocio

El modelo de Casos de Uso del Negocio es el encargado de describir los procesos de una empresa utilizando los casos de uso y los actores, en correspondencia, a su vez, con los procesos del negocio y los clientes (Jacobson, Booch, & Rumbaugh, 2000). Se define a través de: el diagrama de casos de uso, la descripción y el diagrama de actividades de los casos de uso.

2.3.1 Actores del Negocio

Un actor del Negocio es cualquier individuo, grupo, entidad, organización, máquina o sistema de información externos con los que el negocio interactúa. Lo que se modela como actor es el rol que se juega cuando se interactúa con el negocio para beneficiarse de sus resultados (Jacobson, Booch, & Rumbaugh, 2000)

El actor de nuestro negocio se presenta a continuación:

Actor del Negocio	Justificación
Trabajador de la Uniss	Es el interesado en recibir la información referente al Buró Universitario Sindical, el tique de la Cotización y el del Día de Haber.

Tabla 1: Actor del negocio

2.3.2 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 (González, 2005)

Los trabajadores de nuestro negocio se listan a continuación:

Trabajador	Justificación
Personal del Sindicato	Encargado de recopilar la Información referente al Buró Universitario Sindical.
Cotizador	Persona encargada de cobrar la cotización y el día de haber.

Tabla 2: Trabajadores del negocio

2.3.3 Diagramas de casos de uso del Negocio

Los diagramas de casos de uso se utilizan para especificar las funcionalidades y el comportamiento de un sistema mediante su interacción con los usuarios y otros sistemas. O sea es un diagrama que muestra la relación entre los actores y los casos de uso en un sistema (Jacobson, Booch, & Rumbaugh, 2000)

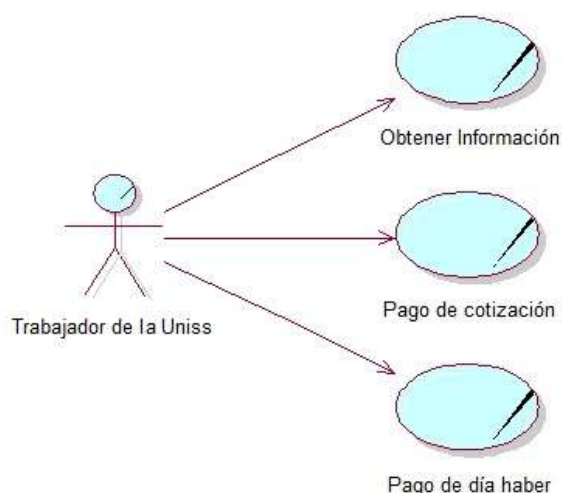


Figura 1: Diagrama de Caso de Uso del Negocio

2.3.4 Descripción de los casos de uso del negocio.

Caso de Uso del Negocio	Obtener Información
Actores	Trabajador de la Uniss
Propósito	Solicitar la información al Personal del Sindicato
<p>RESUMEN: El Caso de Uso comienza cuando el Trabajador de la Uniss necesita alguna información, este la solicita al Personal del Sindicato, que la recopila y luego se la entrega. El Caso de Uso termina cuando el Personal del Sindicato le entrega la información deseada al Trabajador de la Uniss.</p>	
Acción del Actor	Respuesta del proceso del Negocio
1. Solicita al Personal del Sindicato la información que necesita.	
	<p>2. El Personal del Sindicato recibe la solicitud.</p> <p>3. El Personal del Sindicato encuentra la información deseada, de lo contrario ir a Subsección1.</p> <p>4. El Personal del Sindicato elabora un informe con la información solicitada por el Trabajador de la Uniss y este es enviado.</p>
5. Recibe la información solicitada.	
Curso Alternativo de los eventos:	
Subsección 1	6. El Personal del Sindicato informa al Trabajador de la Uniss que esa información no la tienen.

Prioridad	Alta
Mejoras	Se accederá a la información de forma automática, utilizando los documentos previamente almacenados por el sistema.

Tabla 3: Descripción del Caso de Uso Obtener Información.

Caso de Uso del Negocio	Pagar de cotización
Actores	Trabajador de la Uniss
Propósito	Pagar cotización de determinado mes al Personal del Sindicato
RESUMEN: El Caso de Uso comienza cuando el Trabajador de la Uniss paga al encargado de cobrar la cotización en cada sección de un determinado mes, este lo recopila, lo anota en el registro de pago y el caso de uso termina cuando la persona recibe el comprobante de pago.	
Acción del Actor	Respuesta del proceso del Negocio
1. Paga al Personal del Sindicato la cotización de determinado mes.	
	<p>2. El Personal del Sindicato recibe el pago.</p> <p>3. El Personal del Sindicato lo anota en el registro de pago.</p> <p>4. El Personal del Sindicato elabora un comprobante para el Trabajador de la Uniss y se lo entrega.</p>

5. Recibe el comprobante.	
Prioridad	Alta
Mejoras	Se accederá a la información de forma automática, utilizando los documentos previamente almacenados por el sistema.

Tabla 4: Descripción del Caso de Uso Pagar de cotización.

Caso de Uso del Negocio	Pagar de día haber
Actores	Trabajador de la Uniss
Propósito	Pagar día haber en un determinado mes al Personal del Sindicato
RESUMEN: El Caso de Uso comienza cuando el Trabajador de la Uniss paga al encargado de cobrar el día haber en cada sección, este lo recopila, lo anota en el registro de pago y el caso de uso termina cuando la persona recibe el comprobante de pago.	
Acción del Actor	Respuesta del proceso del Negocio
1. Paga al Personal del Sindicato el día haber en un determinado mes.	
	<p>2. El Personal del Sindicato recibe el pago.</p> <p>3. El Personal del Sindicato lo anota en el registro de pago.</p> <p>4. El Personal del Sindicato elabora un comprobante para el Trabajador de la</p>

	Uniss y se lo entrega.
5. Recibe el comprobante.	
Prioridad	Alta
Mejoras	Se accederá a la información de forma automática, utilizando los documentos previamente almacenados por el sistema.

Tabla 5: Descripción del Caso de Uso Pagar de día haber.

2.3.5 Diagramas de actividades del negocio.

Un diagrama de actividades es el que muestra el flujo de actividad a actividad; los diagramas de actividad tratan la vista dinámica de un sistema. Un caso especial de diagrama de estados (aquellos diagramas que tratan la vista dinámica de un sistema) en el cual todos o casi todos los estados son estados de acción y en el cual todas o casi todas las transiciones son disparadas por la terminación de las acciones en los estados origen. (Rumbaugh, Booch, & Jacobson, 2006)

Los diagramas de actividades se listan a continuación:

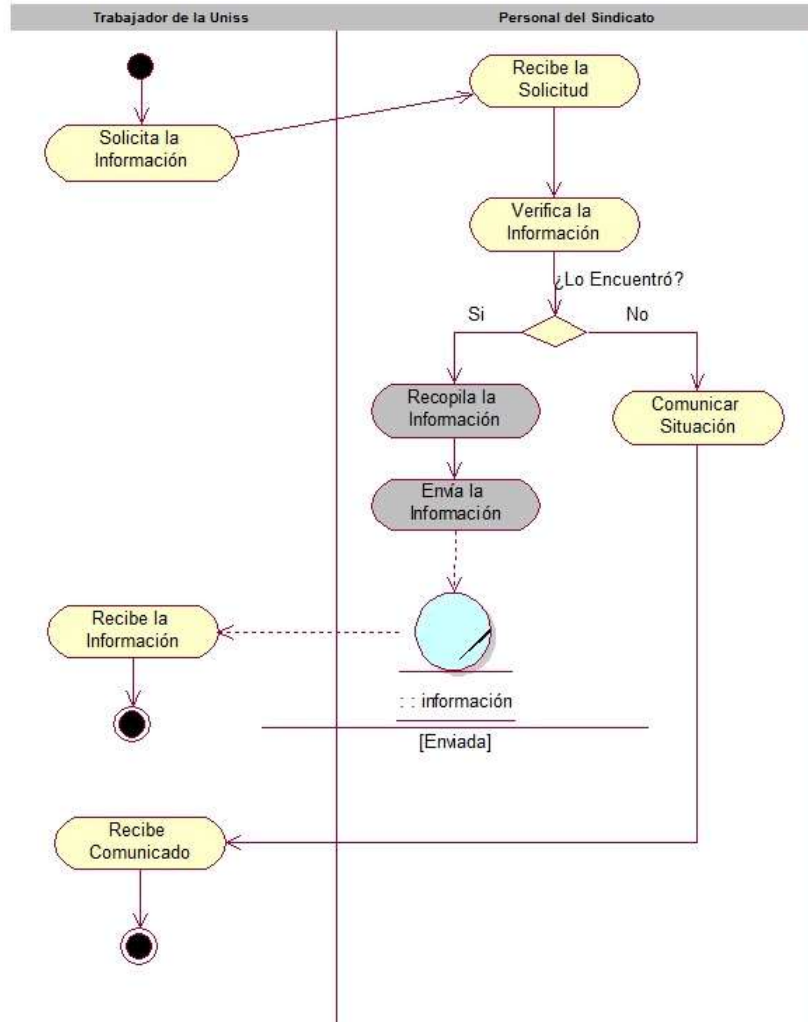


Figura 2: Diagrama de Actividad: Caso de uso Obtener Información.

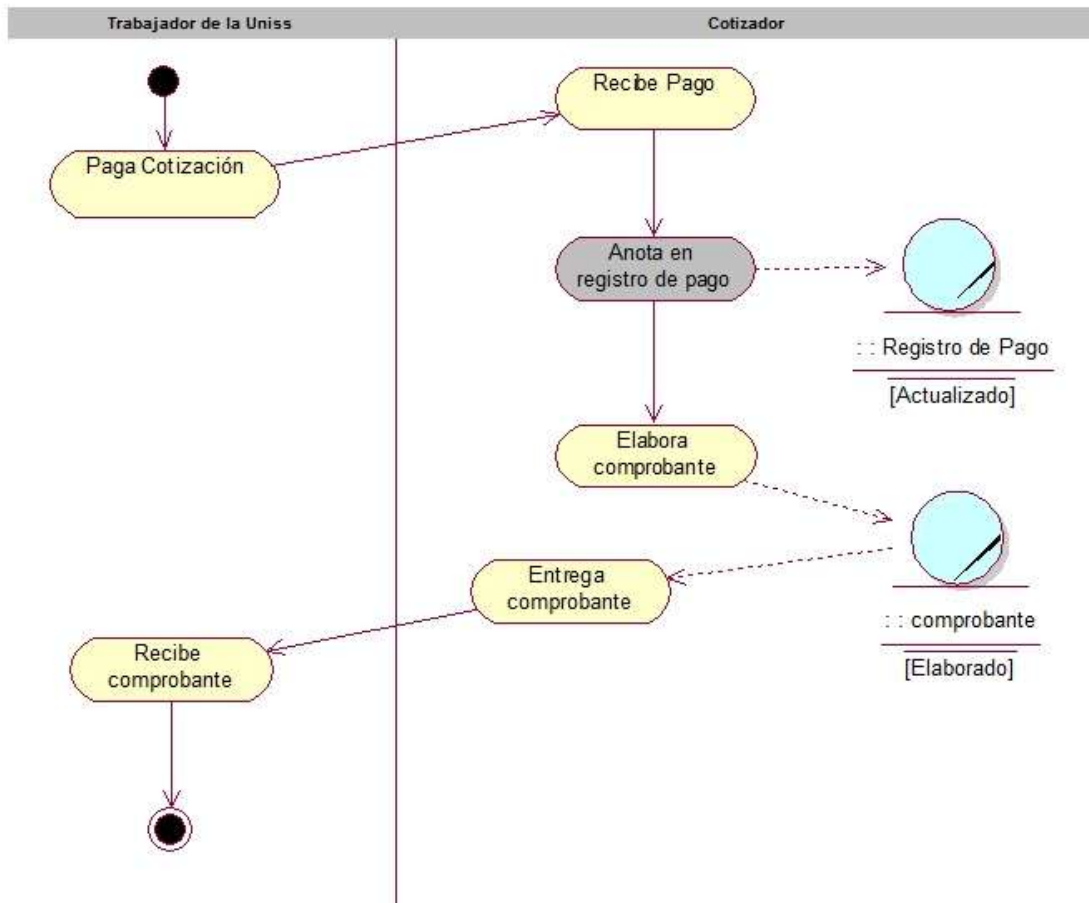


Figura 3: Diagrama de Actividad: Caso de uso Pagar Cotización.

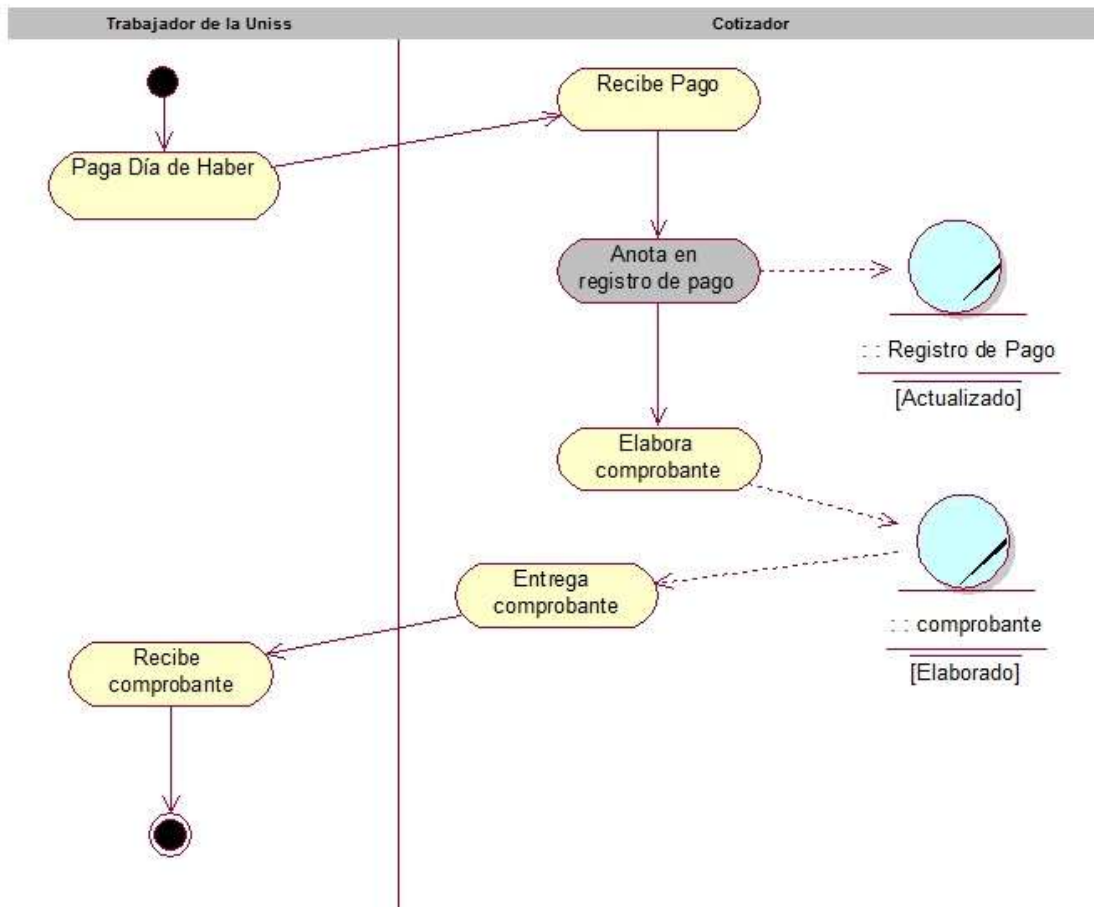


Figura 4: Diagrama de Actividad: Caso de uso Pagar Día Haber.

2.4 Modelo de objetos del negocio

El modelo de objetos del negocio se utiliza para describir la participación de los trabajadores y entidades del negocio, y su colaboración en la realización 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. (González, 2005)

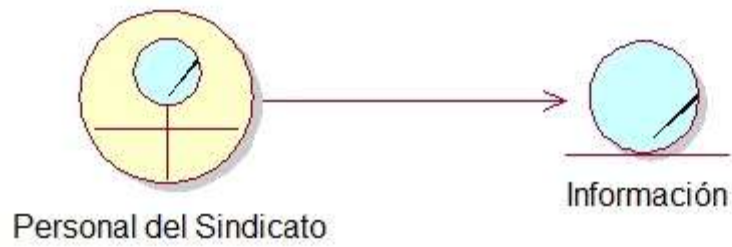


Figura 5: Modelo Objeto Caso de Uso Obtener Información.

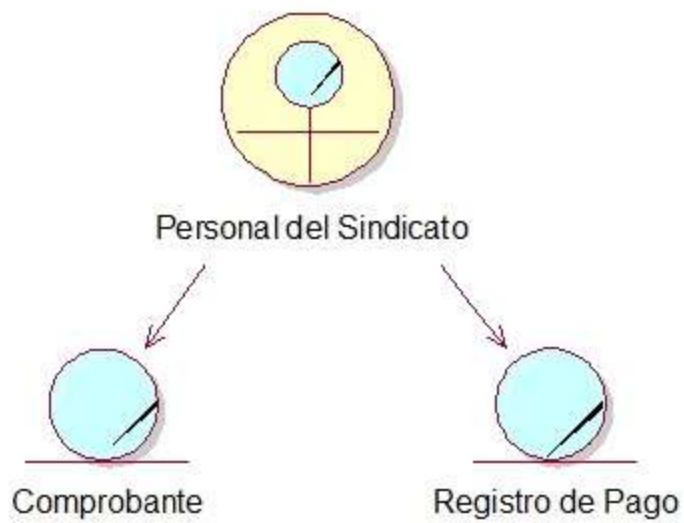


Figura 6: Modelo Objeto Caso de Uso Pago de Cotización.

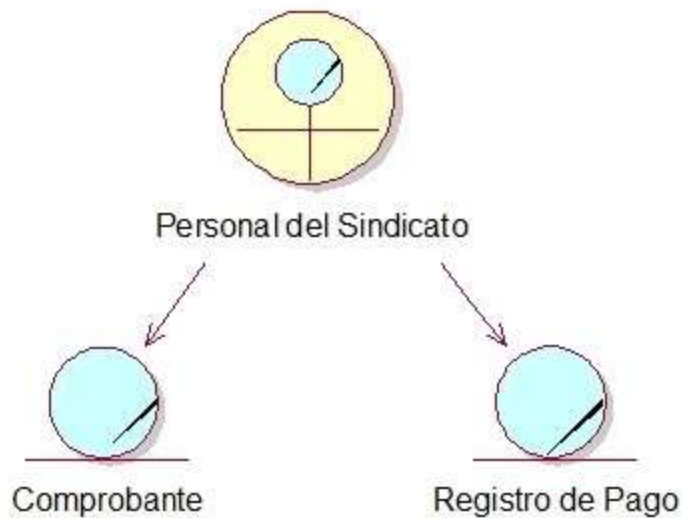


Figura 7: Modelo Objeto Caso de Uso Pago de Día Haber.

2.5 Requisitos funcionales

(González, 2005) plantea que un requerimiento funcional expresa una especificación más detallada de las responsabilidades del sistema que se propone. Ellos permiten determinar, de una manera clara lo que debe hacer el sistema, siempre basándose en las necesidades de los usuarios. Los requerimientos funcionales del sistema propuesto son:

R1: Autenticación.

R2: Cerrar Sesión.

R3: Cambiar Contraseña.

R4: Gestionar Usuario.

4.1 Insertar Usuario.

4.2 Modificar Usuario.

4.3 Eliminar Usuario.

R5: Gestionar Noticias.

5.1 Insertar Noticias.

5.2 Modificar Noticias.

5.3 Eliminar Noticias.

R6: Subir Documentos.

R7: Descargar Documentos.

R8: Gestionar Integrantes.

9.1 Insertar Cumpleaños.

9.2 Modificar Cumpleaños.

9.3 Eliminar Cumpleaños.

R9: Gestionar Sección.

10.1 Insertar Sección.

10.2 Modificar Sección.

10.3 Eliminar Sección.

R10: Gestionar Cotización.

11.1 Insertar Cotización.

11.2 Modificar Cotización.

11.3 Eliminar Cotización.

R11: Gestionar Día de Haber.

12.1 Insertar Día de Haber.

12.2 Modificar Día de Haber.

12.3 Eliminar Día de Haber.

R12: Gestionar Roles.

13.1 Insertar Roles.

13.2 Modificar Roles.

13.3 Eliminar Roles.

R13: Gestionar Área de Trabajo.

14.1 Insertar Área de Trabajo.

14.2 Modificar Área de Trabajo.

14.3 Eliminar Área de Trabajo.

R14: Gestionar Efemérides

15.1 Insertar Efemérides.

15.2 Modificar Efemérides.

15.3 Eliminar Efemérides.

R15: Gestionar Destacados

16.1 Insertar Destacados.

16.2 Modificar Destacados.

16.3 Eliminar Destacados.

R16: Reportar Sección

R17: Reportar Integrantes por Sección

R18: Reportar Usuarios

R19: Reportar Integrantes que faltan por pagar por mes la Cotización

R20: Reportar Integrantes que faltan por pagar Día de Haber

R21: Reportar Jefe de Sección

R22: Reportar Cumpleaños por mes

R23: Reportar Efemérides.

R24: Reportar listado de documentos.

R25: Mostrar Noticia.

R26: Reportar destacados por mes

2.6 Requisitos no funcionales

Los requerimientos no funcionales especifican cualidades, propiedades del sistema, como restricciones del entorno o de la implementación, rendimiento, dependencias de la plataforma. Los requerimientos no funcionales del sistema propuesto son los siguientes:

- **Apariencia o interfaz externa.**

La interfaz debe estar 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 de la aplicación a otro con gran facilidad. Se cuidará porque la aplicación sea lo más interactiva posible.

- **Requerimientos de Usabilidad.**

El sistema podrá ser usado por cualquier persona, instalar el sistema traerá consigo una mayor rapidez de trabajo y por consiguiente un ahorro de materiales.

- **Requerimientos de Rendimiento.**

La eficiencia del sistema estará determinada por el aprovechamiento de los recursos en el modelo a tres capas. El sistema deberá ser lo más estable y confiable posible.

- **Requerimientos de Soporte.**

Se requiere que el producto reciba mantenimiento ante cualquier fallo que ocurra. El sistema será de fácil instalación.

- **Requerimientos de Portabilidad.**

1. El producto correrá sobre una plataforma Web
2. El producto deberá usarse con Mozilla Firefox como navegador.

- **Requerimientos de Seguridad.**

Toda la información estará protegida del acceso no autorizado, los administradores de sistema son los únicos que podrán transformar la información, los operadores solo podrán ver los listados de información.

1. Deberá garantizar la conectividad e integridad de los datos almacenados a través de la red.
2. Deberá garantizar la confidencialidad para proteger la información de acceso no autorizado.

- **Confiabilidad**

La información manejada por la Aplicación Web estará protegida de acceso no autorizado, pues solo la puede modificar, eliminar o insertar el administrador. Además si existe algún fallo se debe garantizar la mínima pérdida de información.

- **Interfaz Interna**

La interfaz interna debe ser compatible con la arquitectura básica de la Universidad de Santi Spíritus “José Martí Pérez”.

- **Requerimientos de Software.**

Para la implantación del sistema se requiere de un servidor con sistema operativo Linux o Windows. Además, se necesita tener un servidor Web Apache con PHP y como servidor de base de datos MySQL.

- **Requerimientos de Hardware.**

1. **Servidor:** La máquina servidora debe tener como mínimo las siguientes características de Hardware: Procesador Pentium III, 512 MB de memoria RAM y 40 GB de capacidad en disco duro.
2. **Cliente:** Las computadoras situadas en los puestos de trabajo de los usuarios requerirán como mínimo 128 MB de memoria RAM. Estas máquinas deben estar conectadas en red con el servidor.

- **Restricciones en el diseño y la implementación.**

Se desea desarrollar una herramienta dinámica para el manejo de la información del Buró Universitario Sindical de la Universidad de Santi Spíritus “José Martí Pérez” en un ambiente de Aplicación Web. Se ha escogido el lenguaje de programación PHP, por las posibilidades multiplataforma que este brinda y lo fácil que resulta de aprender.

- **Políticos Culturales**

A la hora de brindar los servicios que ofrece la Aplicación el nivel social, cultural o étnico, no determinarán una prioridad o limitante.

2.7 Modelo de casos de uso del sistema

El modelado de Casos de Uso es la técnica más simple y que emplean los desarrolladores de software para modelar los requisitos del sistema desde la perspectiva del usuario. El modelo de casos de uso consiste en actores y casos de uso. Los actores representan usuarios y otros sistemas que interaccionan con el sistema y los casos de uso representan el comportamiento del sistema, los escenarios que el sistema atraviesa en respuesta a un estímulo desde un actor (System, 2005)

2.7.1 Actores del sistema

Un actor no es más que un rol que juega un usuario de Caso de Uso cuando interaccionan con estos casos de uso. Los actores representan a terceros fuera del sistema que colaboran con el mismo. Una vez que se han identificado los actores del sistema, se ha identificado el entorno externo del sistema. (Rumbaugh, Booch, & Jacobson, 2006)

Teniendo en cuenta lo planteado anteriormente se definieron los siguientes actores del sistema:

Actores	Justificación
Administrador	Tiene el control total del sistema, es quien crea las cuentas de acceso al mismo y le asigna a cada usuario su ID y su contraseña, así como los permisos; además de poder modificar, insertar, y eliminar cualquier información publicada en la aplicación.
Visitante	Es la persona que tiene acceso a la información del sistema sin necesidad de autenticación, aunque solo tiene permiso para ver los detalles de la información relacionadas con el BUS.
Cotizador	Es la persona que maneja todo lo relacionado con la cotización y el día de haber.

Tabla 6: Actores del Sistema.

2.7.2 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. 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 (Jacobson, Booch, & Rumbaugh, 2000)

Para este sistema se definieron los siguientes casos de uso:

1. Autenticación.
2. Cerrar Sesión.
3. Cambiar Contraseña.
4. Gestionar Usuario.
5. Gestionar Noticias.
6. Subir Documentos.
7. Descargar Documentos.
8. Gestionar Integrantes.
9. Gestionar Sección.
10. Gestionar Cotización.
11. Gestionar Día de Haber.
12. Gestionar Roles.
13. Gestionar Área de Trabajo.
14. Gestionar Efemérides
15. Gestionar Destacados
16. Reportar Sección
17. Reportar Integrantes por Sección
18. Reportar Usuarios
19. Reportar Integrantes que faltan por pagar por mes la Cotización
20. Reportar Integrantes que faltan por pagar Día de Haber
21. Reportar Jefe de Sección
22. Reportar Cumpleaños por mes
23. Reportar Efemérides.
24. Reportar listado de documentos.

25. Mostrar Noticia.

26. Reportar destacados por mes

2.7.3 Diagrama de casos de uso del sistema.

Para facilitar el trabajo con los casos de uso y la organización de los elementos se agrupan los casos de uso en paquetes, los cuales se muestran a continuación.

Nombre del paquete	Funcionalidades	Criterio de agrupamiento	Actores
Seguridad	<ul style="list-style-type: none">• Autenticación.• Cerrar Sesión.• Cambiar contraseña.	Funcionalidades (Requerimientos funcionales)	Administrador y Cotización
Nomencladores	<ul style="list-style-type: none">• Gestionar Usuario.• Gestionar Noticias.• Subir Documentos.• Gestionar Integrantes.• Gestionar Sección.• Gestionar Cotización.• Gestionar Día de Haber.• Gestionar Roles.• Gestionar Área de Trabajo.• Gestionar Efemérides• Gestionar Destacados	Funcionalidades (Requerimientos funcionales)	Administrador y Cotización
Reportes	<ul style="list-style-type: none">• Descargar Documentos.• Reportar Sección• Reportar Integrantes por Sección• Reportar Usuarios• Reportar Integrantes	Funcionalidades (Requerimientos funcionales)	Administrador, Cotización y Visitante

	<p>que faltan por pagar por mes la Cotización</p> <ul style="list-style-type: none"> • Reportar Integrantes que faltan por pagar Día de Haber • Reportar Jefe de Sección • Reportar Cumpleaños por mes • Reportar Efemérides. • Reportar listado de documentos. • Mostrar Noticia. • Reportar destacados por mes 		
--	---	--	--

Tabla 7: Descripción de los paquetes

En la siguiente figura se muestran los paquetes que agrupan a los casos de uso.

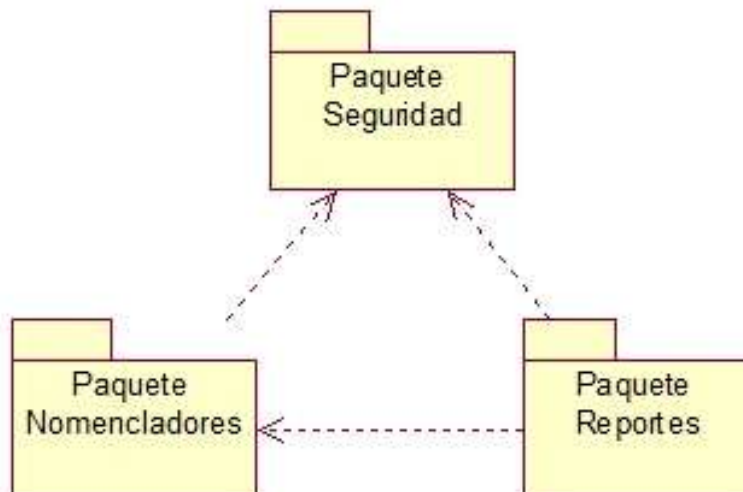


Figura 8: Casos de uso por paquetes.

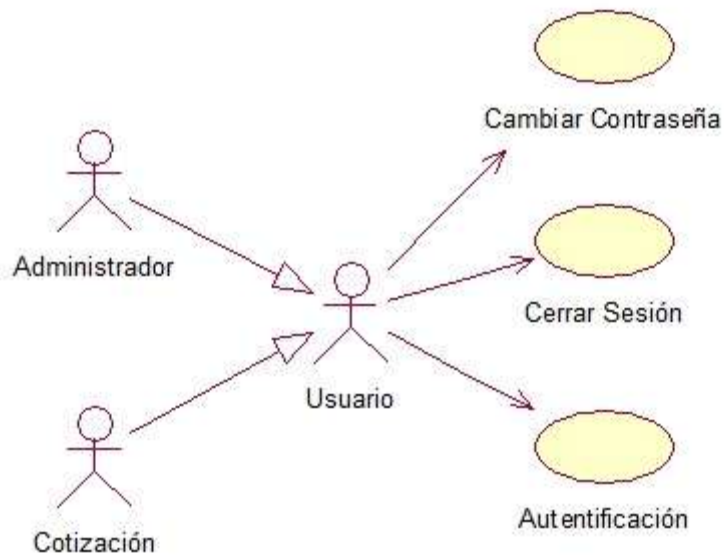


Figura 9: Diagrama de casos de uso del sistema Paquete de Seguridad

Descripción de los casos de uso Paquete de Seguridad Anexo 1.

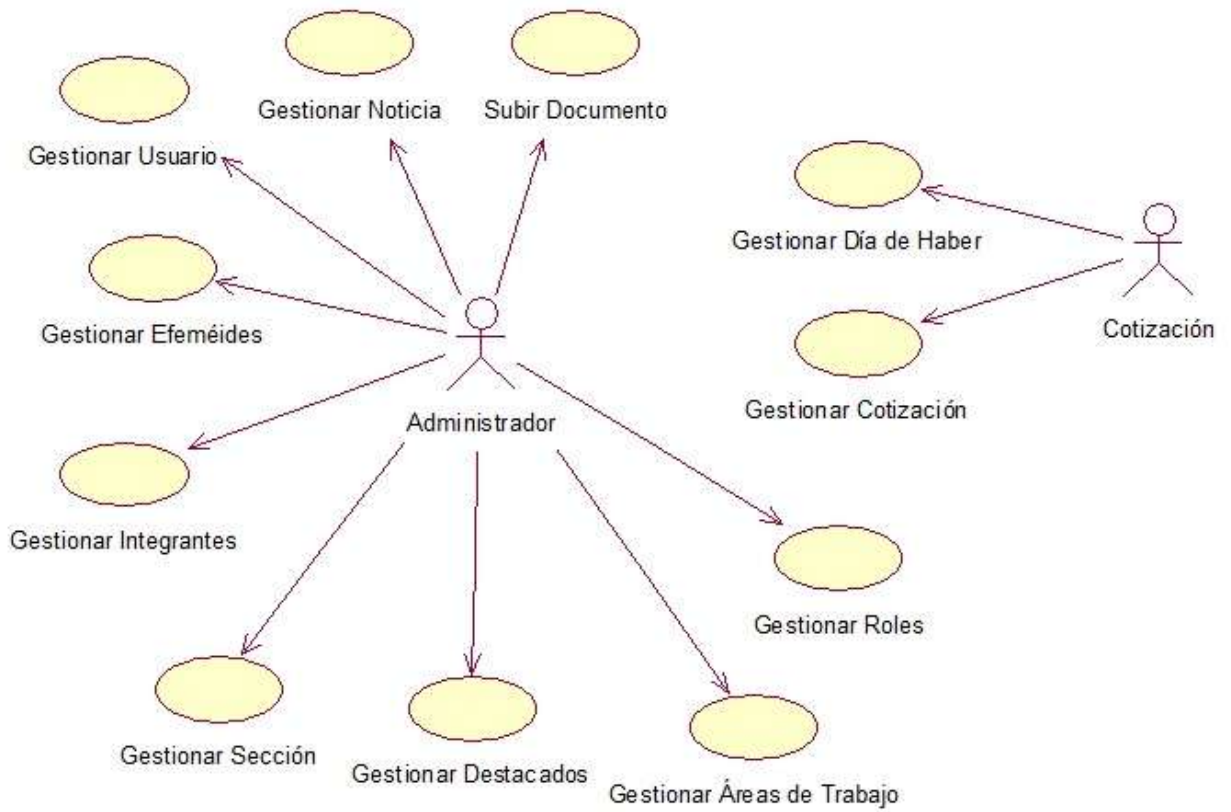


Figura 10: Diagrama casos de uso del Sistema Paquete Nomencladores.

Descripción de los casos de uso Paquete Nomencladores Anexo 2.

Capítulo III: Construcción de la aplicación propuesta para la gestión de la información para el BUS de la Universidad de Sancti Spíritus “José Martí Pérez”.

Introducción

En este capítulo se aborda el diseño y la implementación del sistema utilizando UML para su modelado. Para facilitar este proceso se generan un conjunto de diagramas, tales como: el diagrama de clases del diseño, el diagrama del modelo lógico, físico de datos y el diagrama de componentes.

3.1 Diagrama de clases Web

Una clase de diseño es una abstracción de una clase o construcción similar en la implementación del sistema.(Jacobson, Booch, & Rumbaugh, 2000)

Un diagrama de clases es una colección de elementos declaratorios del modelo, como clases, tipos y sus relaciones; conectados unos a otros y a sus contenidos en forma de grafo.(Jacobson, Booch, & Rumbaugh, 2000)

Los diagramas de clases del diseño fueron elaborados a partir de los diferentes casos de uso del sistema:

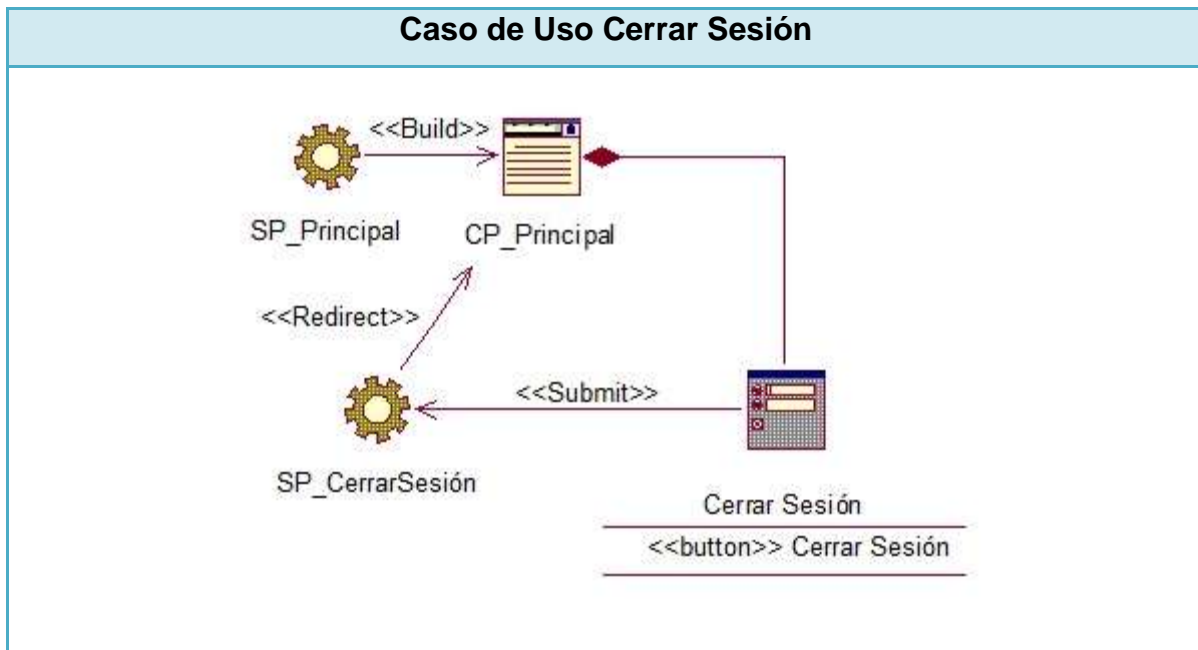


Figura 12: Diagrama de clases Web Caso de Uso Cerrar Sesión

Caso de Uso Autenticación

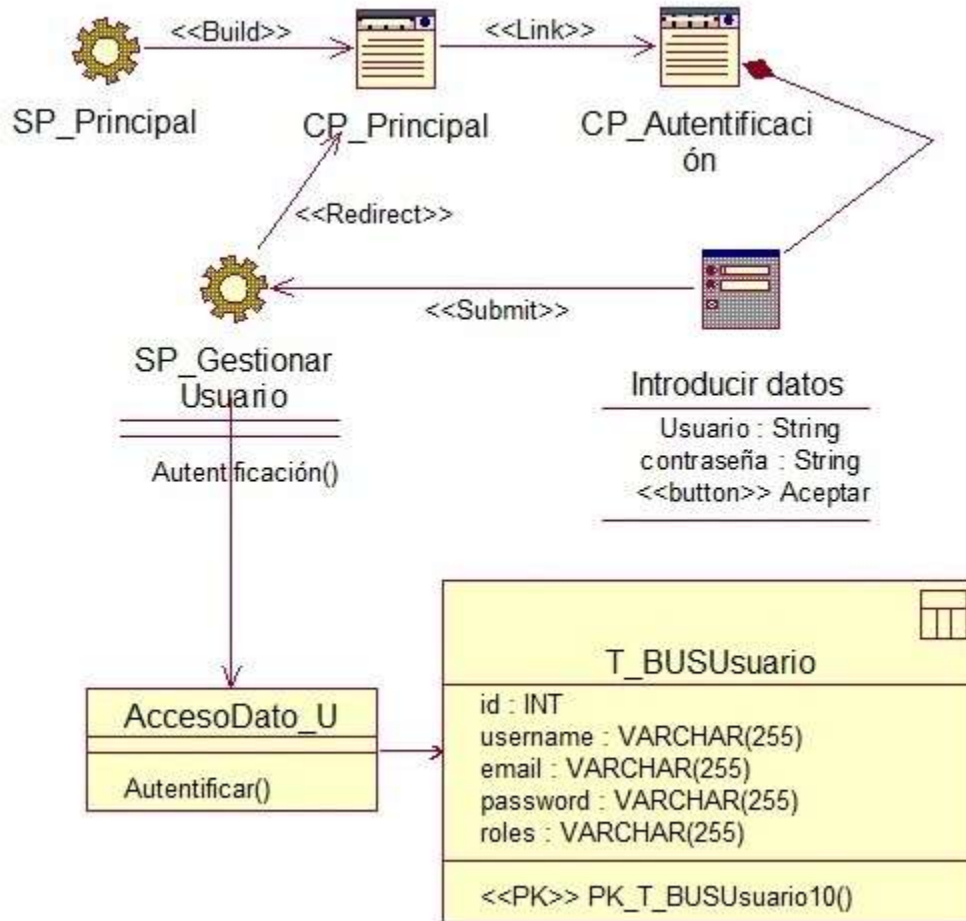


Figura 13: Diagrama de clases Web Caso de Uso Autenticación

Caso de Uso Cambiar Contraseña

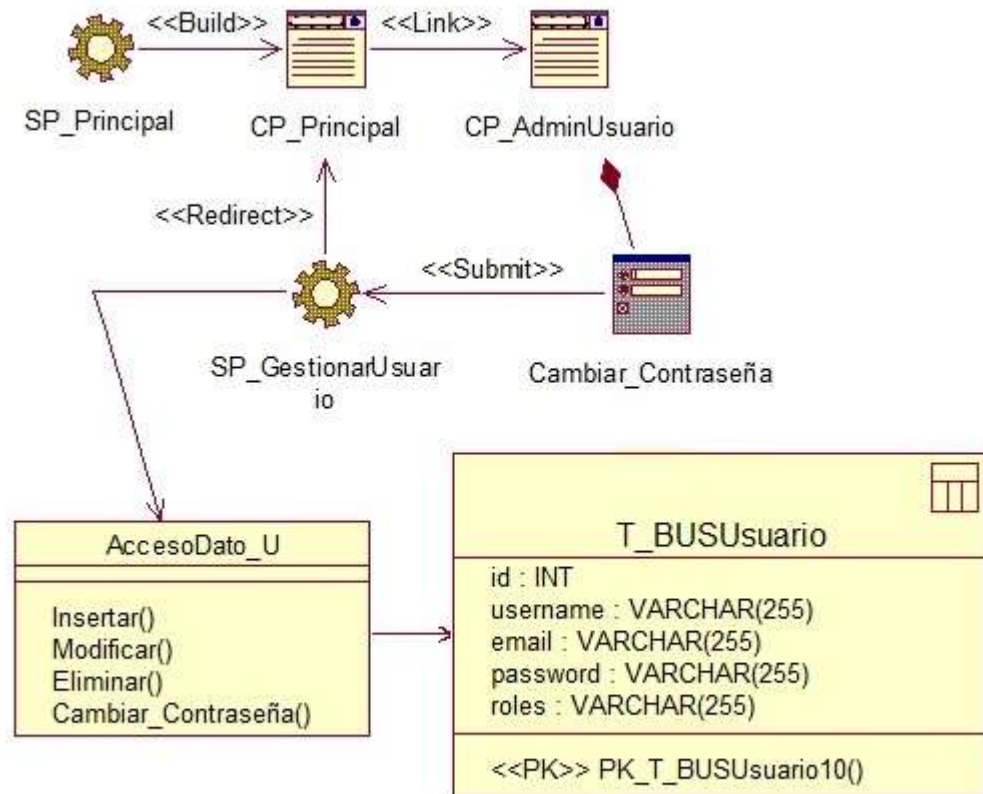


Figura 14: Diagrama de clases Web Caso de Uso Cambiar Contraseña

Paquete Nomencladores

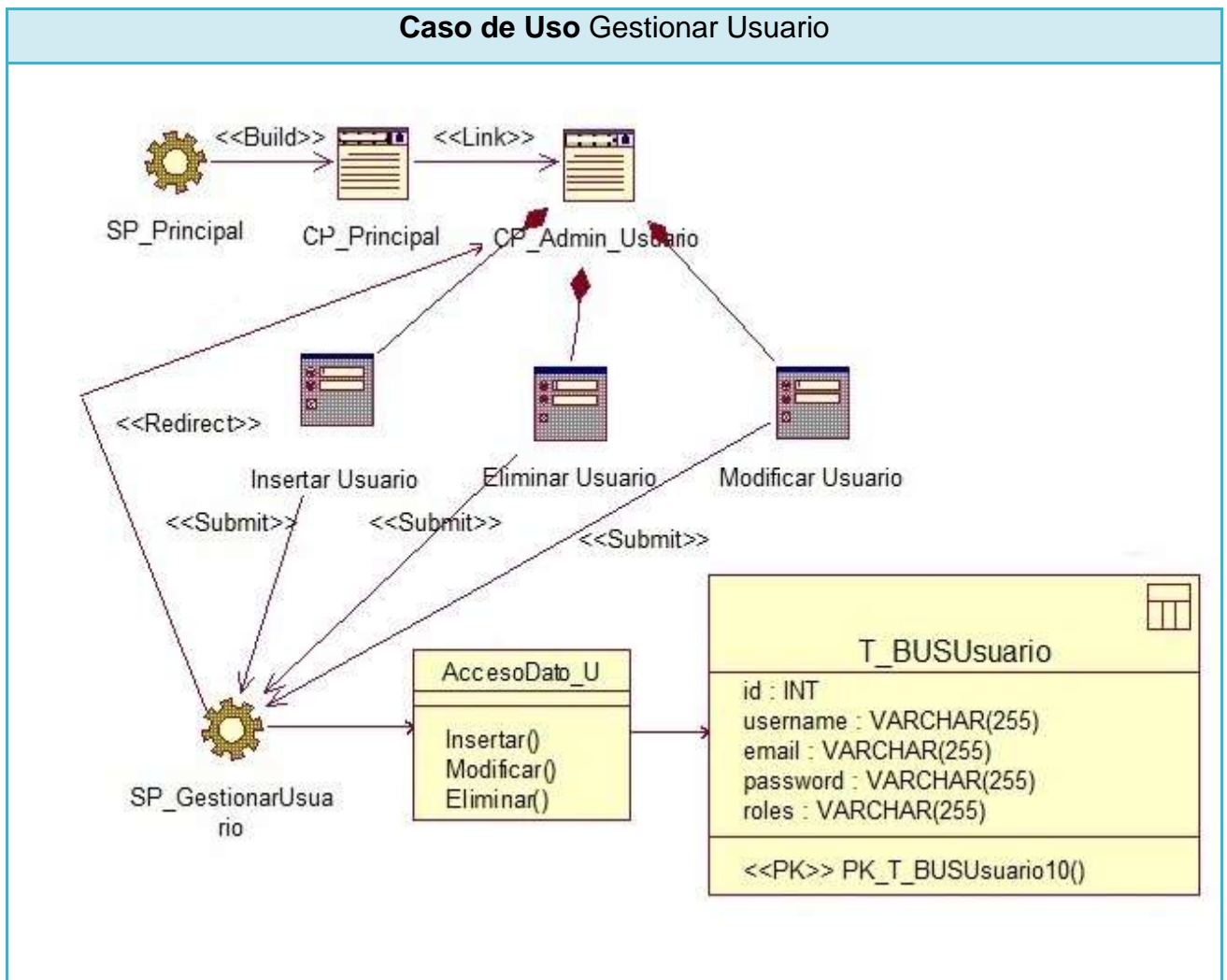


Figura 15: Diagrama de clases Web Caso de Uso Gestionar Usuario

Caso de Uso Gestionar Noticia

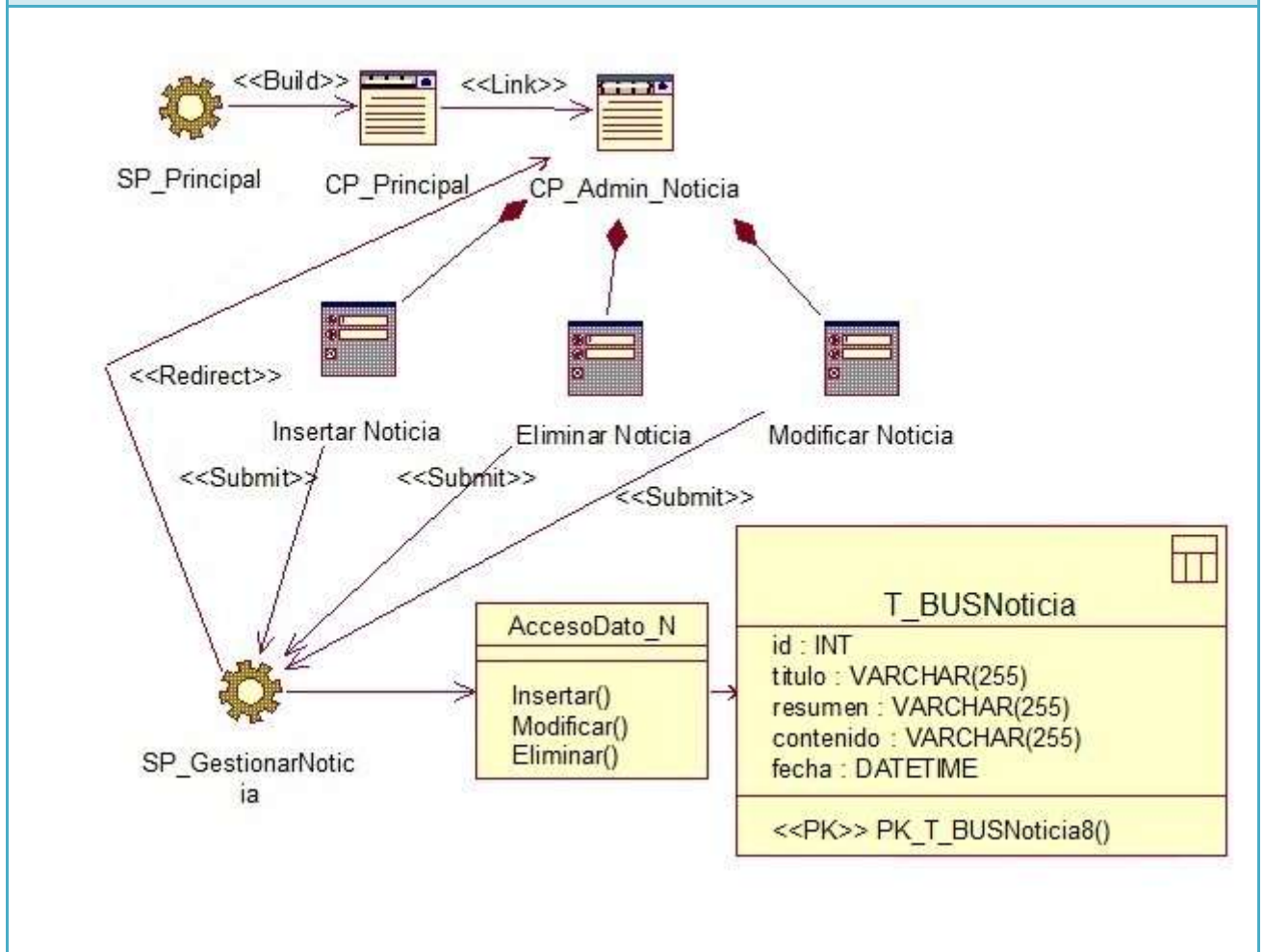


Figura 16: Diagrama de clases Web Caso de Uso Gestionar Noticia

Caso de Uso Gestionar Subir Documento

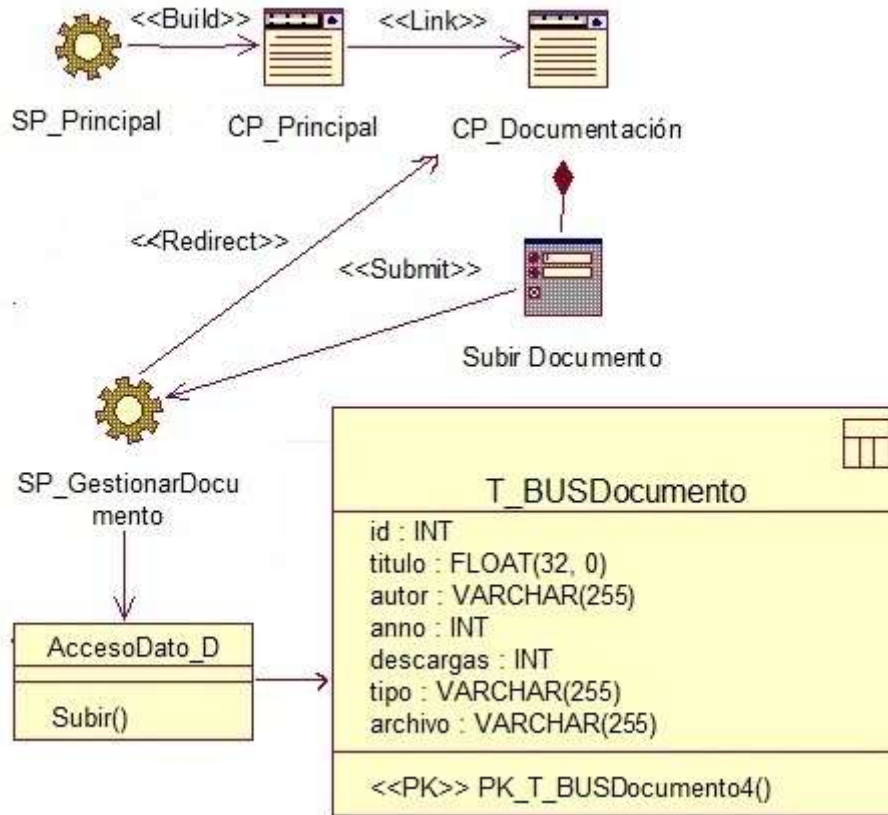


Figura 17: Diagrama de clases Web Caso de Uso Subir Documento

Caso de Uso Gestionar Integrantes

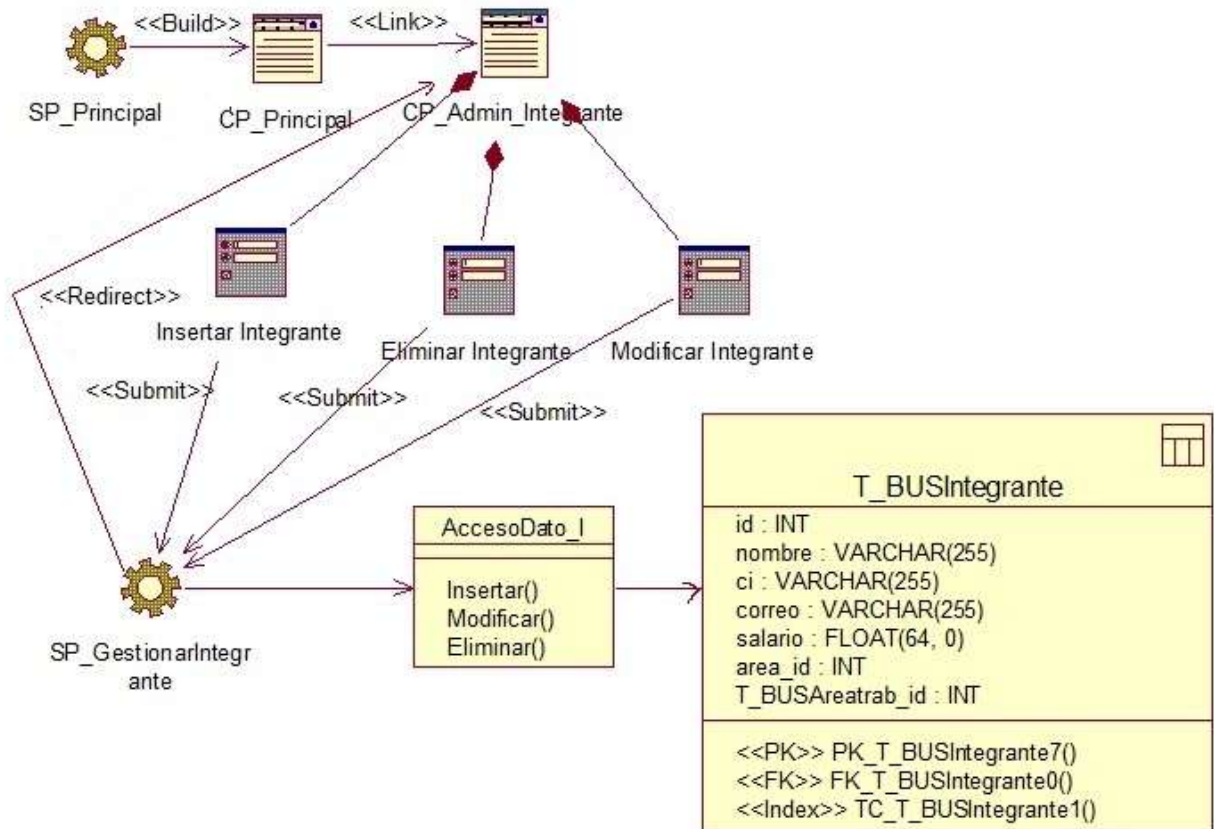


Figura 18: Diagrama de clases Web Caso de Uso Gestionar Integrantes

Caso de Uso Gestionar Sección

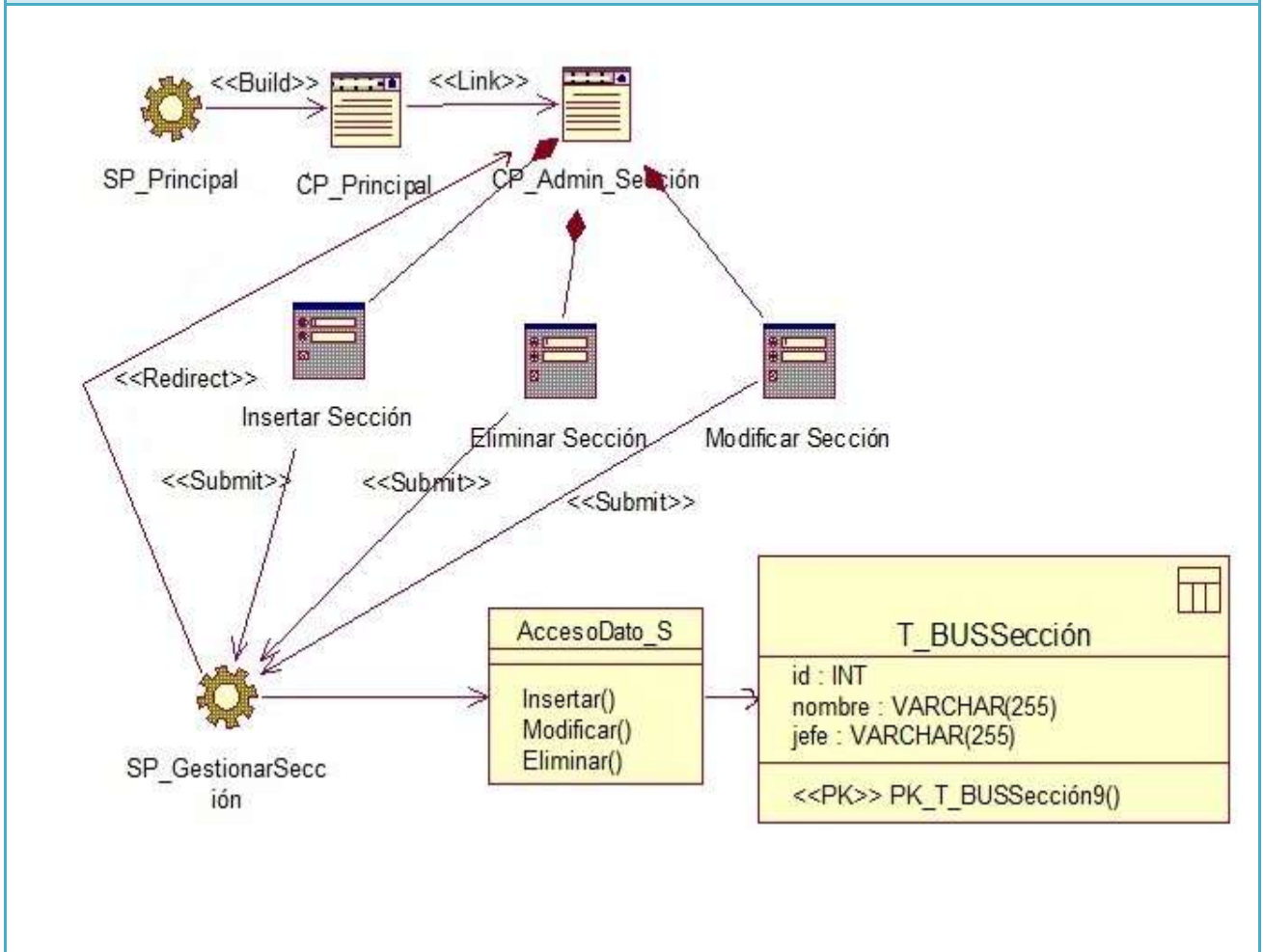


Figura 19: Diagrama de clases Web Caso de Uso Gestionar Sección

Caso de Uso Gestionar Cotización

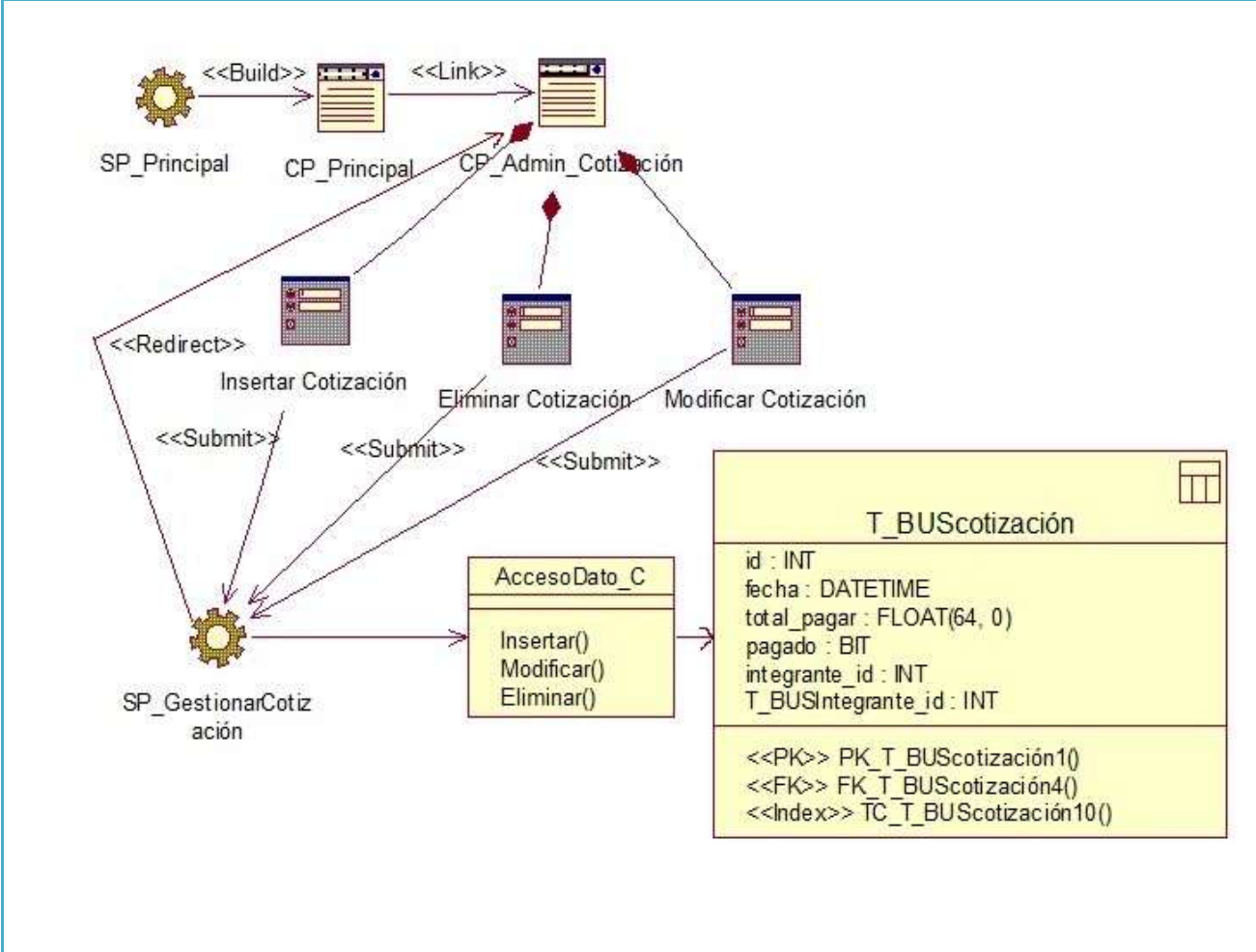


Figura 20: Diagrama de clases Web Caso de Uso Gestionar Cotización

Caso de Uso Gestionar Día de Haber

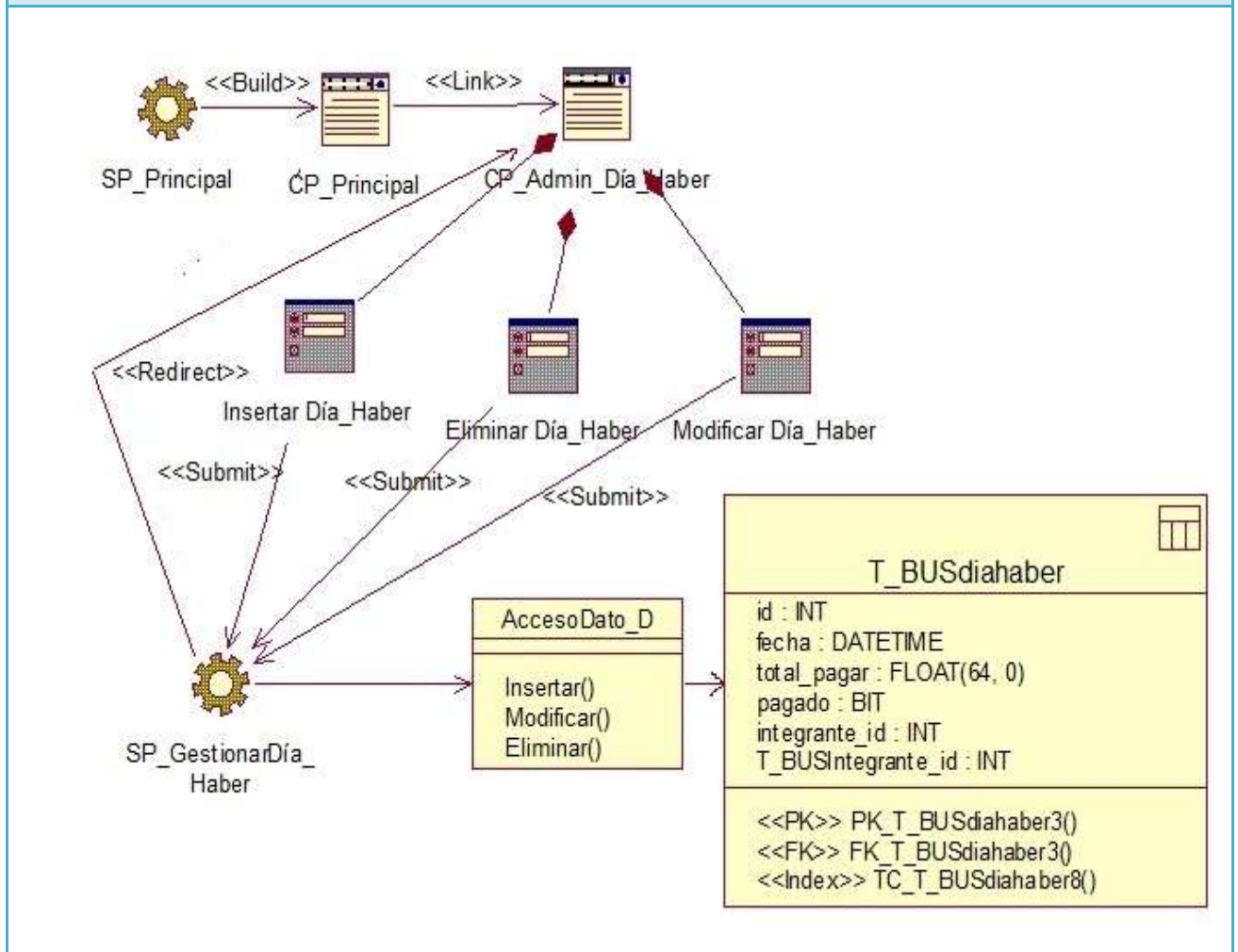


Figura 21: Diagrama de clases Web Caso de Uso Gestionar Día de Haber

Caso de Uso Gestionar Área de Trabajo

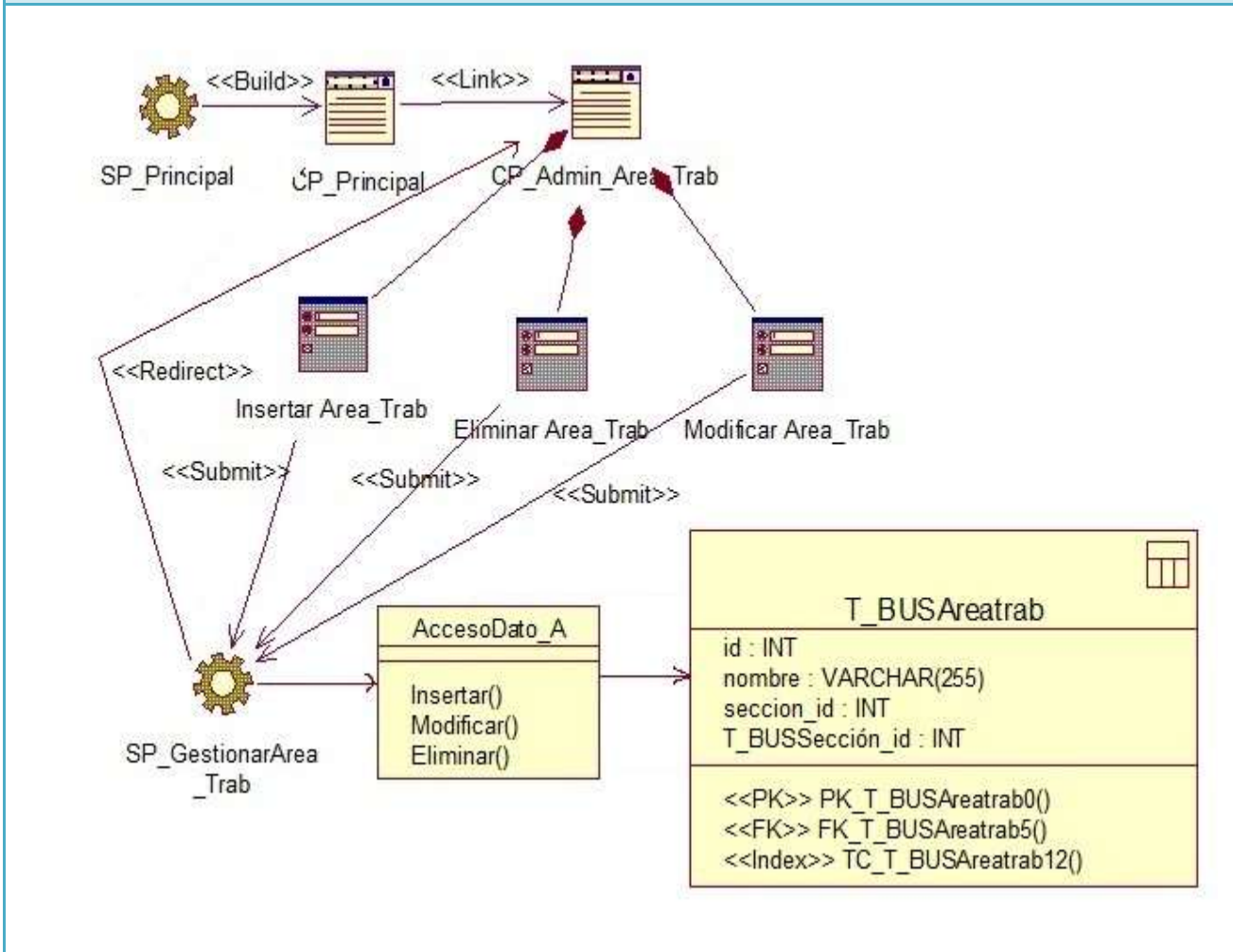


Figura 22: Diagrama de clases Web Caso de Uso Gestionar Área de trabajo

Caso de Uso Gestionar Efemérides

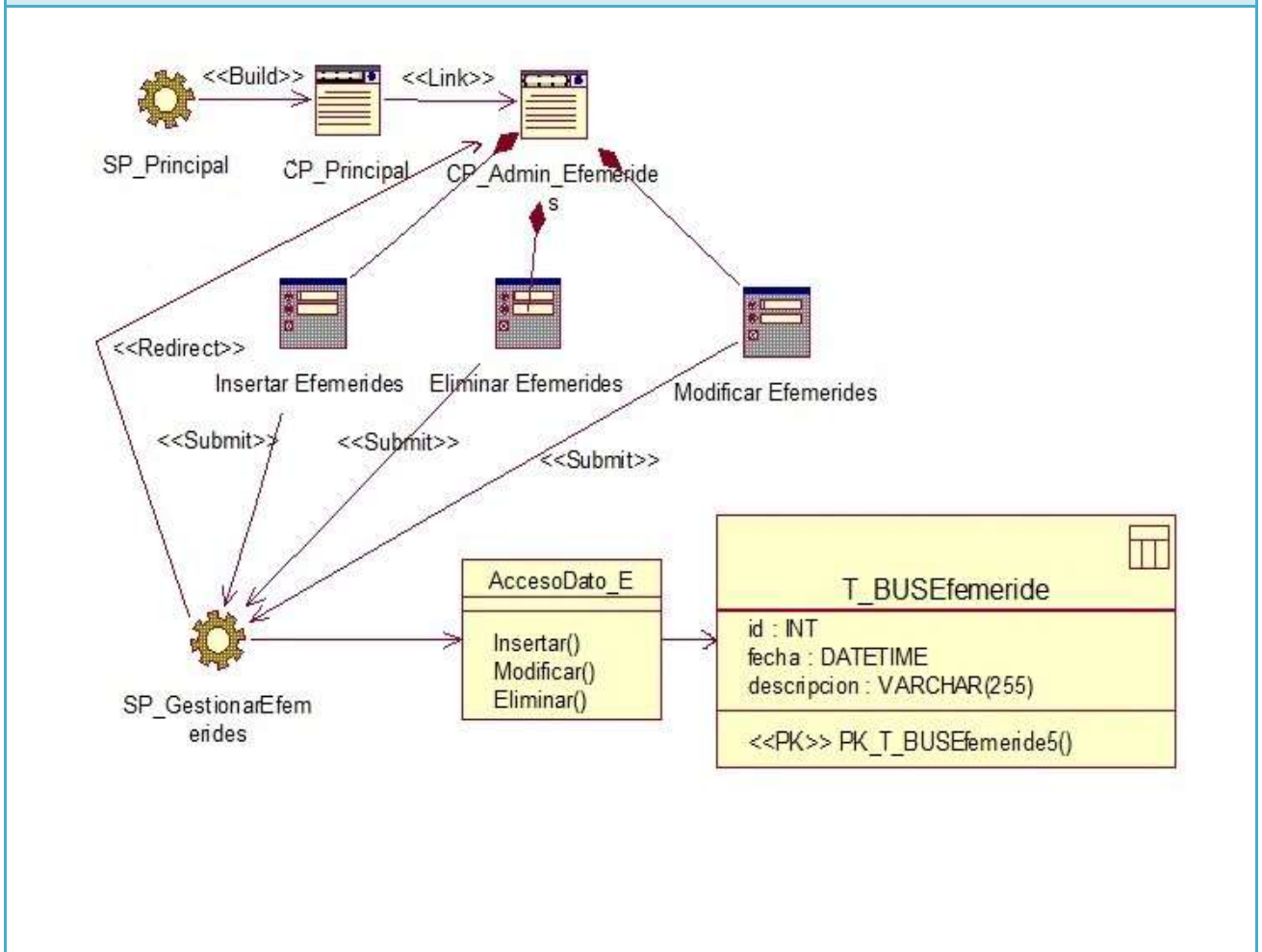


Figura 23: Diagrama de clases Web Caso de Uso Gestionar Efemérides

Caso de Uso Gestionar Destacados

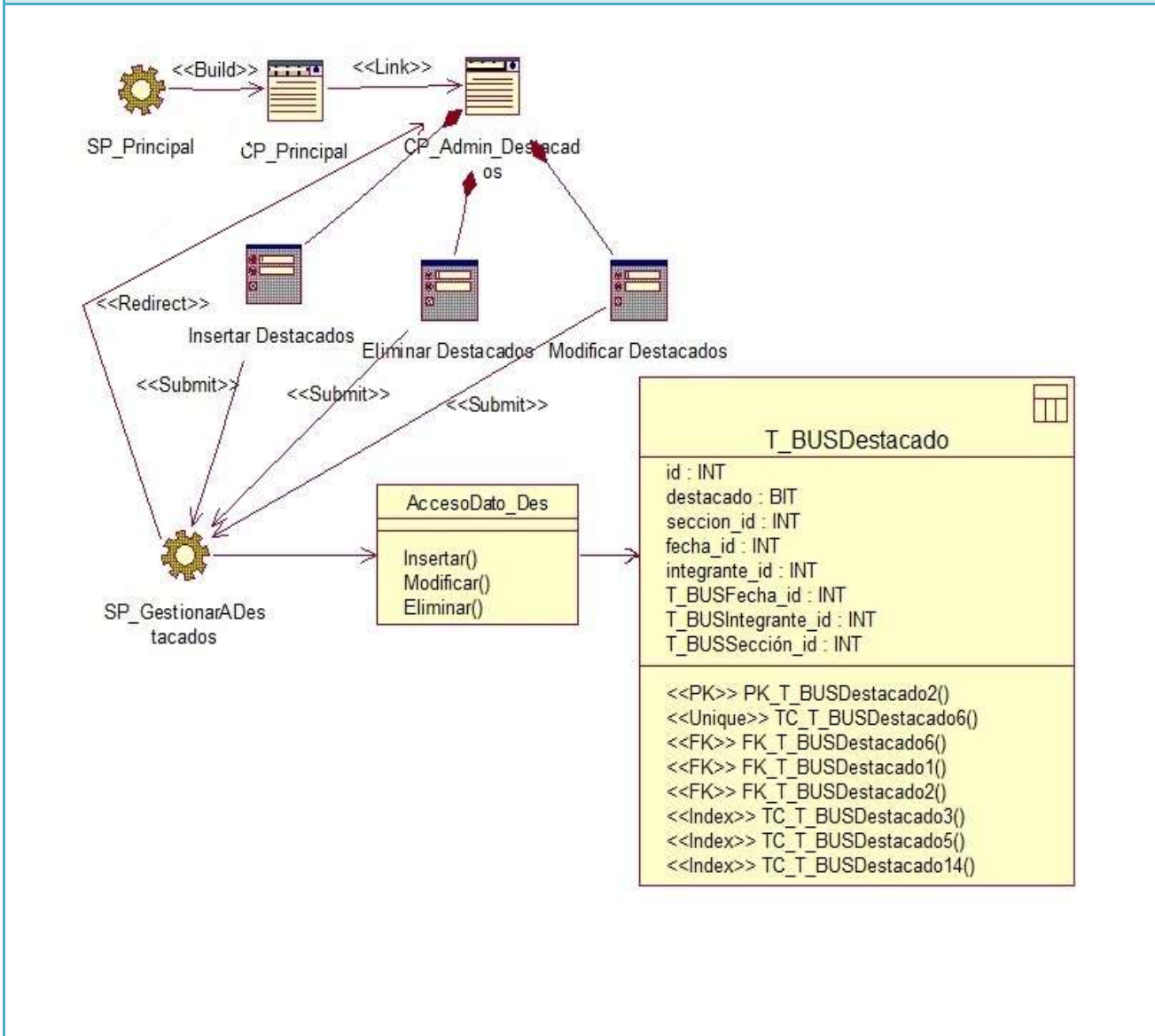


Figura 24: Diagrama de clases Web Caso de Uso Gestionar Destacados

Paquete Reportes

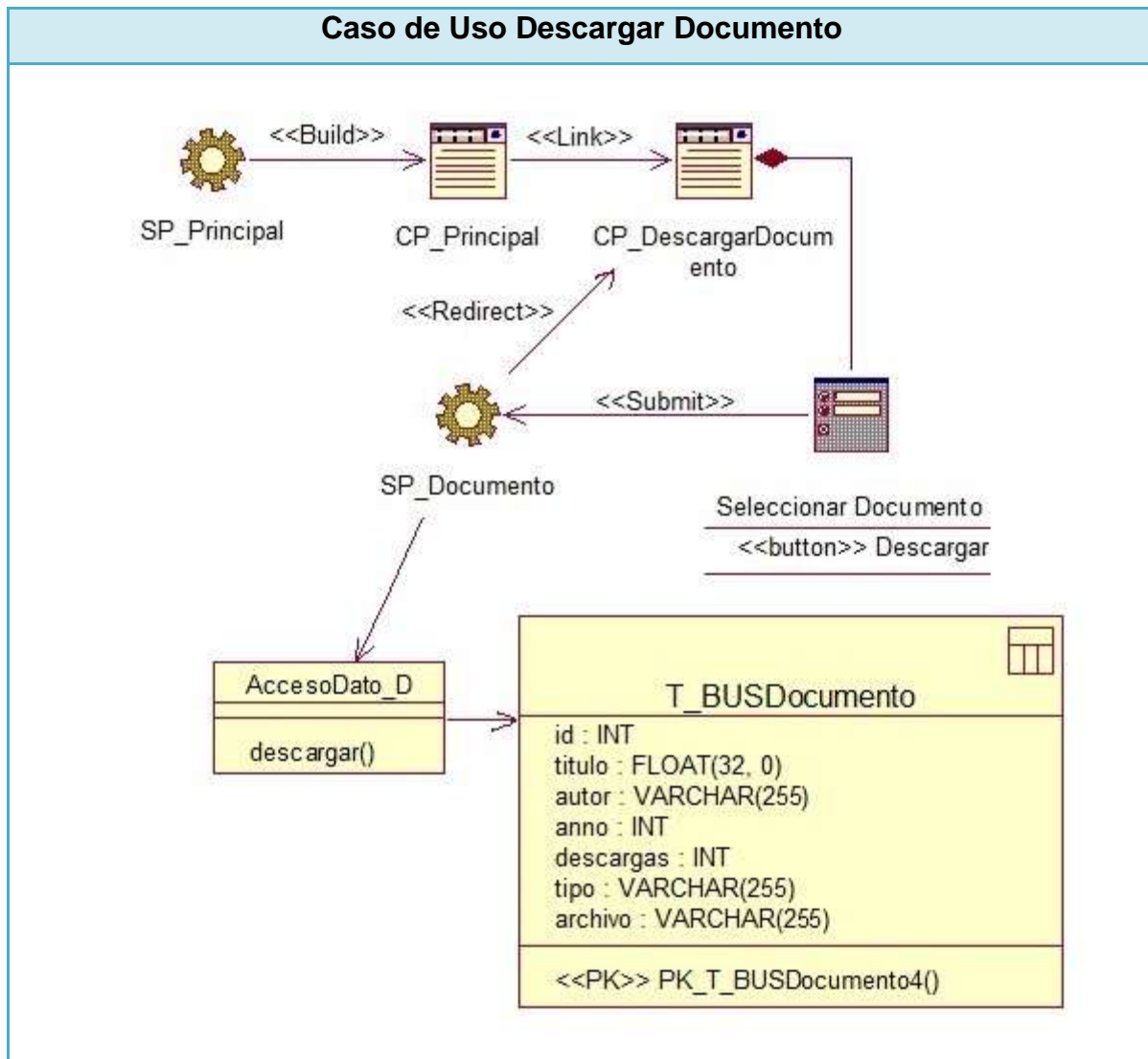


Figura 25: Diagrama de clases Web Caso de Uso Descargar Documento

Caso de Uso Reportar Sección

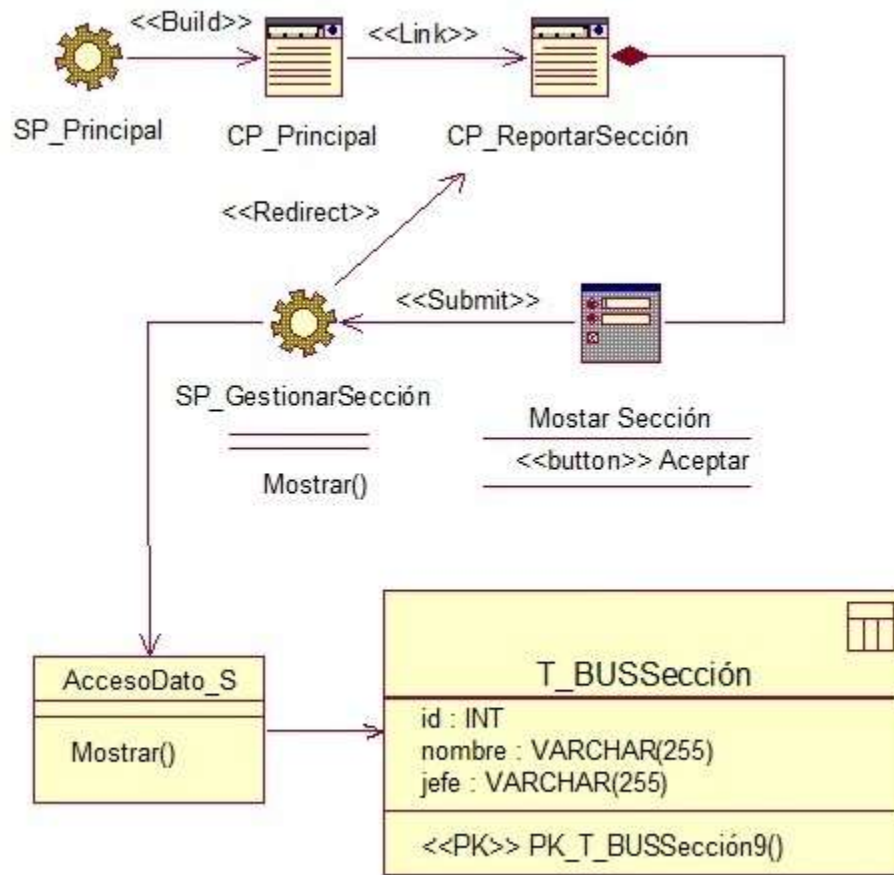


Figura 26: Diagrama de clases Web Caso de Uso Reportar Sección

Caso de Uso Reportar Integrantes por Sección

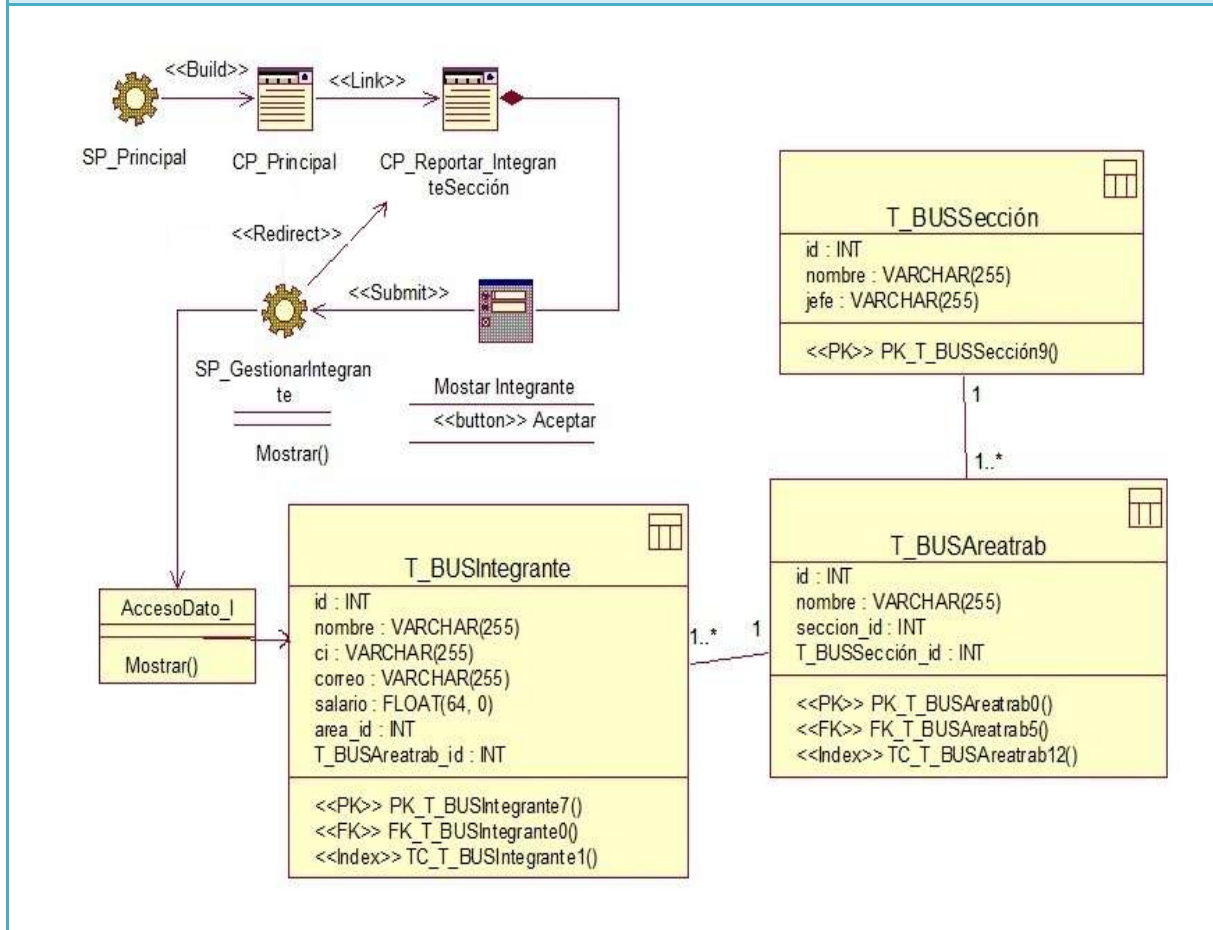


Figura 27: Diagrama de clases Web Caso de Uso Reportar Integrantes por Sección

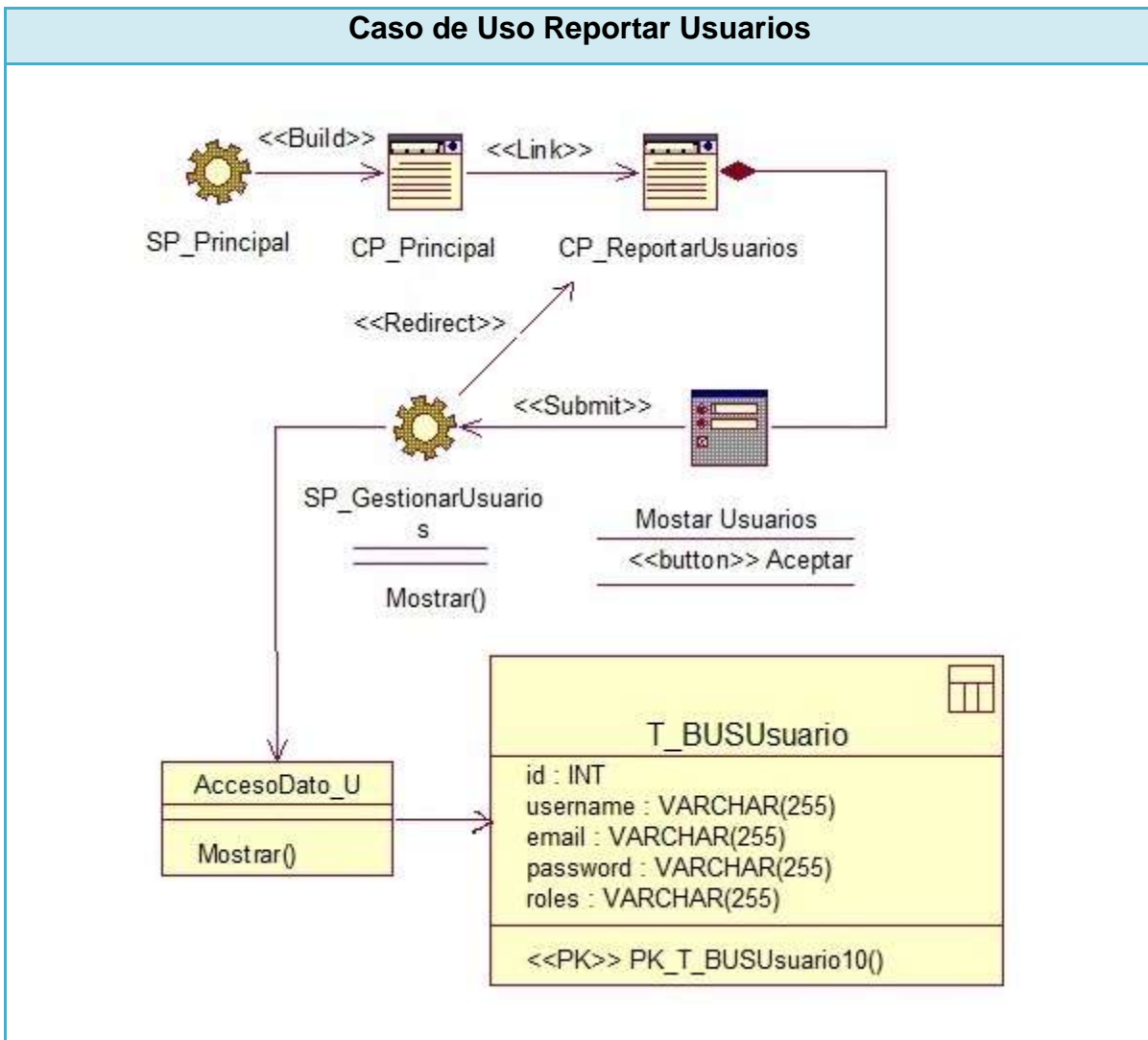


Figura 28: Diagrama de clases Web Caso de Uso Reportar Usuarios

Los restantes Diagramas de clases Web se encuentran en el Anexo 4.

3.2 Diseño de la Base de Datos

Para realizar un adecuado diseño de la base de datos, es necesario tener definidos los datos y sus relaciones, sobre el tema Matos formula: Cuando se va a realizar el diseño de la base de datos para un sistema dado, es necesario determinar los datos que se deben tomar en cuenta y las dependencias funcionales existentes entre ellos. Esto se obtiene luego de realizada la etapa de análisis del sistema y partiendo de lo obtenido en ésta.(Mato García, 2006)

Diagrama de clases persistentes

En el diagrama de clases persistentes aparecen las clases que persisten, las cuales poseen la capacidad de mantener su valor en el espacio y en el tiempo. (Rumbaugh, Booch, & Jacobson, 2006). Está compuesto por clases, asociaciones y atributos; interfaces, con sus operaciones y constantes; métodos; información sobre los tipos de atributos, entre otros.

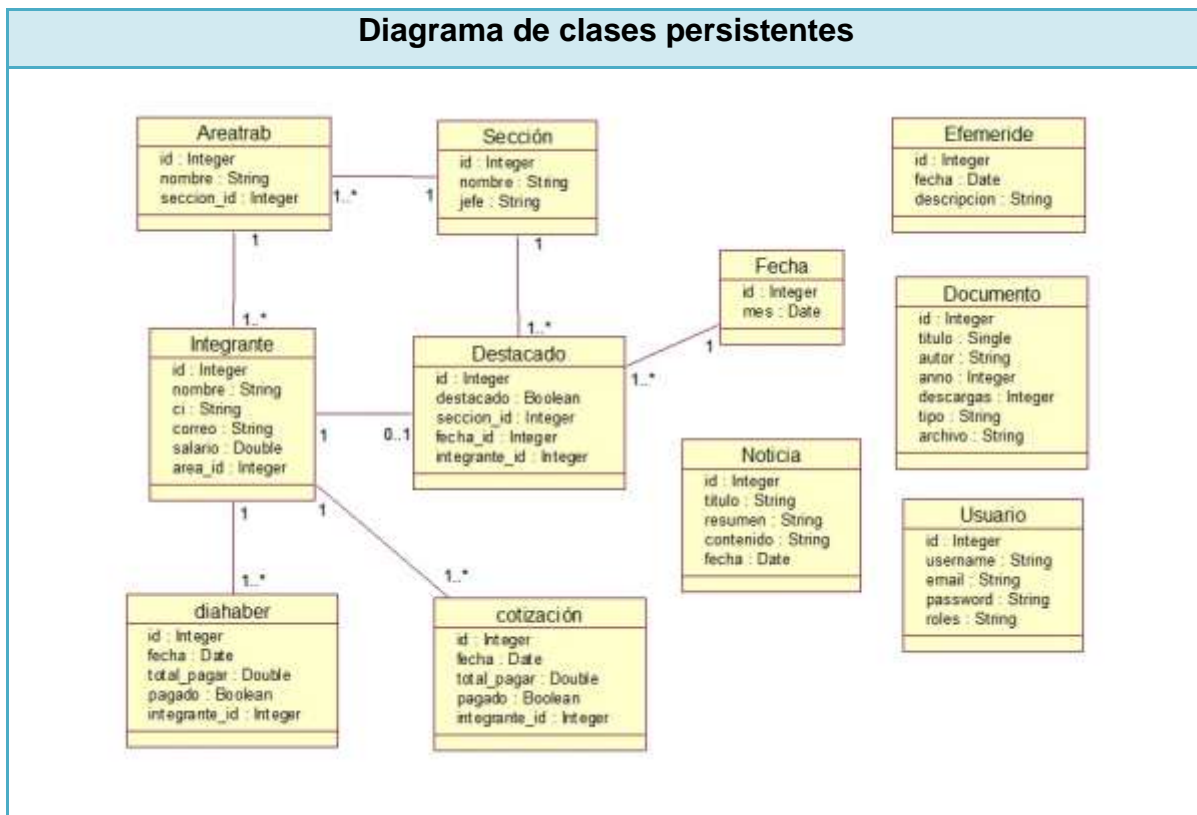


Figura 29: Diagrama de clases persistentes

Diagrama del Modelo Físico de Datos

El modelo físico de datos, representa la estructura o descripción física de las tablas de la base de datos, obtenido a partir del modelo lógico de datos. (Rumbaugh, Booch, & Jacobson, 2006)

Diagrama de Modelo físico de datos

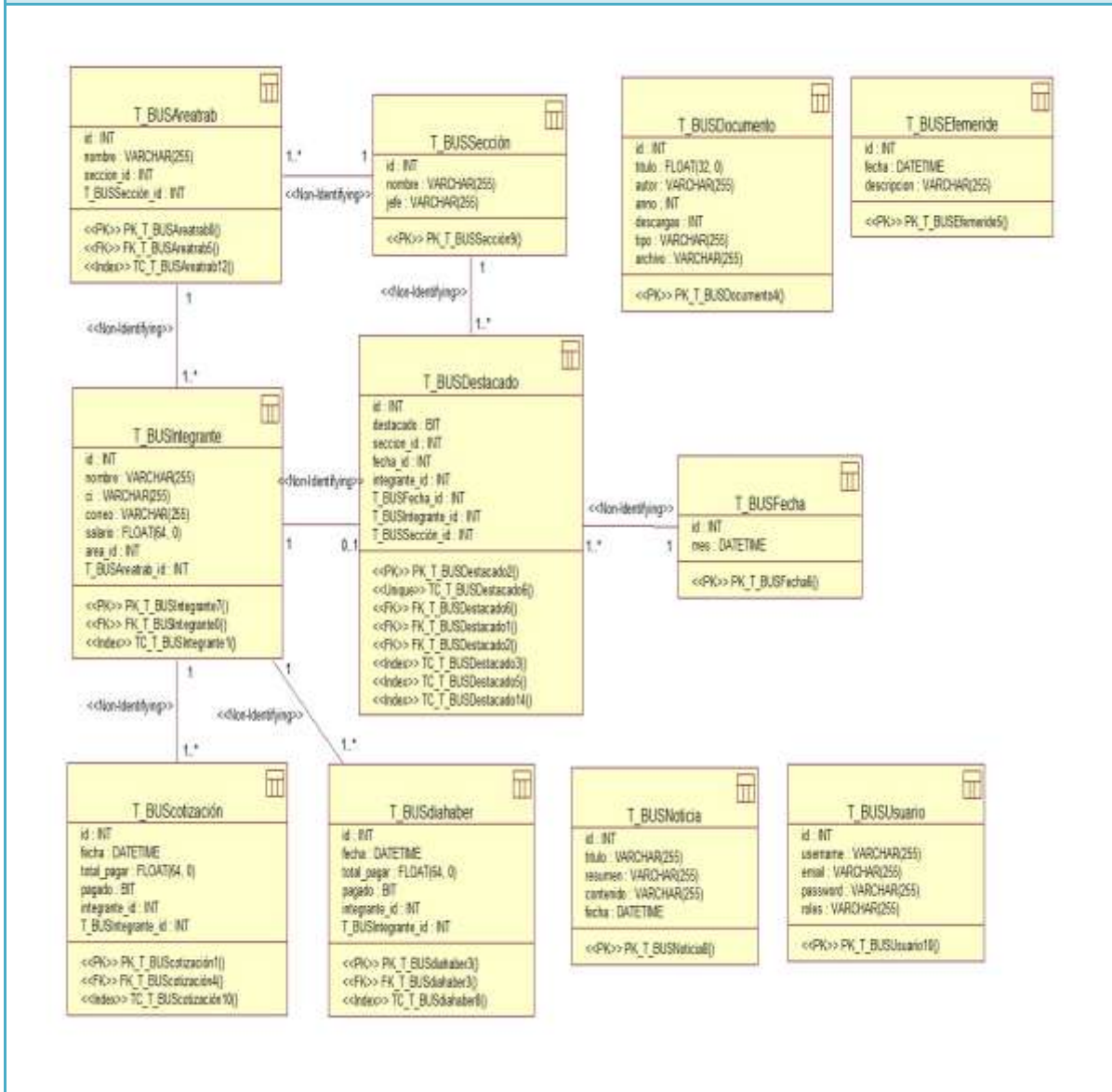


Figura 30: Diagrama de Modelo físico de datos

3.3 Modelo de Implementación

El modelo de implementación describe cómo los elementos del modelo del diseño, se implementan en términos de componentes, describe cómo se organizan los componentes de acuerdo con los mecanismos de estructuración y modularización disponibles en el entorno de implementación y en los lenguajes de programación

utilizados y cómo dependen unos componentes de otros. (Jacobson, Booch, & Rumbaugh, 2000)

Diagrama de despliegue

El diagrama de despliegue muestra la forma en que los componentes se desplegarán en el sistema, en este se detalla las especificaciones de la red, del servidor y los requisitos de hardware. Según Jacobson, Booch, & Rumbaugh un diagrama de despliegue es un diagrama que muestra la configuración de los nodos que participan en la ejecución y de los componentes que residen en ellos. (Rumbaugh, Booch, & Jacobson, 2006)

A continuación se muestra el Diagrama de Despliegue de la aplicación:

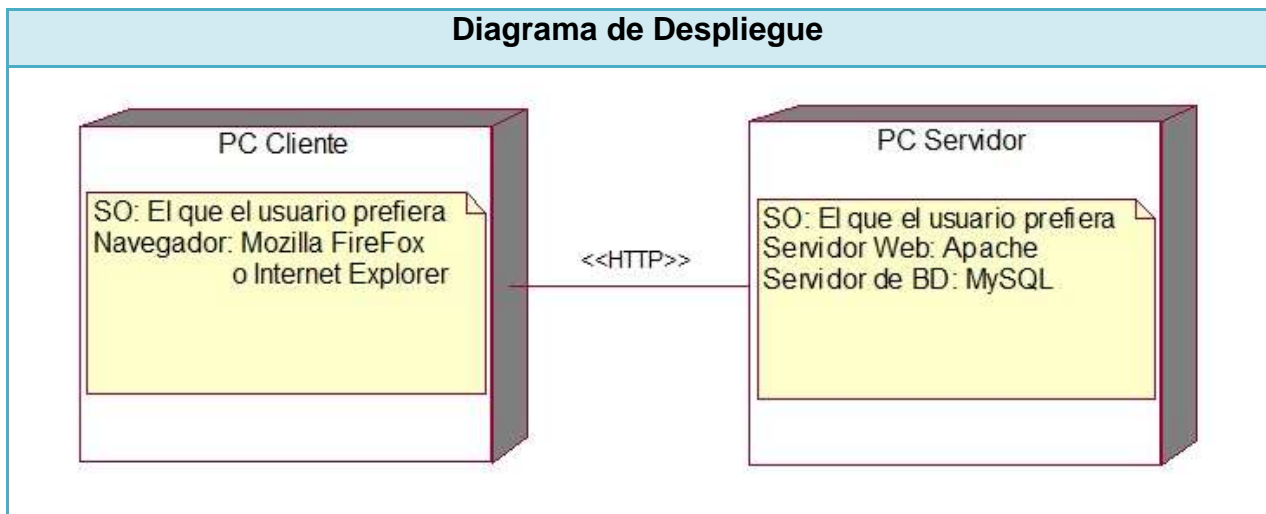


Figura 31: Diagrama de Despliegue

Diagrama de componentes

Los diagramas de componentes contienen los principales elementos de implementación y las relaciones existentes entre ellos, proporcionan un alto nivel de comprensión de la arquitectura del sistema que será construido, permitiendo a los desarrolladores formalizar un plan de trabajo para la ejecución, y tomar

decisiones sobre la asignación de tareas y habilidades necesarias para las mejoras futuras del software.

Sobre los diagramas de componentes Jacobson, Booch, & Rumbaugh afirman, los diagramas de componentes muestran la organización y las dependencias entre un conjunto de componentes, cubren la vista de implementación estática de un sistema, se relacionan con los diagramas de clases en que un componente se corresponde, por lo común, con una o más clases, interfaces o colaboraciones. (Rumbaugh, Booch, & Jacobson, 2006)

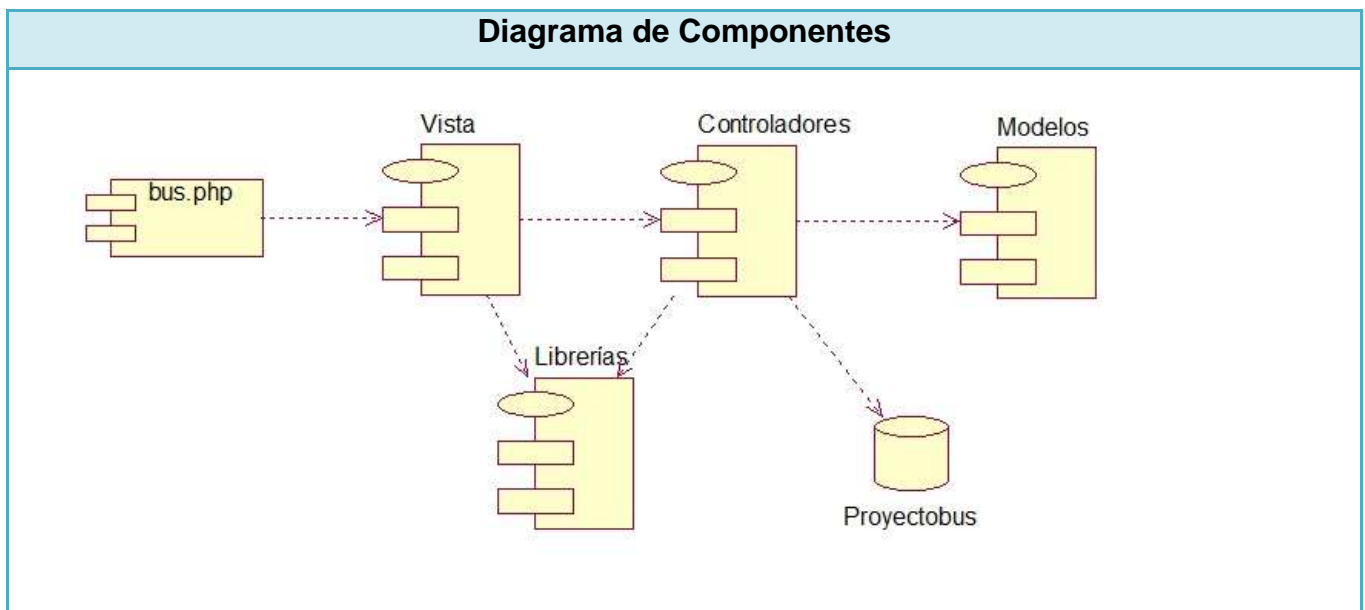


Figura 32: Diagrama de Componentes

Descripción del diagrama de componentes

Paquete vista: Agrupa los ficheros de las clases de interfaz.

Paquete controlador: Agrupa las clases controladoras, y las de AccesoDato que actúan como intermediarias entre las clases del paquete modelo y el paquete vista.

Paquete modelo: Contiene las clases de acceso a los datos.

Proyectobus: Representa al Sistema Gestor de Base de Datos donde se guarda la información de la aplicación, este caso se utilizó MySQL.

Paquete librería: Este paquete contiene las librerías para la validación, generación de tablas, disponibles en el marco de trabajo (framework) Symfony, además de las librerías JavaScript utilizadas para las interfaces que se muestran.

3.4 Principios de diseño

El diseño de la interfaz de una aplicación web, el formato de los reportes, la concepción de la ayuda y el tratamiento de excepciones tiene gran influencia en el éxito o fracaso del mismo. A continuación se describen los principios de diseño seguidos para el desarrollo del sistema en cuestión.

Estándares en la interfaz de la aplicación

La interfaz de una aplicación, es uno de los factores que influyen en el criterio de los usuarios a la hora de su uso, es por ello que se necesita lograr un diseño sencillo y amigable; para lograr estos aspectos se tiene en cuenta la tipografía, navegación y colores. A continuación se muestran algunas características de la interfaz diseñada:

Se utilizaron tonalidades azules, por ser suaves y refrescantes, además que brindan una visión de formalidad a la aplicación. El vocabulario se acerca a la realidad del proceso, de manera que los usuarios se sientan familiarizados con los términos que usualmente utilizan. El tipo de letra utilizada es “Arial (12)” lográndose uniformidad en el diseño del sistema.

La navegación del sistema es sencilla, existe una distribución en las páginas que muestran la información necesaria evitando sobrecargas, las cuales varían en dependencia del rol de cada usuario. La entrada de la información se realiza mediante formularios; cada ventana muestra un encabezado, permitiendo orientar a los usuarios sobre que operaciones puede realizar en ella, los mensajes de error aportan la información necesaria sobre que está ocurriendo en el sistema.

Concepción general de la ayuda

Con el objetivo de que el usuario cuente con un apoyo ante cualquier situación en la aplicación, se desarrolló una ayuda disponible en el menú de cada página, capaz de brindarle al usuario una detallada explicación de cómo funciona el sistema, de manera que el usuario se familiarice con la gestión de los datos, el acceso a los informes o reportes y otras funcionalidades que le brinda el sistema. La ayuda contiene un menú interno que contiene todas las funcionalidades del sistema facilitando el acceso a la información requerida.

Tratamiento de errores

Para garantizar la integridad del sistema, se realiza el tratamiento de errores con el fin de garantizar el mínimo de errores, evitando así la posibilidad de entrada de datos erróneos a la aplicación. Para ello se tuvieron en cuenta dos tipos de validaciones debido al tipo de arquitectura utilizado: validaciones del lado del cliente y validaciones del lado del servidor. Las validaciones del lado del cliente se realizarán utilizando JavaScript, debido que permite dar respuesta rápida, puesto que no necesariamente tiene que ir al servidor, en caso de que no pueda responder a las peticiones, entonces accede al servidor, es decir que los Scripts evitan las peticiones innecesarias al servidor.

Seguridad

Para la protección y confiabilidad de los datos, el sistema exigirá usuario y contraseña para acceder al mismo, que definen el nivel de acceso del usuario logeado. Se mantendrá una traza que permita saber cuándo entraron los usuarios al sistema y a que funcionalidades accedieron. La información almacenada debe ser consistente, responder al usuario que la registrara y se utilizaran validaciones que limitarán la entrada de datos erróneos.

Conclusiones

En el presente capítulo se mostraron los resultados de la etapa de diseño e implementación del sistema, se desarrollaron los diagramas de clases, el diseño de la base de datos y el modelo de despliegue y el diagrama de componentes. Se describieron una serie de estándares a seguir para mantener la integridad de la aplicación, tales como el diseño de la interfaz, la concepción del tratamiento de errores y de la ayuda, los principios generales de diseño, el formato de los reportes y la seguridad de la aplicación.

Conclusiones Generales:

Con la realización del presente proyecto se arribó a las siguientes conclusiones:

1. El estudio de los fundamentos teóricos y metodológicos para la elaboración de un software que facilite la gestión de la información en el BUS permitió determinar que la metodología RUP es la adecuada para el análisis, diseño, implementación y documentación del sistema, lo cual posibilitó la adecuada argumentación de la solución propuesta.
2. Se diseñó un software capaz de automatizar la gestión de la información para el BUS de la Universidad de Sancti Spíritus “José Martí Pérez” sustentado en las bases del software libre, utilizándose RUP como metodología y UML como lenguaje de modelado; además de la programación orientada a objeto y la arquitectura en tres capas lo cual le brinda al sistema una mayor flexibilidad y capacidad de mantenimiento.
3. Se implementó una aplicación web utilizando el framework Symfony; como lenguaje de programación PHP y para la interfaz de usuario HTML, además del uso de JavaScript para las validaciones de entradas del usuario y CSS para el estilo y apariencia. Como herramientas de desarrollo Macromedia Dreamweaver y NetBeans para la programación Web, además de MySQL como SGBD.

Recomendaciones:

1. Agregar nuevas funcionalidades al sistema para propiciar su desarrollo y evolución.

Bibliografía

- Alonso, A., & Hernández, F. (2006). Sistema Gestor de Encuestas. Cienfuegos: Trabajo de diploma para optar por el título de Ingeniería en Informática. *Universidad de Cienfuegos Carlos Rafael Rodríguez*.
- Alonso, L. C. (2006). Módulo Admisión del Sistema del Sistema Automatizado para la gestión de Información de la Misión Milagro. *Ciudad de la Habana*.
- Álvarez, M. A. (2009). *CodeIgniter*. Recuperado el 9 de Enero de 2013, de <http://www.desarrolloweb.com/articulos/codeigniter.html>
- Annan, K. (2003). Discurso Inaugural de la WSIS.
- Cadenas, Y. (2010). Aplicación Web para la gestión de la información en las Oficinas de Seguridad para las Redes Informáticas. *Sancti-Spíritus: Trabajo de Diploma para optar por el título de Ingeniería en Informática, Facultad de Ingeniería, Universidad de Sancti-Spíritus José Martí Pérez*.
- Carral, J. A. (1998). Internet, origen, presente y futuro. *España*.
- Castells, M. (2009). Internet y la sociedad red. *Lección inaugural del programa de doctorado sobre la sociedad de la información y el conocimiento*.
- Castro R, F. (2004). Segundo Congreso FEU 1980. Referencia realizada en el discurso de clausura de Universidad 2004. *La Habana. Cuba*.
- De la Iglesia, J. (2012). La web. La responsabilidad social universitaria 2.0.
- Deleuze, G. (2012). *Escomberoides*. Recuperado el 10 de Febrero de 2013, de <http://www.escomberoides.blogspot.com>
- Eguiluz, J. (2012). Desarrollo web con Symfony 2.
- Fernández, F., López, J., & Vallejo, M. (2008). Infraestructura de comunicaciones para la creación, modelado y gestión de servicios y redes para el hogar.
- García Ros, R., & Pérez González, F. (2009). Una aplicación web para la identificación de estudiantes de nuevo acceso a la Universitat en situación de "riesgo académico"(repertorios estratégicos y gestión del tiempo). *@ tic. revista d'innovació educativa*, 10-17.
- González, A. (2005). Modelamiento del negocio. *Centro de Estudios de Ingeniería de Sistemas (CEIS)*.
- Guitert i Catasús, M., Romeu, T., & Pérez-Mateo, M. (2007). Competencias TIC y trabajo en equipo en entornos virtuales. *RUSC: revista de universidad y sociedad del conocimiento*.

- Gutiérrez, P. P. (2009). Desarrollo agil con Yupp PHP Framework A/C.
- Harasim, L. M. (1995). Learning Networks: A field guide to teaching and learning online. *The MIT Press*.
- Jacobson, I., Booch, G., & Rumbaugh, J. (2000). El proceso unificado de desarrollo. *Madrid: Addison Wesley*.
- Lago, R. (2007). Patrones de diseño software.
- Mato García, R. M. (2006). Sistema de Base de Datos. *La Habana: Félix Varela*.
- Montuschi, L. (2012). *Ideas*. Recuperado el 16 de mayo de 2013, de <http://www.ideas.repec.org>
- Mora, S. L. (2002). Programación de aplicaciones web: historia, principios básicos y clientes web. *Editorial Club Universitario*.
- Netbeans. (2009). Recuperado el 14 de Mayo de 2013, de <http://www.netbeans.org>
- Noreña, M. (2013). *Fovissste*. Recuperado el 9 de Marzo de 2013, de <http://www.fovissste.gob.mx>
- Olmedo Ramos, J. (2001). Lengua, Cultura y Escritura en la Sociedad Virtual: tres décadas de redacción (1971-2001). *Círculo de Lingüística aplicada a la comunicación*.
- PCC. (2000). *Presentación para los Comités Provinciales del PCC, Dirección de Informatización de la Sociedad.MIC*.
- PCC. (2011). *Resolución del VI Congreso del PCC. Lineamientos de la política económica y social del Partido y la Revolución*.
- Project Team, D. (2012). Doctrine 2 ORM Documentation Release 2.1.
- Pulgarín Guerrero, A., & Muñoz Cañavate, A. (2009). La Gestión de la Información en las empresas de Extramadura. *En: Revista española de documentación científica Vol32, No 4, 107-132*.
- Rodríguez, M., & Alejo, O. J. (2006). Portal de Psicología de la Universidad de Cienfuegos. *Trabajo de Diploma para optar por título de Ingeniería en Informática. Universidad de Cienfuegos Carlos Rafael Rodríguez*.
- Rojas Mesa, Y. (2006). De la gestión de información a la gestión del conocimiento. *Acimed*.
- Ruiz, M. H. (2006). Programación Web avanzada Soluciones rápidas y efectivas para desarrolladores de sitios. *La Habana: Felix Varela*.
- Rumbaugh, J., Booch, G., & Jacobson, I. (2006). El proceso unificado de desarrollo de software. *La Habana: Félix Varela*.
- SNTECD. (2011). *Sindicato nacional de la Educación, la Ciencia y el Deporte*. Recuperado el 13 de 5 de 2013, de <http://www.sindicatoeducación.cu>

- Stallings, W. (2012). *Comunicaciones y redes de computadoras. Pearson educación.*
- System, P. S. (2005). *Modelado de sistemas con UML*. Recuperado el 27 de Enero de 2013, de <http://es.tldp.org/tutoriales/doc-modelado-sistemas-UML-multiple-html/c124.html>
- Tedei, L. (2009). *Frameworks en PHP y otros lenguajes.*
- Valle Fasco, M. (2007). *La Universidad Territorial Universalizada Sustentable. Tesis en opción al grado de Doctor en Ciencias Pedagógicas. Universidad de Oriente.*
- Vedora, U. (2006). *Sistema de gestión de la información de Estudiantes Becarios Extranjeros (SIBEX). Trabajo de Diploma para optar por el título de Ingeniería. Universidad de Cienfuegos Carlos Rafael Rodríguez.*
- Vidal, C. (2006). *Sistema de Envío de Correos por Lote(SECL). Trabajo de Diploma para optar por título de Ingeniería en Informática. Universidad de Cienfuegos Carlos Rafael Rodríguez.*

Anexos:

Anexo 1 Descripción de los casos de uso paquete Seguridad

Caso de uso	Autenticación
Actores:	Usuario
Propósito	Controlar el acceso al software.
Resumen:	comienza cuando el usuario solicita la entrada al software al introducir sus datos de usuario, si está registrado y sus datos son correctos tiene acceso al software con determinados permisos, en caso contrario se le informa a través de un mensaje. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.
Referencias	R1
Prototipos	Anexo 5

Anexo 1.1 Descripción del caso de uso “Iniciar sesión”

Caso de uso	Cerrar Sesión
Actores:	Usuario
Propósito	Culminar las acciones en el sistema.
Resumen:	El caso de uso se inicia y culmina cuando el usuario termina de trabajar en el sistema.
Referencias	R2
Prototipos	Anexo 6

Anexo 1.2 Descripción del caso de uso “Cerrar Sesión”

Caso de uso	Cambiar contraseña
Actores:	Usuario
Propósito	Mantener la seguridad del sistema y la confiabilidad de los datos al ser la contraseña del usuario únicamente de su conocimiento.
Resumen: El caso de uso comienza cuando el usuario solicita el cambio de su contraseña, introduce los datos pertinentes y de ser correctos el sistema guarda la nueva contraseña, en caso contrario emite un mensaje de error. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.	
Referencias	R3
Prototipos	Anexo 7

Anexo 1.3: Descripción del caso de uso “Cambiar Contraseña”

Anexo 2 Descripción de los casos de uso paquete Nomencladores

Caso de uso	Gestionar usuario
Actores:	Administrador
Propósito	Tener controlado el acceso al software a través de usuarios con determinados privilegios.
Resumen: comienza cuando el administrador accede al menú gestionar usuarios para introducir un nuevo usuario o para modificar o eliminar uno existente, luego introduce los datos pertinentes, de ser correctos el sistema ejecuta la acción, en caso contrario emite un mensaje de error. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.	
Referencias	R4
Prototipos	Anexo 8

Anexo 2.1: Descripción del caso de uso “Gestionar usuario”

Caso de uso	Gestionar Noticia
Actores:	Administrador
Propósito	Gestionar las diferentes noticias del mes en el sistema.
Resumen:	El caso de uso se inicia cuando el Administrador desea gestionar (insertar, eliminar, modificar) alguna noticia para el uso del usuario en el sistema.
Referencias	R5
Prototipos	Anexo 9

Anexo 2.2: Descripción del caso de uso “Gestionar Noticia”

Caso de uso	Subir documento
Actores:	Administrador
Propósito	Subir archivos de interés referentes al tema del BUS.
Resumen:	El Caso de uso se inicia cuando el administrador accede al sistema para subir archivos. Para subir la información, se debe seleccionar el lugar donde se encuentra y se carga. El caso de uso finaliza cuando se ejecuta la operación antes descrita
Referencias	R6
Prototipos	Anexo 10

Anexo 2.3: Descripción del caso de uso “Subir documentación”

Caso de uso	Gestionar Integrantes
Actores:	Administrador
Propósito	Tener controlado todas las personas que pertenecen al BUS.
Resumen: comienza cuando el administrador accede al menú gestionar integrante para introducir un nuevo integrante o para modificar o eliminar uno existente, luego introduce los datos pertinentes, de ser correctos el sistema ejecuta la acción, en caso contrario emite un mensaje de error. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.	
Referencias	R7
Prototipos	Anexo 11

Anexo 2.4: Descripción del caso de uso “Gestionar Integrantes”

Caso de uso	Gestionar Sección
Actores:	Administrador
Propósito	Tener controlado todas las secciones que pertenecen al BUS.
Resumen: comienza cuando el administrador accede al menú gestionar sección para introducir un nueva sección o para modificar o eliminar una existente, luego introduce los datos pertinentes, de ser correctos el sistema ejecuta la acción, en caso contrario emite un mensaje de error. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.	
Referencias	R8
Prototipos	Anexo 12

Anexo 2.5: Descripción del caso de uso “Gestionar Sección”

Caso de uso	Gestionar Cotización
Actores:	Administrador, Cotización
Propósito	Tener controlado todas las personas que hayan pagado la cotización de determinado mes.
Resumen: comienza cuando el administrador o cotización accede al menú gestionar cotización para introducir un nuevo pago de cotización o para modificar o eliminar una existente, luego introduce los datos pertinentes, de ser correctos el sistema ejecuta la acción, en caso contrario emite un mensaje de error. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.	
Referencias	R9
Prototipos	Anexo 13

Anexo 2.6: Descripción del caso de uso “Gestionar Cotización”

Caso de uso	Gestionar Día de Haber
Actores:	Administrador, Cotización
Propósito	Tener controlado todas las personas que hayan pagado el Día de Haber
Resumen: comienza cuando el administrador o cotización accede al menú gestionar Día de Haber para introducir un nuevo pago de Día de Haber o para modificar o eliminar una existente, luego introduce los datos pertinentes, de ser correctos el sistema ejecuta la acción, en caso contrario emite un mensaje de error. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.	
Referencias	R10
Prototipos	Anexo 14

Anexo 2.7: Descripción del caso de uso “Gestionar Día de Haber”

Caso de uso	Gestionar Roles
Actores:	Administrador
Propósito	Tener controlado el acceso al software a través de usuarios con determinados privilegios.
Resumen: comienza cuando el administrador accede al menú gestionar roles para introducir un nuevo rol o para modificar o eliminar uno existente, luego introduce los datos pertinentes, de ser correctos el sistema ejecuta la acción, en caso contrario emite un mensaje de error. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.	
Referencias	R11
Prototipos	Anexo 15

Anexo 2.8: Descripción del caso de uso “Gestionar Roles”

Caso de uso	Gestionar Área de Trabajo
Actores:	Administrador
Propósito	Tener controlado todas las Áreas de Trabajo que pertenecen al BUS.
Resumen: comienza cuando el administrador accede al menú gestionar área de trabajo para introducir una nueva área de trabajo o para modificar o eliminar una existente, luego introduce los datos pertinentes, de ser correctos el sistema ejecuta la acción, en caso contrario emite un mensaje de error. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.	
Referencias	R12
Prototipos	Anexo 16

Anexo 2.9: Descripción del caso de uso “Gestionar área de trabajo”

Caso de uso	Gestionar Efemérides
Actores:	Administrador
Propósito	Gestionar algún acontecimiento importante ocurrido en la historia para que esté presente en el sistema.
Resumen: comienza cuando el administrador accede al menú gestionar efemérides para introducir una nueva efeméride o para modificar o eliminar una existente, luego introduce los datos pertinentes, de ser correctos el sistema ejecuta la acción, en caso contrario emite un mensaje de error. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.	
Referencias	R13
Prototipos	Anexo 17

Anexo 2.10: Descripción del caso de uso “Gestionar Efemérides”

Caso de uso	Gestionar Destacados
Actores:	Administrador
Propósito	Tener controlado todos los destacados que hay en un mes que pertenecen al BUS.
Resumen: comienza cuando el administrador accede al menú gestionar destacado para introducir un nuevo destacado o para modificar o eliminar una existente, luego introduce los datos pertinentes, de ser correctos el sistema ejecuta la acción, en caso contrario emite un mensaje de error. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.	
Referencias	R14
Prototipos	Anexo 18

Anexo 2.11: Descripción del caso de uso “Gestionar Destacado”

Anexo 3 Descripción de los casos de uso paquete Reporte

Caso de uso	Descargar Documento
Actores:	Usuario
Propósito	Que el usuario pueda descargar los documentos existentes.
Resumen: comienza cuando el usuario accede a la lista de documentos y selecciona uno para descargarlo. El caso de uso finaliza cuando se ejecuta la operación antes descritas.	
Referencias	R15
Prototipos	Anexo 19

Anexo 3.1: Descripción del caso de uso “Descargar Documento”

Caso de uso	Reportar Sección
Actores:	Usuario
Propósito	Tener conocimiento de las secciones que pertenecen al BUS.
Resumen: comienza cuando el usuario accede al menú sección, el sistema genera el reporte y en caso de existir datos los muestra, en caso contrario muestra un mensaje comunicando la información.	
Referencias	R16
Prototipos	Anexo 20

Anexo 3.2: Descripción del caso de uso “Reportar Sección”

Caso de uso	Reportar Integrantes por Sección
Actores:	Usuario
Propósito	Tener conocimiento de los integrantes de cada sección.
Resumen: comienza cuando el usuario accede al menú sección y selecciona integrantes por sección, el sistema genera el reporte y en caso de existir datos lo muestra, en caso contrario muestra un mensaje comunicando la información.	
Referencias	R17
Prototipos	Anexo 21

Anexo 3.3: Descripción del caso de uso “Reportar Integrantes por Sección”

Caso de uso	Reportar Usuarios
Actores:	Usuario
Propósito	Tener conocimiento de los usuarios de la aplicación.
Resumen: comienza cuando el usuario accede al menú usuario, el sistema genera el reporte y en caso de existir datos lo muestra, en caso contrario muestra un mensaje comunicando la información.	
Referencias	R18
Prototipos	Anexo 22

Anexo 3.4: Descripción del caso de uso “Reportar Usuarios”

Caso de uso	Reportar Integrantes que faltan por pagar por mes la cotización
Actores:	Usuario
Propósito	Tener conocimiento de los Integrantes que faltan por pagar la cotización.
Resumen: comienza cuando el usuario accede al menú cotización y selecciona el reporte, el sistema lo genera y en caso de existir datos lo muestra, en caso contrario muestra un mensaje comunicando la información.	
Referencias	R19
Prototipos	Anexo 23

Anexo 3.5: Descripción del caso de uso “Reportar Integrantes que faltan por pagar por mes la cotización”

Caso de uso	Reportar Integrantes que faltan por pagar Día de Haber
Actores:	Usuario
Propósito	Tener conocimiento de los Integrantes que faltan por pagar el Día de Haber.
Resumen: comienza cuando el usuario accede al menú Día de Haber y selecciona el reporte, el sistema lo genera y en caso de existir datos lo muestra, en caso contrario muestra un mensaje comunicando la información.	
Referencias	R20
Prototipos	Anexo 24

Anexo 3.6: Descripción del caso de uso “Reportar Integrantes que faltan por pagar Día de Haber”

Caso de uso	Reportar Jefes de Sección
Actores:	Usuario
Propósito	Tener conocimiento de los jefes de cada sección.
Resumen: comienza cuando el usuario accede al menú Sección y selecciona el reporte, el sistema lo genera y en caso de existir datos lo muestra, en caso contrario muestra un mensaje comunicando la información.	
Referencias	R21
Prototipos	Anexo 25

Anexo 3.7: Descripción del caso de uso “Reportar Jefes de Sección”

Caso de uso	Reportar Cumpleaños por mes
Actores:	Usuario
Propósito	Tener conocimiento de los cumpleaños de cada integrante.
Resumen: comienza cuando el usuario accede al menú Cumpleaños, el sistema lo genera y en caso de existir datos lo muestra, en caso contrario muestra un mensaje comunicando la información.	
Referencias	R22
Prototipos	Anexo 26

Anexo 3.8: Descripción del caso de uso “Reportar Cumpleaños por mes”

Caso de uso	Reportar Efemérides
Actores:	Usuario
Propósito	Tener conocimiento de las efemérides.
Resumen: comienza cuando el usuario accede al menú efemérides, el sistema lo genera y en caso de existir datos lo muestra, en caso contrario muestra un mensaje comunicando la información.	
Referencias	R23
Prototipos	Anexo 27

Anexo 3.9: Descripción del caso de uso “Reportar efemérides”

Caso de uso	Reportar listado de documentos
Actores:	Usuario
Propósito	Tener conocimiento de cada documento.
Resumen: comienza cuando el usuario accede al menú documentación, el sistema lo genera y en caso de existir datos lo muestra, en caso contrario muestra un mensaje comunicando la información.	
Referencias	R24
Prototipos	Anexo 28

Anexo 3.10: Descripción del caso de uso “Reportar listado de documentos”

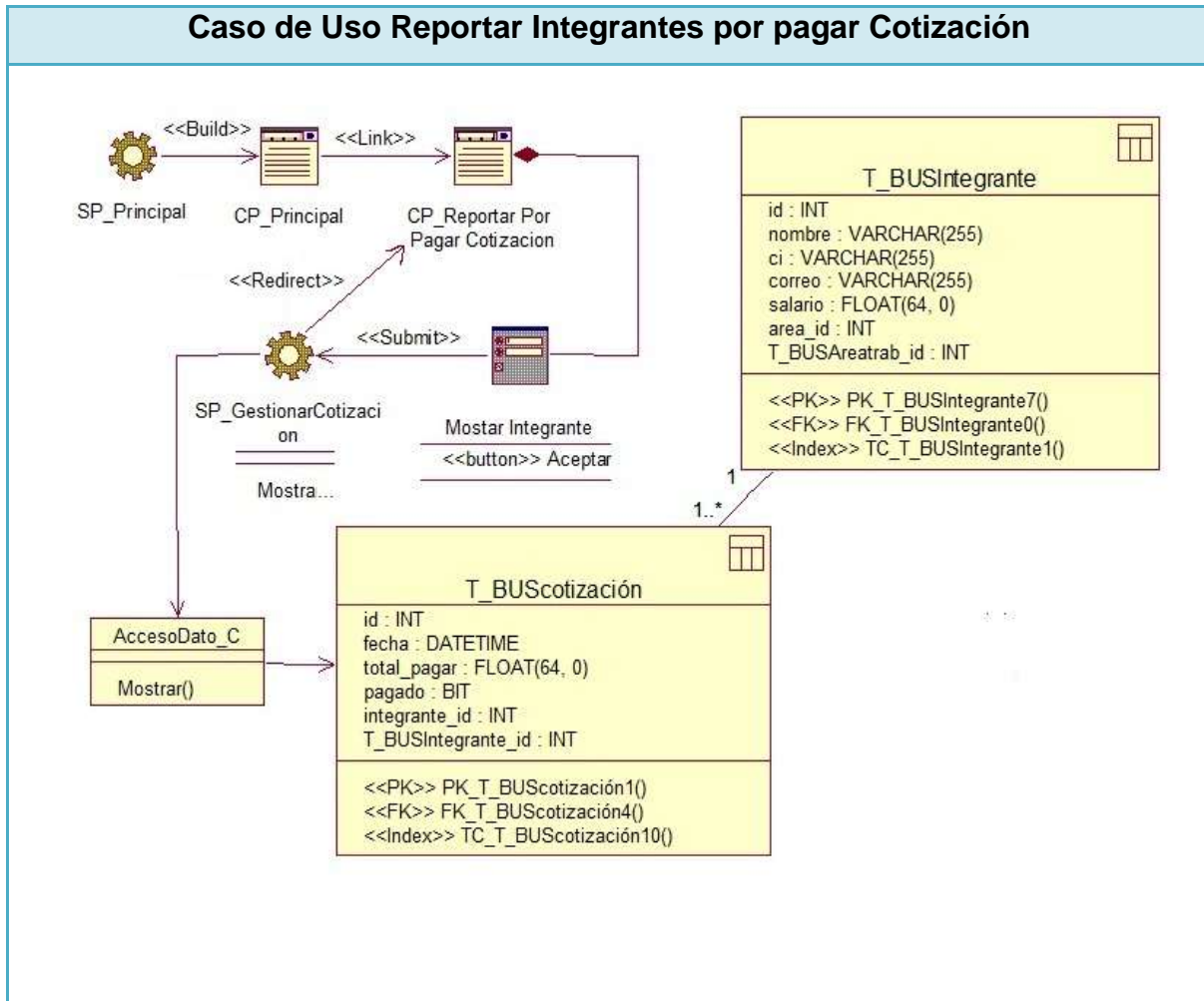
Caso de uso	Mostrar Noticia
Actores:	Usuario
Propósito	Tener información sobre la noticia.
Resumen: comienza cuando el usuario accede al menú noticia y selecciona la dekada, el sistema la genera y en caso de existir datos lo muestra, en caso contrario muestra un mensaje comunicando la información.	
Referencias	R25
Prototipos	Anexo 29

Anexo 3.11: Descripción del caso de uso “Mostrar Noticia”

Caso de uso	Reportar destacados por mes
Actores:	Usuario
Propósito	Tener información sobre los destacados de cada mes.
Resumen: comienza cuando el usuario accede al menú destacados y selecciona un mes, el sistema la genera y en caso de existir datos lo muestra, en caso contrario muestra un mensaje comunicando la información.	
Referencias	R26
Prototipos	Anexo 30

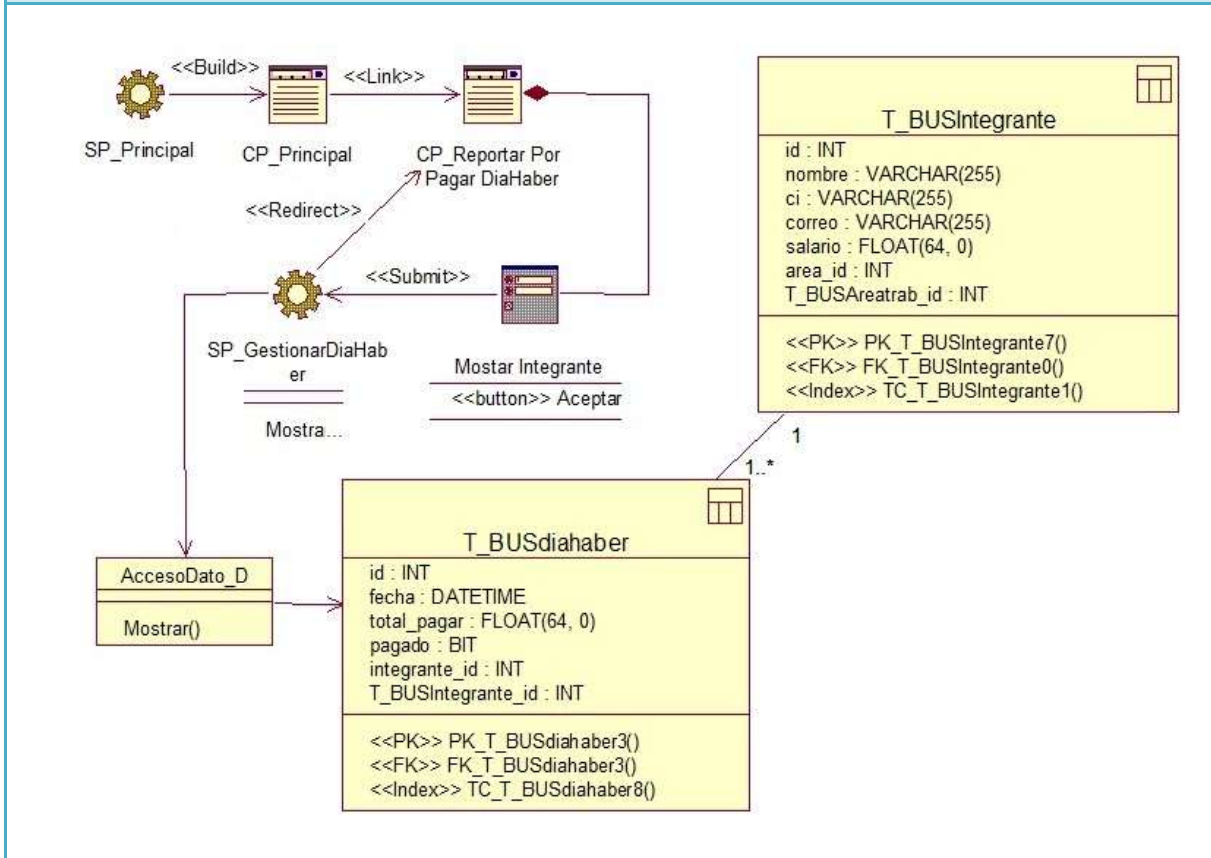
Anexo 3.12: Descripción del caso de uso “Reportar destacados por mes”

Anexo 4 Diagrama de clases Web



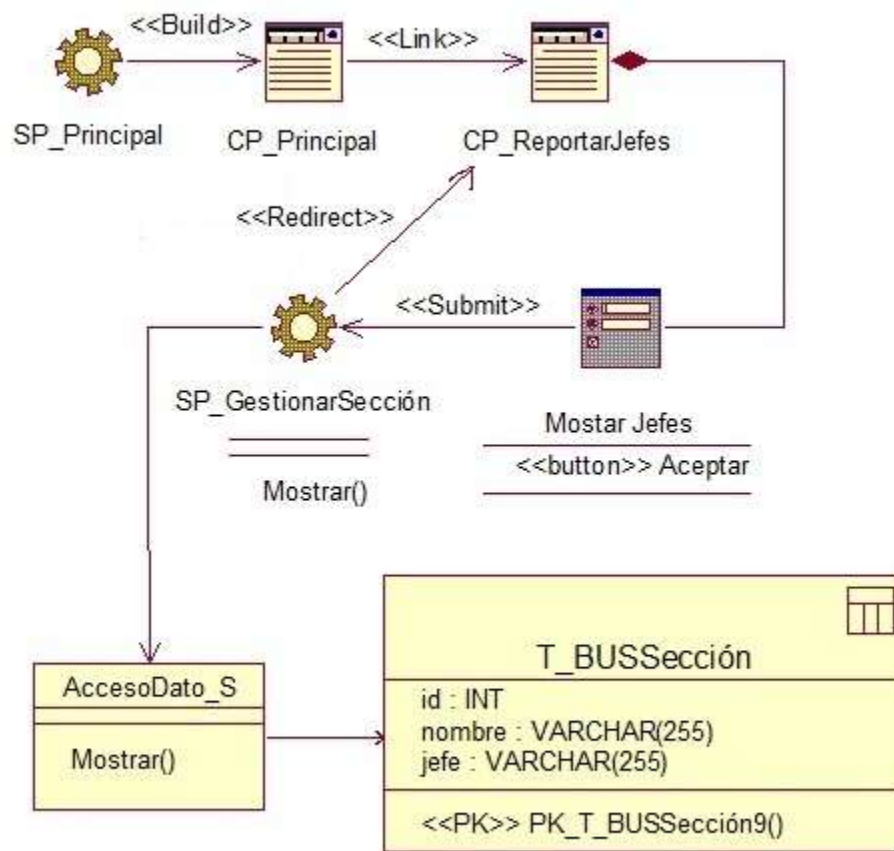
Anexo 4.1: Diagrama de clases Web Caso de Uso Reportar Integrantes por pagar cotización

Caso de Uso Reportar Integrantes por pagar Día de Haber



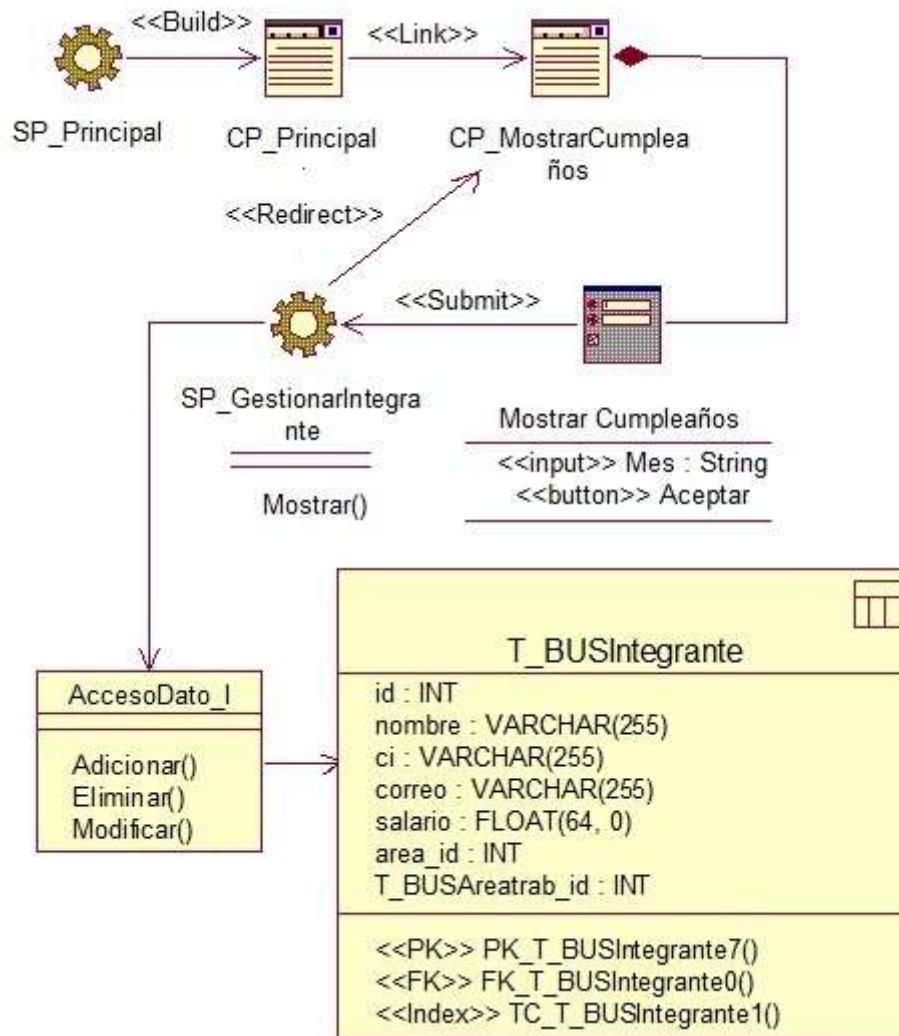
Anexo 5.2: Diagrama de clases Web Caso de Uso Reportar Integrantes por pagar día de haber

Caso de Uso Reportar Jefe Sección



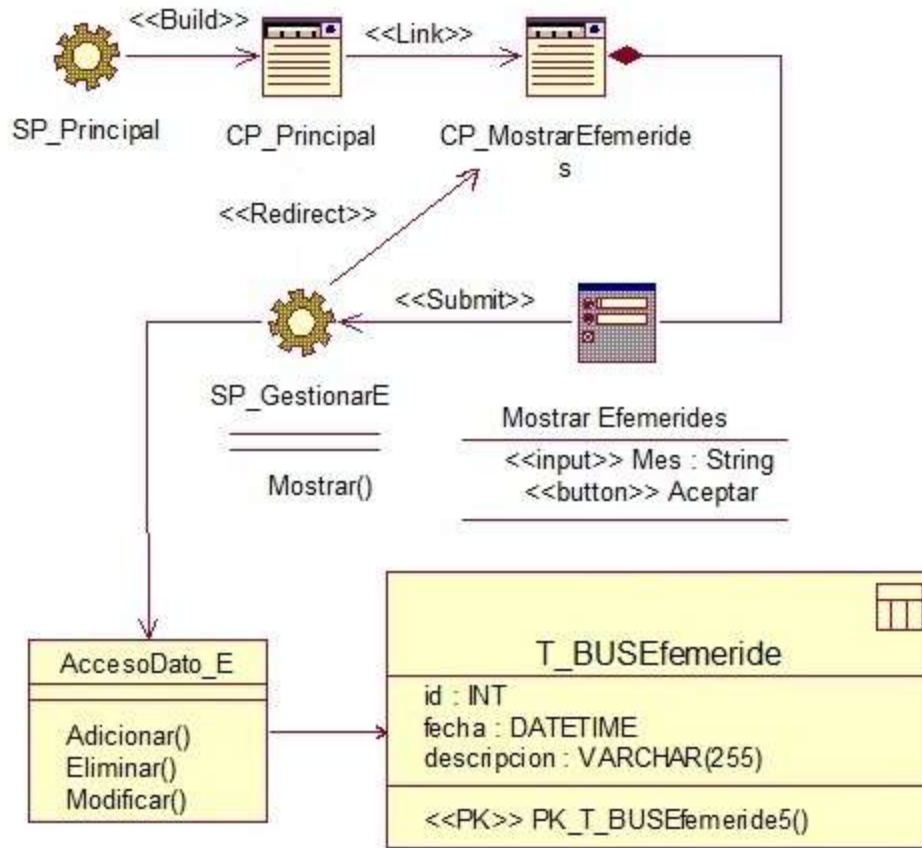
Anexo 6.3: Diagrama de clases Web Caso de Uso Reportar Jefe Sección

Caso de Uso Reportar Cumpleaños por mes



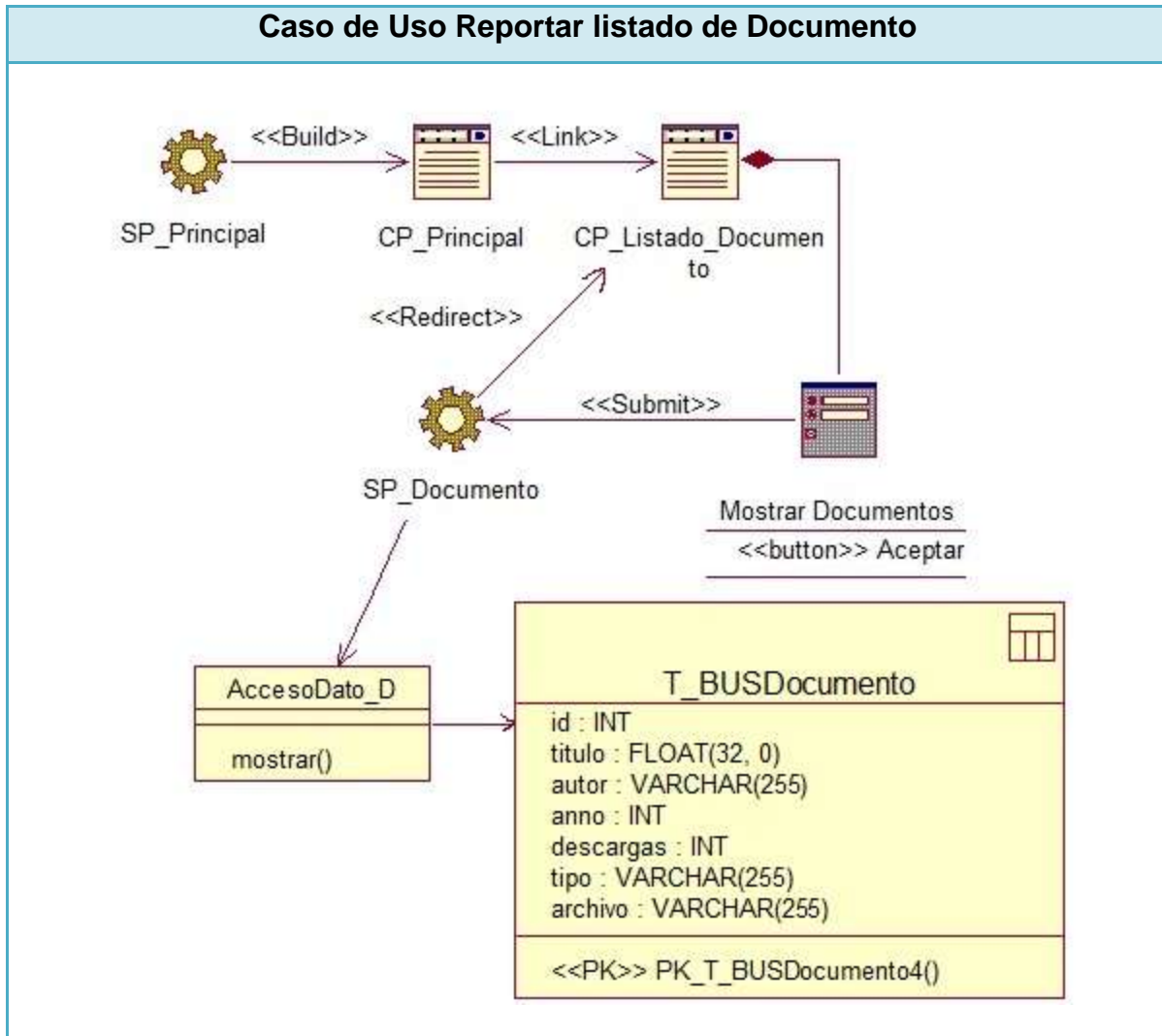
Anexo 7.4: Diagrama de clases Web Caso de Uso Reportar Cumpleaños por mes

Caso de Uso Reportar efemérides

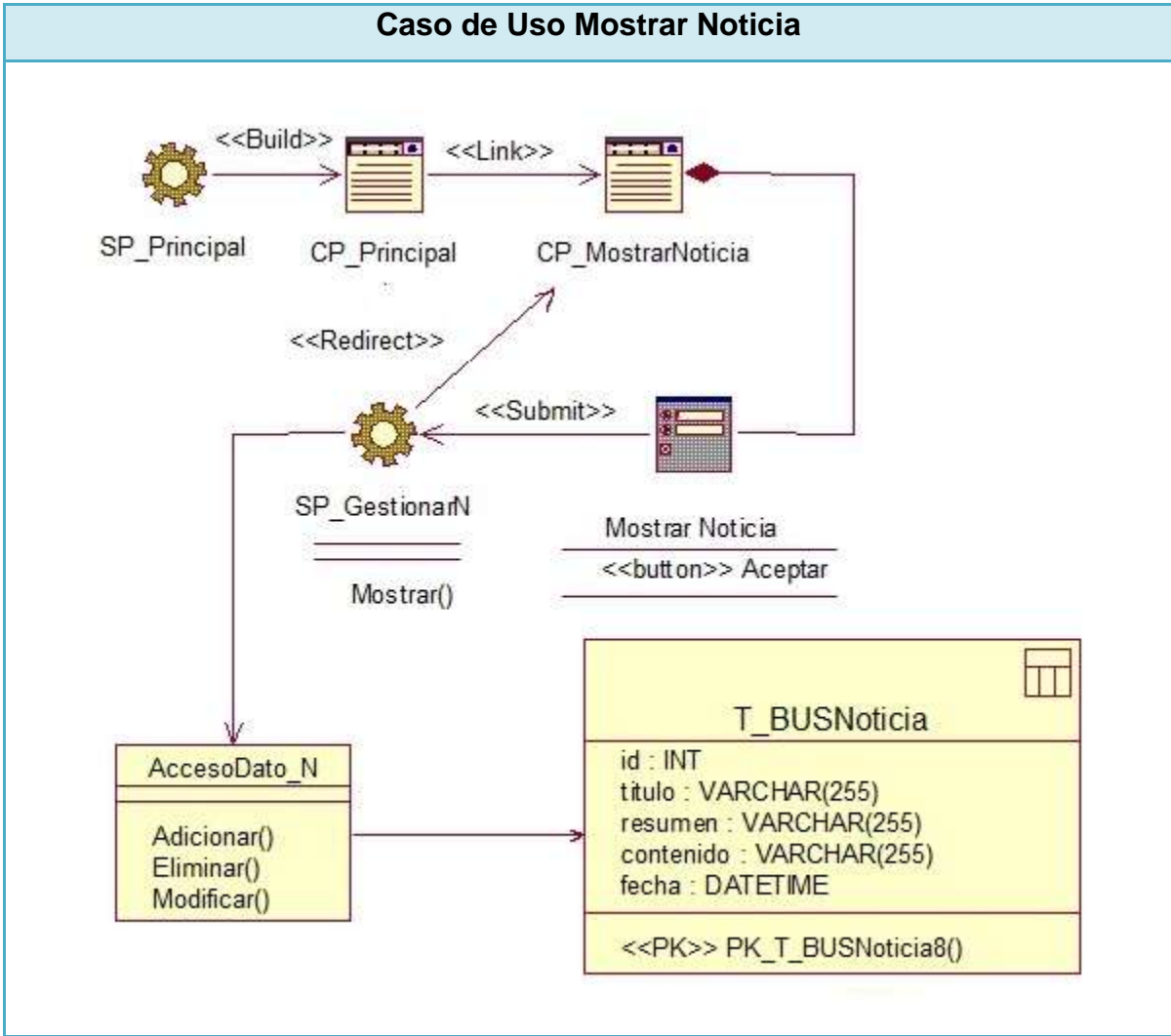


Anexo 8.5: Diagrama de clases Web Caso de Uso Reportar efemérides

Caso de Uso Reportar listado de Documento

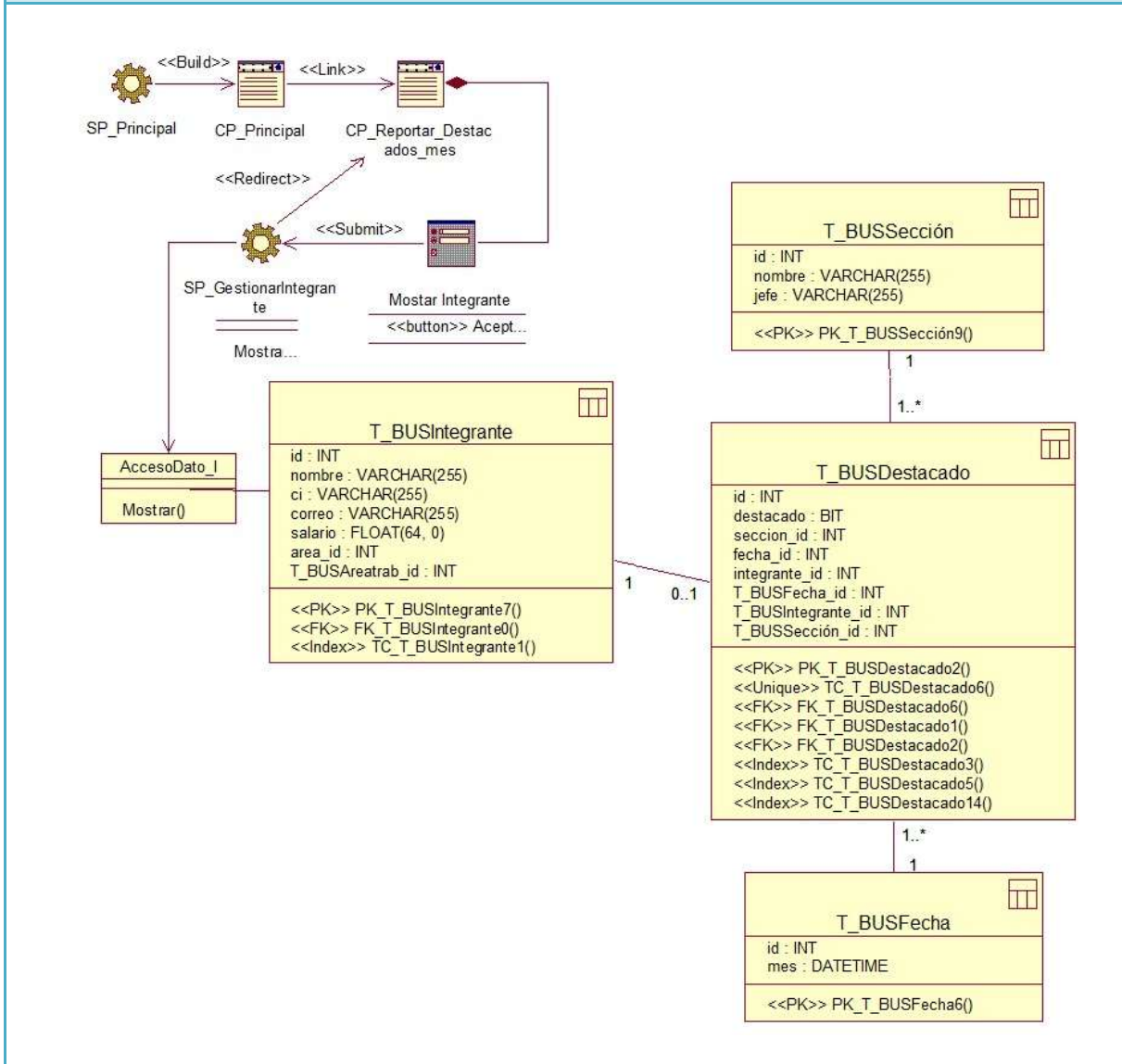


Anexo 9.6: Diagrama de clases Web Caso de Uso Reportar listado de documento



Anexo 10.7: Diagrama de clases Web Caso de Uso Mostrar Noticia

Caso de Uso Reportar Destacados por mes



Anexo 11.8: Diagrama de clases Web Caso de Uso Reportar Destacados por mes

Anexo 5 Prototipo caso de uso: Autenticar usuario

Formulario de acceso

Usuario

Contraseña

Entrar

Anexo 6 Prototipo caso de uso: Cerrar Sesión



Juan Antonio Mesas Jimenez

Salir

Anexo 7 Prototipo caso de uso: Cambiar Contraseña

Editar usuario

Nombre

orlando

Usuario

orlando

Correo

orlando@suss.co.cu

Contraseña




Repita la contraseña

Administrador

Editar usuario

Anexo 8 Prototipo caso de uso: Gestionar usuario

Listado de usuarios

Nombre	Usuario	Correo	Tipo de Usuario
1 orlando	orlando	orlando@suss.co.cu	 
			

Anexo 9 Prototipo caso de uso: Gestionar Noticia

Título	Fecha	
1 <u>1 de Mayo</u>	2014-05-01	 
		

Anexo 10 Prototipo caso de uso: Subir documento

Insertar un Documento

Título

Autor

Año

Tipo de Archivo

Descargas

Archivo






Anexo 11 Prototipo caso de uso: Gestionar Integrantes

Nombre		Correo	
1	<u>Pepe Hernández Miel</u>	pepe@suss.co.cu	 
			




Anexo 12 Prototipo caso de uso: Gestionar Sección

Nombre Sección		Reportes	
1	<u>Rectorado</u>	<u>Integrantes de Sección</u>	 
2	<u>Vicerectorado</u>	<u>Integrantes de Sección</u>	 
			

Anexo 16 Prototipo caso de uso: Gestionar Área de Trabajo

Área de Trabajo		
1	<u>Departamento Defensa</u>	 
2	<u>Administración Docente 2</u>	 
		

Anexo 19 *Prototipo caso de uso: Descargar Documento*

Título	Autor	Tipo de Archivo	
1 Acta del Sindicato	BUS	Acta	 
			

Anexo 21 *Prototipo caso de uso: Reportar Integrantes por Sección*

Nombre de Integrante
1 Pepe Hernández Miel

Los Prototipos del Anexo 20 al 28, incluyendo el anexo 30 tienen el mismo aspecto que el mostrado anteriormente (anexo 21)

Anexo 29 *Prototipo caso de uso: Reportar Integrantes por Sección*

1 de Mayo

Día de los Trabajadores

Este es un día muy especial para los trabajadores cada año

Anexo 31 Prototipo Página Principal

BUS

Buro Universitario Sindical

[Inicio](#) [Nuestras Secciones](#) [ANUR](#) [Documentación](#) [Noticias](#) [Enlaces Útiles](#) [Reportes](#)

UNISS
UNIVERSIDAD
SANCTI SPIRITUS



Buró Universitario Sindical (BUS)

Es un órgano de representación y coordinación sindical, electo cada 2 años y medio. Sus miembros salen de las propuestas realizadas en las 10 secciones sindicales de la Universidad de Sancti Spiritus "José Martí Pérez" (Uniss). El secretario del BUS representa a los trabajadores ante el órgano de dirección de la Uniss y otras instancias administrativas y sindicales. El BUS está encargado de orientar, coordinar el trabajo de las secciones sindicales así como asesorar y capacitar sus dirigentes.

BLOQUE ADMINISTRACIÓN

[AUTENTICARSE](#)

Últimas Noticias

1 de Mayo

2014-05-01 | 0 Visitas



UNIVERSIDAD DE SANCTI SPIRITUS "JOSÉ MARTÍ PÉREZ" ©