

Universidad de las Ciencias Informáticas

Facultad 10.



“Automatización del Servicio de Impresión de la Dirección de Información.”

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

Autores: Randy Llanes Alfonso
Yunel Cabrera García

Tutor: Ing. Lityuen A. Porras Herrera

Ciudad de la Habana 10 de junio del 2009
“Año del 50 Aniversario de la Revolución

DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter no exclusivo.

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

Randy Llanes Alfonso

Yunel Cabrera García

Ing. Lityuen A. Porras Herrera

DATOS DEL CONTACTO

Ing. Lityuen Amalia Porras Herrera

Graduada de Ingeniería en Ciencias Informáticas en la Universidad de la Ciencias Informáticas (UCI) en el año 2008. Actualmente forma parte del departamento de Ingeniería y Gestión de Software de la facultad 10 como responsable de la disciplina de Práctica Profesional 2. Pertenece al grupo de Informatización de la Dirección de Información del proyecto Gestión Bibliotecaria perteneciente al polo de la Gestión de la Información y el Conocimiento de la facultad 10, donde desempeña el rol de Analista.

AGRADECIMIENTOS

Randy Llanes Alfonso

A mis compañeros y amigos de la UCI, que siempre me han ayudado a lo largo de toda la carrera.

A mi tutora Lityuen y a Kenia, que son las principales responsables de la realización de este trabajo, apoyándome siempre que necesité su ayuda...

A todos los profesores de los que de una forma u otra han contribuido a mi formación profesional.

A mis padres, mis abuelos, a mi tía Raida, en fin a toda mi familia por apoyarme siempre y confiar en que algún día llegara a ser ingeniero.

Yunel Cabrera García

A mis padres Reina García Piña y German Cabrera Pérez por brindarme siempre su apoyo, y especialmente a mi novia Maidi Leivi Gutiérrez Martínez, por estar siempre ahí para mí.

DEDICATORIA

Randy Llanes Alfonso

Dedico este trabajo a toda mi familia, pero muy especialmente a mi mamá y a mi abuela Elsa, que desde pequeño siempre velaron por mi futuro y mis estudios, que confiaron en mí y siempre me apoyaron, y porque sin ellas, nunca hubiera podido alcanzar las metas que he logrado.

Yunel Cabrera García

Dedico los resultados de este trabajo a mi madre Reina García Piña y especialmente a mi padre German Cabrera y a mi novia Maidi Leivi Gutiérrez Martínez, pues ellos siempre fueron mi inspiración durante la realización de este trabajo.

RESUMEN

El ser humano, en su afán de adquirir nuevos conocimientos ha tenido la necesidad de guardar de alguna manera las investigaciones y estudios que ha realizado.

Primero de forma manuscrita, luego con el surgimiento de la imprenta se ha logrado recoger, la historia y desarrollo de la humanidad.

En sus inicios, la imprenta, trabajaba el texto hueco sobre una tablilla de madera incluyendo dibujos confeccionados por artesanos. Estos se acoplaban a una mesa de trabajo, también de madera, y se impregnaban de tres colores de tinta que eran los que existían entonces: negro, rojo o azul. Después se aplicaba el papel y con rodillo se fijaba la tinta.

El decursar del tiempo y el auge de las nuevas tecnologías, trae aparejada la imprenta digital, que ha propiciado notables avances en el servicio de impresión a través de la red electrónica.

Este tipo de impresión proporciona diversas ventajas como por ejemplo: el ahorro de tiempo y costos, la posibilidad de imprimir online, enviar pedidos de impresión por correo electrónico, entre otros.

La Dirección de Información de la UCI ha comenzado con un proceso de automatización para lograr un sistema que controle los servicios que se brindan en la biblioteca. Este sistema constará entre otros con un módulo para facilitar la impresión de documentos a los estudiantes.

En el presente trabajo se lleva a cabo el desarrollo de dicho módulo el cual permitirá realizar de forma más eficiente y cómoda la impresión de documentos a los estudiantes, permitiéndoles acceder al servicio desde cualquier PC conectada a la red. Para ello se realizó un estudio profundo de los diferentes servicios de impresión brindados por las distintas bibliotecas y universidades alrededor del mundo y en Cuba, además de estudiar los distintos tipos de herramientas para poder desarrollar el sistema que se desea implementar.

Palabras claves

Dirección de Información, impresión, módulo, biblioteca, impresión digital, impresión en red.

TABLA DE CONTENIDOS

Ing. Lityuen Amalia Porras Herrera	III
Graduada de Ingeniería en Ciencias Informáticas en la Universidad de la Ciencias Informáticas (UCI) en el año 2008. Actualmente forma parte del departamento de Ingeniería y Gestión de Software de la facultad 10 como responsable de la disciplina de Práctica Profesional 2. Pertenece al grupo de Informatización de la Dirección de Información del proyecto Gestión Bibliotecaria perteneciente al polo de la Gestión de la Información y el Conocimiento de la facultad 10, donde desempeña el rol de Analista.....	III
Randy Llanes Alfonso.....	IV
Yunel Cabrera García.....	IV
Randy Llanes Alfonso.....	V
Dedico este trabajo a toda mi familia, pero muy especialmente a mi mamá y a mi abuela Elsa, que desde pequeño siempre velaron por mi futuro y mis estudios, que confiaron en mí y siempre me apoyaron, y porque sin ellas, nunca hubiera podido alcanzar las metas que he logrado.....	V
Yunel Cabrera García.....	V

INTRODUCCIÓN

La necesidad de información es universal y, en estos tiempos, hay más información disponible que nunca. Para que todos los ciudadanos puedan beneficiarse del conocimiento acumulado por la humanidad, esta información debe estar disponible de forma global y gratuita para todos. Para cubrir esa necesidad surge la biblioteca pública, como centro que favorece la difusión de la información y su aprovechamiento por parte de todos los interesados.

La Biblioteca Pública es sobre todo un servicio de información; ofrece información con diversas funciones, a través de diferentes servicios y en diversos soportes y formatos. Su propósito fundamental es satisfacer las necesidades informativas de los individuos y los grupos sociales.

La Biblioteca Pública ofrece la información a través de diferentes medios, que constituyen sus servicios:

Consulta: posibilita a todos los ciudadanos el acceso a los recursos propios y externos, facilitando para ello las instalaciones y equipos necesarios, unas colecciones y servicios de acceso amplio y actualizado, con servicios de orientación y apoyo para su utilización.

Préstamo: ofrece a los usuarios la posibilidad de llevarse en préstamo temporal los materiales de la biblioteca, para utilizarlos en el momento y lugar que consideren más conveniente. Además de para los individuos, la Biblioteca Pública ofrece préstamo en favor de colectivos u organizaciones, facilitándoles el acceso a la información y de esta forma conseguir sus propios fines como grupo.

Información: un servicio que trata de responder a las demandas concretas de información de los ciudadanos. El personal bibliotecario, utilizando su conocimiento profesional y los recursos a los que la biblioteca tiene acceso, es capaz de responder a una multitud y diversidad de cuestiones, desde las que se refieren a autores y obras, para identificarlas y localizarlas, hasta las que piden datos concretos.

Información local: dentro del servicio de información, la Biblioteca Pública, como entidad eminentemente local, debe procurar el mayor nivel de profundidad en lo que se refiere a la información local. Para ello, alberga una colección completa de los materiales informativos en todo tipo de soportes que contengan información relacionada con su localidad, así como establece y mantiene relaciones con todas las instituciones locales que puedan ser generadoras o consumidoras de información de interés local. [1]

La Biblioteca de la Universidad de las Ciencias Informáticas (UCI), que pertenece a la Dirección de Información, brinda una serie de servicios con características particulares que inciden de manera sustancial en el proceso de bibliotecología sobre los estudiantes. Su característica peculiar es que es una biblioteca prácticamente digital, pues la mayoría de la información está almacenada digitalmente. Esto repercute de una manera u otra en la dinámica de los procesos bibliotecarios para el desarrollo de nuevos servicios digitales encaminados a mejorar las condiciones de estudio de los estudiantes.

Uno de los servicios con los que cuenta la Dirección de Información es el de impresión de documentos a estudiantes y profesores de la universidad. Este servicio aún no está automatizado completamente. Actualmente el proceso para imprimir solo está automatizado en el local de impresión, el resto del proceso es necesario hacerlo de forma manual mediante una autorización escrita, que en muchos casos puede no ser confiable. Además en la oficina de impresión de documentos se acumula gran cantidad de estudiantes en espera de recibir el servicio, el cual a veces se torna un poco lento debido a que cada estudiante que desee imprimir debe llevar su documento en un dispositivo de almacenamiento extraíble. Esto puede provocar demora a la hora de copiar dicho documento, ya que cada dispositivo tiene que ser analizado para detectar posibles virus. De esta manera, se hace necesaria la creación de una nueva aplicación realizada totalmente con herramientas libres, pues el sistema actual es un programa con el que solo trabajan las secretarías de la oficina de impresión y tiene solamente funcionalidades para chequear el consumo de hojas mensuales pero además está construido con software propietario.

Dada la situación anterior se plantea el siguiente **problema científico**: ¿Cómo realizar de manera más eficiente el servicio de impresión de documentos en la UCI?, siendo el **objeto de estudio** de esta investigación el proceso de impresión de documentos que se brinda en las diferentes instituciones y teniendo como **campo de acción** el servicio de impresión que brinda la Dirección de Información de la Universidad de las Ciencias informáticas.

Como **Idea a defender** se tiene que si se desarrolla una aplicación Web, basada en un Sistema de Manipulación de Contenidos (CMS), se puede lograr que el servicio de impresión de la Dirección de Información de la UCI sea más eficiente, rápido y totalmente automatizado, permitiendo a los estudiantes una mayor comodidad y rapidez a la hora de imprimir un documento.

El trabajo tiene como **objetivo general** desarrollar una aplicación basada en software libre para facilitar el proceso de impresión de documentos de la Dirección de Información. Entre los **objetivos específicos** se plantean los siguientes:

- Realizar un análisis sobre el estado actual del servicio de impresión de la Dirección de Información.

- Hacer un estudio sobre los Sistemas de Manipulación de Contenidos (CMS) para el desarrollo del sistema.
- Realizar el levantamiento de requisitos para el sistema que se desea desarrollar.
- Realizar el análisis y diseño del sistema para el servicio de impresión.
- Diseñar la interfaz de usuario del módulo del servicio de impresión.
- Implementar un sistema para automatizar el servicio de impresión utilizando el CMS que se adapte al sistema que se desea desarrollar.
- Implantar el sistema en la Dirección de Información.

Con el propósito de guiar, controlar, evaluar y perfilar el trabajo hacia el alcance de los objetivos trazados, se definieron las siguientes **tareas**:

- Estudiar el entorno de trabajo.
- Analizar las tecnologías que se utilizan en el mundo para llevar a cabo sistemas como el que se pretende desarrollar.
- Declarar los requisitos que debe cumplir el módulo.
- Construir diagramas que permitan una mejor comprensión del funcionamiento del módulo.
- Realizar un análisis y diseño que permitan a este módulo formar parte de un sistema integrado con otros módulos para la Dirección de Información.
- Seleccionar las herramientas para llevar a cabo el proyecto y la elección de la plataforma en la que se desarrollará la aplicación.
- Implementar una aplicación Web que brinde la solución al problema planteado.

Para llevar a cabo las tareas propuestas y lograr satisfactoriamente el resultado final de la investigación se utilizaron los siguientes métodos científicos: el analítico-sintético, la modelación y la observación, de manera que permitieran fundamentar la investigación realizada y ofrecieran los datos precisos para la validación de la propuesta.

Se pretende finalmente desarrollar un sistema para la automatización del servicio de impresión brindado por la Dirección de Información.

El presente trabajo se encuentra estructurado en 4 capítulos, los cuales resumen la siguiente información:

En el **Capítulo 1** se realiza la **fundamentación teórica** del tema, describiendo el objeto de estudio, los sistemas existentes vinculados al campo de acción, las tendencias y tecnologías actuales seleccionadas para el desarrollo de la aplicación y el por qué de su selección, además se explican los principales conceptos a tratar durante la investigación.

En el **Capítulo 2** se realiza el **modelamiento del sistema** a través del modelo de negocio, mostrando una explicación su funcionamiento, determinando los casos de uso del negocio, identificando a los actores y trabajadores que intervienen en el mismo. Se definen los requerimientos funcionales y no funcionales del módulo a desarrollar, se muestran las descripciones de cada caso de uso, así como el diagrama de casos de uso del sistema.

En el **Capítulo 3** se realiza el **análisis y diseño del sistema**. En este se plantean los detalles relacionados con el análisis y diseño del sistema que se propone desarrollar. Se utilizan para su modelado los diagramas de clases del análisis y diseño web, que se necesitan para el almacenamiento de la información persistente, así como los diagramas de interacción correspondientes. También se muestra la descripción de las clases que se utilizarán para cumplir con los requisitos del sistema a desarrollar.

En el **Capítulo 4** se realiza la **implementación y prueba**. Se plantean los detalles relacionados con la implementación del sistema que se propone como una primera versión mediante los diagramas de despliegue y componentes. Finalmente se muestran las pruebas desarrolladas con el objetivo de verificar la funcionalidad del mismo.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Con el desarrollo de este capítulo se pretende brindar al lector una panorámica del funcionamiento e importancia de los servicios de impresión, tanto en universidades como en bibliotecas públicas en el mundo y en nuestro país. Para ello se brinda una breve reseña histórica de la evolución de la impresión de documentos. Luego se hace referencia a algunos de los diferentes servicios de impresión en el mundo, Cuba y la Universidad de las Ciencias Informáticas. Se incluirá además, un estudio de las tendencias, tecnologías, métodos y herramientas a utilizar para el desarrollo del nuevo sistema de impresión de la Dirección de Información de la UCI.

1.1 El servicio de impresión como alternativa eficiente para acceder a los conocimientos

Desde tiempos remotos, el ser humano ha querido adquirir nuevos conocimientos, y con ellos la necesidad de guardar de alguna manera lo que aprendían o las investigaciones que realizaban, y de esta forma surgieron los primeros manuscritos para posteriormente dar paso a los libros. Una vez que una persona escribía un libro o un artículo determinado existía la necesidad de poder tener varios ejemplares, y hasta el año 1449 y aun en años posteriores, los libros se difundían en copias manuscritas por copistas, muchos de los cuales eran monjes y frailes dedicados exclusivamente al rezo y a la réplica de ejemplares por encargo del propio clero o de reyes y nobles. Las ilustraciones y las letras capitales eran producto decorativo y artístico del propio copista, que decoraba cada ejemplar que realizaba según su gusto o visión. Cada uno de sus trabajos podía durar hasta diez años.

La imprenta había sido inventada por los chinos siglos antes, pero en la alta Edad Media se utilizaba en Europa para publicar panfletos publicitarios o políticos, etiquetas, y trabajos de pocas hojas; para ello se trabajaba el texto en hueco sobre una tablilla de madera, incluyendo los dibujos -un duro trabajo de artesanía-. Una vez confeccionada, se acoplaba a una mesa de trabajo, también de madera, y se impregnaban de tinta negra, roja, o azul (sólo existían esos colores). Después se aplicaba el papel y con rodillo se fijaba la tinta. El desgaste de la madera era considerable por lo que no se podían hacer muchas copias con el mismo molde. Cada impresor fabricaba su propio papel, estampando una marca de agua a modo de firma de impresor. Por estas marcas de agua es por lo que se conocen sus trabajos.

Con el paso de los siglos y con el desarrollo vertiginoso de las tecnologías surge la imprenta digital, y con esta el servicio de impresión a través de la red electrónica desplegándose nuevos horizontes en el mundo de la impresión. El ahorro de tiempo y costos ofrecido por las nuevas técnicas digitales valen también para la industria editorial que se beneficia de la rapidez y amplias

posibilidades que la impresión digital ofrece. Algunas de las ventajas que brinda son: la posibilidad de enviar pedidos por correo electrónico, imprimir online, hacer comunicaciones rápidas y utilizar formatos universales como el PDF.

1.2 El servicio de impresión de documentos de bibliotecas universitarias y públicas en el mundo.

En diversas partes del mundo, existen bibliotecas tanto universitarias como públicas que brindan servicio de impresión digital.

En la **Universidad de Valencia** (España) la biblioteca Eduard Bosca ha puesto en funcionamiento un servicio mediante el cual cualquier usuario puede enviar a imprimir los documentos que le interesen desde cualquier ordenador de la Biblioteca.

En la República Bolivariana de Venezuela la biblioteca de la **Universidad Simón Bolívar** ofrece un autoservicio de impresión en red, el cual se realiza utilizando una tarjeta electrónica inteligente con un determinado saldo que una vez enviado el documento a la impresora se presenta la tarjeta en el lector de cualquiera de las máquinas impresoras, e instantáneamente se recoge el documento impreso

La **Universidad de Alicante**, también en España presta servicio de impresión de páginas web, trabajos y artículos desde los ordenadores de uso público de la Biblioteca General. A este servicio se puede acceder a través de la red mediante cualquiera de los ordenadores de la biblioteca, seleccionando el documento a imprimir y de esta forma el trabajo se almacena en la cola de impresión del ordenador del servicio de copistería de la Biblioteca General. En el momento de recoger el trabajo y para identificar el de cada usuario, es imprescindible conocer la hora en que se ha dado la orden de impresión y el código alfanumérico del ordenador desde el que se ha enviado. Este código aparece en una etiqueta pegada al ordenador y está formado por el número correlativo de cada ordenador, dentro de la sección y de la planta de la biblioteca en la que se encuentra.

Las colas de impresión se mantienen cada día hasta la hora de cierre del servicio. A partir de ese momento los trabajos de impresión no son recuperables.

Además, en todos los ordenadores de uso público de la Biblioteca General es posible grabar en disquete tanto páginas web como trabajos propios de los usuarios. Una vez hecha la grabación, es necesario dirigirse con el disquete al servicio de copistería de la planta baja de la Biblioteca General (BG) para solicitar la impresión en papel.

La Universidad de Alicante presta este servicio a profesores y alumnos. Las impresiones realizadas para profesores y servicios de la Universidad se pueden gestionar mediante vales firmados. Dichos vales permanecen en poder de la empresa propietaria de los servicios de impresión y están a disposición de la Universidad durante un período de un año a partir de la fecha de emisión.

Las impresiones realizadas para los alumnos se abonan por los interesados directamente en el servicio de copistería de la BG en el momento de su ejecución. Dicho pago se puede efectuar tanto en moneda de curso legal o mediante la tarjeta de identificación universitaria. [2]

En el **Tecnológico de Monterrey**, México, la Biblioteca ofrece el nuevo servicio de Impresión Express: una computadora exclusiva para imprimir rápidamente archivos guardados en un dispositivo de almacenamiento USB. De ahí se puede enviar al sistema de impresión del Campus, para entonces poder recoger las páginas en cualquier kiosco del Campus.

En la **Biblioteca Nacional Ernesto J. Castillero R.** de Panamá, el servicio de impresión desde las computadoras se ofrece en la Sala de Referencia. Sus costos varían de acuerdo al tipo de impresión, solo texto B/.0.25; imagen en blanco y negro B/.0.25 e imagen en color B/.0.75.

1.2.1 Servicio de impresión de documentos de bibliotecas universitarias y públicas en Cuba.

Algunas de las bibliotecas e instituciones cubanas también ofrecen servicio de impresión de documentos en formato digital, entre las que brindan este servicio se encuentran:

El **Instituto de Información Científica y Tecnológica (IDICT)** el cual tiene una imprenta que cuenta con una moderna tecnología y materiales de gran calidad además de un personal altamente capacitado, ágil y capaz de satisfacer las más altas exigencias de los clientes. Se realizan diseños, realización, impresión, encuadernación y acabado de libros, revistas, folletos, papelería de oficina, modelajes en general, tarjetas de presentación, plegables promocionales, talonarios de comedor, block de notas, libretas telefónicas, certificados, diplomas, almanaques de mesa, almanaques de bolsillo, agendas de mesa, foliado de documentos, tarjetas de felicitación, fin de año, etc.

Los clientes deben presentar los trabajos en COREL DRAW, PAGE MAKER y WORD. Todos los trabajos se diseñan y realizan de forma digital en blanco y negro, colores planos y cuatricromía. La impresión se realiza por el sistema OFFSET y RISOGRAF, la encuadernación se realiza a FEXIBACK y alambre presillado, se hace encuadernación de oficina con BILDING y el empaquetado es RETRACTILADO. [3]

El **Instituto Superior Politécnico José Antonio Echeverría (CUJAE)** ofrece un servicio de impresión de documentos, especialmente tesis de maestría y doctorado.

1.3 El servicio de impresión de documentos en la UCI

En la Universidad de las Ciencias Informáticas, la biblioteca, que pertenece a la Dirección de Información, brinda un servicio de impresión al estudiante totalmente gratuito y de una manera peculiar y única en el país. Para imprimir, el estudiante requiere una autorización firmada por el jefe de su brigada con la cantidad de hojas que necesite (cada brigada puede imprimir 60 hojas mensuales), una vez autorizado puede ir al departamento de impresión con el documento que en un dispositivo de almacenamiento extraíble, lo copia en la computadora disponible para imprimir a través de una secretaria y luego recoge el documento impreso a la hora indicada por esta.

1.4 ¿Por qué realizar un nuevo sistema de impresión de documentos para la Dirección de Información de la Universidad de las Ciencias Informáticas?

Existen varias causas por las cuales se hace necesaria la creación de un nuevo sistema. Las más significativas son:

- El servicio de impresión aún no está automatizado por completo, pues actualmente solo está automatizado en el local de impresión, el resto del proceso es necesario hacerlo de forma manual mediante una autorización escrita, que en muchos casos puede no ser confiable.
- En la oficina de impresión se acumula una gran cantidad de estudiantes en espera de recibir el servicio, el cual a veces se torna un poco lento debido a que cada estudiante que desee imprimir debe llevar su documento en un dispositivo de almacenamiento extraíble, el cual demora en copiar porque cada uno tiene que ser analizado para detectar posibles virus.
- El sistema actual está construido con software propietario, por lo que se hace necesario el desarrollo de una aplicación basada totalmente en software libre.

1.5 ¿Qué es el software libre?

Software libre (en inglés free software) es el software que, una vez obtenido, puede ser usado, copiado, estudiado, modificado y redistribuido libremente. El software libre hace alusión a las principales libertades a las que tienen derecho los usuarios de este tipo de software:

- La libertad de usar el programa, con cualquier propósito (libertad 0).
- La libertad de estudiar cómo funciona el programa, y adaptarlo a tus necesidades (libertad 1). El acceso al código fuente es una condición previa para esto.

- La libertad de distribuir copias, con lo que puedes ayudar a tu vecino (libertad 2).
- La libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie. (libertad 3). El acceso al código fuente es un requisito previo para esto.”

1.5.1 Algunas ventajas del Software Libre.

Costo: el costo total de propiedad del sistema operativo libre Linux es menos de la mitad que el de Windows. Gran parte del ahorro proviene de no tener que pagar licencia y de sus menores costos de administración.

Innovación tecnológica: el desarrollo en comunidad de este sistema y el conocimiento del Código Fuente, propician que a cada instante, un desarrollador necesite nuevas actualizaciones y las realice él mismo, proponiendo nueva funcionalidad al programa.

Requisitos de hardware y durabilidad de las soluciones: un mejor soporte en general para las versiones antiguas de software y de plataformas de hardware o software minoritarias. No necesitas comprar nuevas versiones para estar actualizado.

Escrutinio público: el proceso de revisión pública al que está sometido el desarrollo del software libre imprime un gran dinamismo al proceso de corrección de errores. Cada mejora es socializada libremente, la comunidad puede cambiar la realidad de las innovaciones.

Desarrollo de la industria local: en el Software Libre no hay costo de licencia debido al derecho a copia y, al disponer del código fuente de la aplicación, es posible desarrollar internamente las mejoras o las modificaciones necesarias. De este modo, se contribuye a la formación de profesionales en nuevas tecnologías y al desarrollo local y de la industria nacional de software.

Datos personales, privacidad y seguridad: el Software Libre, por su carácter abierto, dificulta la introducción de código malicioso, espía o de control remoto, debido a que el código lo revisan muchos usuarios y desarrolladores que pueden detectar posibles puertas traseras. En el mundo del software libre, cualquier programador puede realizar una auditoría para comprobar que no se ha introducido ningún código malicioso, y, a su vez, cualquier entidad puede añadir libremente encriptación adicional a la aplicación que utilice para proteger sus datos.

Adaptación del software: el Software Libre permite personalizar, gracias al hecho de que se dispone del código fuente, los programas tanto como sea necesario hasta que cubran exactamente cualquier necesidad. [4]

1.5.2 Desventaja del Software Libre.

Se puede considerar como una desventaja del software libre la existencia de más de una imagen y cada desarrollador o usuario utiliza la que más conveniente le resulte y así la comunidad no se pone de acuerdo hacia cual versión se debe migrar lo cual repercute en los usuarios finales que no son especialistas en tecnologías informáticas, por ejemplo secretarías, estudiantes y población en general que se sientan frente a una computadora y tiene una versión diferente lo cual puede dificultar su trabajo.

A pesar de tener esta desventaja las libertades y ventajas muestran la medida de la viabilidad económica de este sistema libre, que por tener esta condición no es necesariamente gratuito, sino que brinda la posibilidad de comercializarlo, regalarlo, prestarlo con total libertad ya que está protegido legalmente. Evidentemente, a pesar de las lagunas técnicas que pueda tener aún el Software Libre, es la alternativa para los países subdesarrollados. [4]

1.6 El Software Libre como alternativa de solución para la automatización de los servicios de la Dirección de Información de la UCI.

La biblioteca de la UCI que pertenece a la Dirección de Información es la encargada de promover el conocimiento entre los estudiantes y profesores, por lo que debe asegurar a su comunidad de usuarios la mejor utilización de sus recursos y el mejor servicio posible, entre ellos, mejorar el servicio de impresión a estudiantes. Para esto en el mundo actual existen diferentes herramientas tecnológicas que son de gran ayuda para facilitar y mejorar la calidad de los servicios, pero los elevados costos y compromisos económicos que puede implicar la compra de un software de automatización, pueden ser un obstáculo para el obtener dichas tecnologías. Por esta razón, la utilización de Software Libre representa una alternativa útil y factible, que cada vez gana en calidad, utilización y reconocimiento de su aporte.

1.7 ¿Qué es una arquitectura?

Una arquitectura es un entramado de componentes funcionales que aprovechando diferentes estándares, convenciones, reglas y procesos, permite integrar una amplia gama de productos y servicios informáticos, de manera que pueden ser utilizados eficazmente dentro de una organización.

Debemos señalar que para seleccionar el modelo de una arquitectura, hay que partir del contexto tecnológico y organizativo del momento y, que la arquitectura Cliente/Servidor requiere una determinada especialización de cada uno de los diferentes componentes que la integran.

1.8 Aplicaciones de tipo Cliente-Servidor

Los elementos principales de la arquitectura cliente-servidor son justamente el elemento llamado cliente y el otro elemento llamado servidor, donde el servidor es un proveedor de servicios y el cliente es el consumidor de estos servicios, interactuando ambos mediante un mecanismo de pasaje de mensajes: el pedido de servicio y la respuesta.

Normalmente el servidor es una máquina bastante potente que actúa de depósito de datos y funciona como un Sistema Gestor de Base de Datos (SGBD). Por otro lado los clientes suelen ser estaciones de trabajo que solicitan varios servicios al servidor. Ambas partes deben estar conectadas entre sí mediante una red. [5]

Entre las principales características de la arquitectura cliente/servidor se pueden destacar las siguientes:

- El servidor presenta a todos sus clientes una interfaz única y bien definida.
- El cliente no necesita conocer la lógica del servidor, sólo su interfaz externa.
- El cliente no depende de la ubicación física del servidor, ni del tipo de equipo físico en el que se encuentra, ni de su sistema operativo.
- Los cambios en el servidor implican pocos o ningún cambio en el cliente.

1.8.1 ¿Qué es un cliente?

Es el que inicia un requerimiento de servicio. El requerimiento inicial puede convertirse en múltiples requerimientos de trabajo a través de redes LAN o WAN. La ubicación de los datos o de las aplicaciones es totalmente transparente para el cliente.

1.8.2 ¿Qué es un servidor?

Es cualquier recurso de cómputo dedicado a responder a los requerimientos del cliente. Los servidores pueden estar conectados a los clientes a través de redes LANs o WANs, para proveer de múltiples servicios a los clientes tales como: impresión, acceso a bases de datos, fax, procesamiento de imágenes, etc.

1.8.3 Ventajas de la arquitectura cliente/servidor

El modelo cliente/servidor se recomienda, en particular, para redes que requieran un alto grado de fiabilidad. Las principales ventajas son:

- **Recursos centralizados:** debido a que el servidor es el centro de la red, puede administrar los recursos que son comunes a todos los usuarios, por ejemplo: una base de datos centralizada se utilizaría para evitar problemas provocados por datos contradictorios y redundantes.
- **Seguridad mejorada:** ya que la cantidad de puntos de entrada que permite el acceso a los datos no es importante.

- **Administración a nivel del servidor:** ya que los clientes no juegan un papel importante en este modelo, requieren menos administración.
- **Red escalable:** gracias a esta arquitectura, es posible quitar o agregar clientes sin afectar el funcionamiento de la red y sin la necesidad de realizar mayores modificaciones. [5]

1.8.4 Desventajas de la arquitectura cliente/servidor

La arquitectura cliente/servidor también tiene las siguientes desventajas:

- **Costo elevado:** debido a la complejidad técnica del servidor.
- **Un eslabón débil:** el servidor es el único eslabón débil en la red de cliente/servidor, debido a que toda la red está construida en torno a él. Afortunadamente, el servidor es altamente tolerante a los fallos (principalmente gracias al sistema RAID). [6]

1.9 Sistemas de gestión de contenidos (CMS)

Un Sistema de Gestión de Contenidos (CMS: Content Management System) es una aplicación web que le permite desde su Navegador de Internet, y sin necesidad de tener grandes conocimientos informáticos, mantener los contenidos de un Sitio Web siempre actualizados en tiempo real. Permite controlar de forma independiente el diseño, la estructura y el contenido del sitio, además se utilizan bases de datos para permitir que la actualización del Sitio Web se realice a través de sencillos formularios que actualizan plantillas estandarizadas. Estos Sistemas de Gestión de Contenidos son modulares, o sea el creador del sitio web decidirá que módulos desea incorporar a su sitio y cuáles no.

1.9.1 Algunas ventajas de los sistemas de gestión de contenidos

Los CMS cuentan con una serie de ventajas entre las cuales se encuentran:

Control de acceso: permite gestionar los diferentes permisos a cada área del web por parte de los usuarios, estableciendo diferentes roles para los distintos tipos de usuario.

Mantenimiento de gran cantidad de páginas: a través del sistema de control de acceso y asignación de roles, permite distribuir los trabajos de creación, edición y mantenimiento con permisos de acceso a las diferentes áreas. Así como gestionar los metadatos de cada documento, las versiones, la publicación y caducidad de páginas y los enlaces rotos, entre otros aspectos.

Inclusión de nuevas funcionalidades en el sitio: Se pueden incluir nuevos módulos sin la necesidad de grandes cambios en la aplicación web. Esto permite que el sistema pueda crecer y adaptarse a las futuras necesidades.

Reutilización de componentes: un CMS permite recuperar y reutilizar documentos, páginas, además de cualquier objeto publicado o almacenado.

Consistencia de la web: Los CMS pueden aplicar un mismo estilo en todas las páginas a través de la utilización del CSS (Cascading Style Sheets u hojas de estilo en cascada), y aplicar una misma estructura mediante patrones de páginas. De esta forma se garantiza una buena consistencia en la web.

Cambios del aspecto de la web: Los CMS facilitan los cambios con la utilización, por ejemplo, del estándar CSS con lo que se consigue la independencia de presentación y contenido, de esta forma un cambio en el contenido no afectaría el aspecto de la web.

1.9.2 ¿CMS, libre o propietario?

Se puede hacer una primera división de los CMS según el tipo de licencia escogido. Por una parte están los CMS comercializados por empresas que consideran el código fuente un activo más que tienen que mantener en propiedad, y que no permiten que terceros tengan acceso. Por la otra tenemos los de código fuente abierto, desarrollados por individuos, grupos o empresas que permiten el acceso libre y la modificación del código fuente. La disponibilidad del código fuente posibilita que se hagan personalizaciones del producto, correcciones de errores y desarrollo de nuevas funciones. Este hecho es una garantía de que el producto podrá evolucionar incluso después de la desaparición del grupo o empresa creadora. Utilizar una herramienta de gestión de contenidos de código abierto tiene otra ventaja que hace decidirse a la mayoría de usuarios: su coste. Habitualmente todo el software de código abierto es de acceso libre, es decir, sin ningún coste en licencias. Sólo en casos aislados se hacen distinciones entre empresas y entidades sin ánimo de lucro o particulares. En comparación, los productos comerciales pueden llegar a tener un coste que sólo una gran empresa puede asumir. En cuanto al soporte, los CMS comerciales acostumbran a dar soporte profesional, con un coste elevado en muchos casos, mientras que los de código abierto se basan más en las comunidades de usuarios que comparten información y solución a los problemas. En el mercado hay CMS de calidad tanto comerciales como de código abierto. Muchos CMS de código abierto están poco elaborados (aunque en plena evolución), pero también lo encontramos entre los comerciales. En definitiva, un buen CMS de código abierto es mucho más económico que su homólogo comercial, con la ventaja de disponer de todo el código fuente y de una extensa comunidad de usuarios. Por todos estos motivos, y como apuesta por la filosofía del software libre, sólo se presentan algunos CMS de código abierto.

1.9.3 Algunos tipos de CMS

Plone (de Zope)

Distribuido con licencia GNU GPL, está basado en el servidor de aplicaciones Zope y programado en Python. Solución multiplataforma: Linux, Solaris, Windows o Mac. Puede utilizarse como servidor intranet o extranet, un sistema de publicación de documentos y una herramienta de trabajo en grupo para colaborar entre entidades distantes.

El proyecto Plone comenzó en 1999 por Alan Runyan, Alexander Limi, y Vidar Andersen. Rápidamente se convirtió en uno de los populares y poderosos Sistemas de Gestión de Contenido de Código Abierto. En el 2004 se creó por la Fundación Plone para proteger y promover el uso de Plone.

Joomla!

Gestor de contenidos desarrollado en PHP, funciona junto con MySQL y permite autenticación mediante LDAP. En Joomla! se incluyen características como: hacer caché de páginas para mejorar el rendimiento, indexación web, feed RSS, versiones imprimibles de páginas, flash con noticias, blogs, foros, polls, calendarios, búsqueda en el sitio web, e internacionalización del lenguaje.

La primera versión de Joomla! (Joomla! 1.0.0) fue publicada el 16 de septiembre de 2005. Se trataba de una versión mejorada de Mambo 4.5.2.3 combinada con otras modificaciones de seguridad y anti-bugs. [7]

e107

Es un CMS escrito en PHP, que usa MySQL como base de datos. Es completamente gratuito y está en constante desarrollo.

Desarrollado desde finales de 1998, e107 se ha hecho un espacio importante en el mundo de los CMS de libre distribución. No es de los más conocidos pero incorpora una serie de características muy depuradas a la hora de la administración de contenidos que lo convierten en uno de los favoritos en muchos lugares del mundo.

door108

Es un sistema de gestión de contenidos escrito íntegramente en PHP que utiliza el sistema de base de datos MySQL para el almacenamiento del contenido, basado en e107. door108 es gratis, libre, totalmente personalizable y en constante desarrollo.

Drupal

Es un sistema de gestión de contenido para sitios Web. Permite publicar artículos, imágenes, u otros archivos y servicios añadidos como foros, encuestas, votaciones, blogs y administración de usuarios y permisos. Drupal es un sistema dinámico: en lugar de almacenar sus contenidos en

archivos estáticos en el sistema de ficheros del servidor de forma fija, el contenido textual de las páginas y otras configuraciones son almacenados en una base de datos y se editan utilizando un entorno Web incluido en el producto.

1.10 El CMS Drupal, una potente herramienta para la construcción del nuevo servicio de impresión de documentos de la UCI

Con el nuevo servicio de impresión de la UCI, se espera como aporte práctico proveer al personal encargado, de una aplicación Web con nuevas funcionalidades además de las que ya posee el servicio actual. Se pretende brindar una interfaz de administración para la completa edición y actualización del contenido que se maneja en el sitio, sin necesidad de ningún conocimiento técnico por parte de él o los editores, todo esto basándose en un sistema de usuarios y roles definidos por el administrador del sitio.

Después de haber realizado un amplio estudio sobre los Sistemas de Manipulación de Contenidos (CMS), analizando las características y particularidades de cada uno de ellos, el CMS Drupal se presenta como una buena herramienta para desarrollar el sitio web mediante el cual se podrá acceder al servicio de impresión de documentos de la UCI, pues entre sus características permite:

Permisos basados en roles: Los administradores de Drupal no tienen que tediosamente configurar los permisos para cada usuario, simplemente asignar usuarios a roles y darles permisos.

Búsquedas: Todo el contenido en Drupal tiene el índice de búsqueda, lo que facilita la información al usuario.

Autenticación de los usuarios: Los usuarios se pueden registrar y autenticar a nivel local ó utilizando una fuente de autenticación externa como Jabber, Blogger, LiveJournal u otro sitio de Drupal. Para uso en una intranet, Drupal se puede integrar con un servidor LDAP.

Plantillas: El sistema de plantillas de Drupal separa el contenido de la presentación permitiendo controlar el aspecto del sitio. Las plantillas se crean con el estándar de codificación HTML y PHP lo que significa que no se tiene que aprender un lenguaje propio de plantillas.

Control de versiones: El control de versiones de Drupal rastrea las actualizaciones del contenido, lo que se ha cambiado, la fecha y hora de los cambios realizados y más.

Plataforma Soporte para Apache ó IIS, Unix / Linux / BSD / Solaris / Windows / Mac OS. Drupal ha sido diseñado desde el principio para ser multi-plataforma. No sólo se puede utilizar ya sea con Apache o IIS de Microsoft, también se puede usar Drupal en Linux, BSD, Solaris, Windows y Mac OS X plataformas. Independencia de la base de datos - Si bien muchos de los usuarios de Drupal utilizan MySQL, MySQL se sabe que no es solución para todos. Drupal es construido con una capa de abstracción de bases de datos que le permite utilizar Drupal con

MySQL y PostgreSQL, al igual que otras bases de datos propietarias.

Multi-idioma: Drupal está diseñado para satisfacer las necesidades de una audiencia internacional y proporciona un marco completo para crear un sitio web en varios idiomas. Todo el texto puede ser traducido utilizando una interfaz gráfica de usuario, mediante la importación de las traducciones existentes, o mediante la integración con otras herramientas de traducción como el GNU gettext.

Administración y análisis, Seguimiento y Estadísticas: Drupal puede imprimir informes basados en navegador con información sobre referencias, el contenido y la forma de navegación de los usuarios de un sitio. Presentación de informes - Todas las importantes actividades y eventos del sistema son capturados en un registro de eventos para ser revisado por un administrador en un momento posterior.

Administración vía web: Drupal se puede administrar completamente mediante un navegador web, lo que permite acceder a ella desde todo el mundo y no requiere software adicional para ser instalado en el equipo. [8]

1.11 Metodologías de desarrollo de software

Una metodología es un conjunto de procedimientos, técnicas, herramientas y un soporte documental que ayuda a los desarrolladores a realizar nuevo software. Una metodología puede seguir uno o varios modelos de ciclo de vida, es decir, el ciclo de vida indica qué es lo que hay que obtener a lo largo del desarrollo del proyecto pero no cómo hacerlo. La metodología indica cómo hay que obtener los distintos productos parciales y finales.

Independientemente de que se pueden utilizar diferentes metodologías para llevar a cabo el control del proceso de producción de software existen características que son inviolables a la hora de su utilización para que el trabajo de desarrollo se realice eficientemente. La norma 1074 de la IEEE (Instituto de Ingenieros Eléctricos y Electrónicos) establece que toda metodología de desarrollo de software debe incluir como serán capturados los requisitos, como se realizará el diseño, la implementación y las pruebas del producto que se desea construir.

1.11.1 Algunas metodologías de desarrollo de software

SCRUM

Desarrollada por Ken Schwaber, Jeff Sutherland y Mike Beedle. Define un marco para la gestión de proyectos, que se ha utilizado con éxito durante los últimos 10 años. Está especialmente indicada para proyectos con un rápido cambio de requisitos. Sus principales características se pueden resumir en dos. El desarrollo de software se realiza mediante iteraciones, denominadas sprints, con una duración de 30 días. El resultado de cada sprint es un incremento ejecutable que se muestra al cliente. La segunda característica importante son las reuniones a lo largo proyecto. Éstas son las verdaderas protagonistas, especialmente la reunión diaria de 15 minutos del equipo

de desarrollo para coordinación e integración.

Programación Extrema (Extreme Programming, XP)

Es una metodología ágil concentrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico.

Crystal

Se trata de un conjunto de metodologías para el desarrollo de software caracterizadas por estar centradas en las personas que componen el equipo y la reducción al máximo del número de artefactos producidos. Ha sido desarrollada por Alistair Cockburn el en lugar de partir solamente de su experiencia personal para construir una teoría de cómo deben hacerse las cosas, complementa su experiencia directa con la búsqueda activa de proyectos y ver cómo trabajan. Además no teme alterar sus puntos de vista con base en sus descubrimientos. El desarrollo de software se considera un juego cooperativo de invención y comunicación, limitado por los recursos a utilizar. El equipo de desarrollo es un factor clave, por lo que se deben invertir esfuerzos en mejorar sus habilidades y destrezas, así como tener políticas de trabajo en equipo definidas. Estas políticas dependerán del tamaño del equipo, estableciéndose una clasificación por colores, por ejemplo Crystal Clear (3 a 8 miembros) y Crystal Orange (25 a 50 miembros).

RUP

Dentro de las metodologías fuertes la que más se destaca es el **Proceso Unificado de Modelado (RUP)**. Es una metodología orientada a objetos que tiene como características principales: estar dirigido por los casos de uso, ser centrado en la arquitectura, y ser iterativo e incremental.

Es un proceso de desarrollo de software configurable que se adapta a través de los proyectos variados en tamaños y complejidad. Se basa en muchos años de experiencia en el uso de la tecnología orientada a objetos en el desarrollo de software de misión crítica en una variedad de industrias por la compañía Rational.

El Proceso Unificado sirve como guía a los equipos de proyecto en cómo administrar el desarrollo

iterativo de un modo controlado mientras se balancean los requerimientos del negocio, el tiempo al mercado y los riesgos del proyecto. Describe los diversos pasos involucrados en la captura de los requerimientos y en el establecimiento de una guía arquitectónica lo más pronto posible, para diseñar y probar el sistema hecho de acuerdo a los requerimientos y a la arquitectura.

Describe qué entregables producir, cómo desarrollarlos y también provee patrones. Es soportado por herramientas que automatizan entre otras cosas, el modelado visual, la administración de cambios y las pruebas. Se caracteriza básicamente por ser vital la captura de requisitos, iteración actual condicionada por la anterior, se necesita de un buen líder de proyecto para garantizar el trabajo del equipo de desarrollo, se realiza un gran número de artefactos lo que puede provocar retrasos por mala preparación de los analistas, las responsabilidades están divididas y es aplicable a todo tipo de proyecto asumiendo sus extensiones.

1.11.2 ¿Por qué utilizar RUP?

Después de haber visto y estudiado las distintas clasificaciones de metodologías de desarrollo que existen así como las posibilidades que brinda cada una y teniendo en cuenta las características del trabajo que se va a desarrollar se seleccionó como metodología de desarrollo a RUP. Se tuvo en cuenta que unifica los mejores elementos de metodologías que le antecieron, que está preparado para desarrollar grandes y complejos proyectos, ofrece una gran gama de ventajas por ser orientada a objetos y además utiliza el UML como lenguaje de modelado y representación visual.

1.12 Tecnologías y lenguajes a utilizar para el desarrollo del sistema

Existen lenguajes de programación de scripts que residen en el lado del cliente y lenguajes que residen en el lado del servidor. Cada uno de ellos tiene sus ventajas e inconvenientes.

1.12.1 Lenguajes del lado de cliente

Un lenguaje del lado cliente es totalmente independiente del servidor, lo cual permite que la página pueda ser albergada en cualquier sitio. Pero la página no se verá bien si el ordenador cliente no tiene instalados los **plug-in** adecuados. El código, tanto del hipertexto como de los **scripts**, es accesible a cualquiera y ello puede afectar a la seguridad.

- **HTML**, HyperText Markup Language (Lenguaje de Marcas de Hipertexto), es el lenguaje de marcado predominante para la construcción de páginas web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes. HTML se escribe en forma de "etiquetas", rodeadas por corchetes angulares (<,>). HTML también puede incluir un script (por ejemplo Javascript), el cual puede afectar el comportamiento de navegadores web y otros procesadores de HTML.

• **Javascript** es un lenguaje con muchas posibilidades, utilizado para crear pequeños programas que luego son insertados en una página web y en programas más grandes, orientados a objetos mucho más complejos. Con Javascript es posible crear diferentes efectos e interactuar con los usuarios.

Este lenguaje posee varias características, entre ellas podemos mencionar que es un lenguaje basado en acciones que posee menos restricciones. Gran parte de la programación en este lenguaje está centrada en describir objetos, escribir funciones que respondan a movimientos del mouse, aperturas, utilización de teclas, cargas de páginas entre otros.

Es necesario resaltar que hay dos tipos de Javascript: por un lado está el que se ejecuta en el cliente, este es el Javascript propiamente dicho, aunque técnicamente se denomina Navigator Javascript. Pero también existe un Javascript que se ejecuta en el servidor, es más reciente y se denomina LiveWire Javascript.

1.12.2 Lenguajes del lado del servidor

Un lenguaje del lado servidor es independiente del navegador utilizado, que no necesitará **plug-in** especiales para visualizar correctamente cualquier página. Además, el código de los **script** puede ocultarse al terminal cliente, que sólo verá el código HTML terminado. Por contra, la prueba y depuración de un script deberá hacerse desde el servidor.

PHP

Es un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor. El código PHP se incluye entre etiquetas especiales de comienzo y final que permiten entrar y salir del modo PHP. Lo que distingue a PHP de la tecnología Javascript, la cual se ejecuta en la máquina cliente, es que el código PHP es ejecutado en el servidor. PHP puede hacer cualquier cosa que se pueda hacer con un script CGI, como procesar la información de formularios, generar páginas con contenidos dinámicos, o mandar y recibir cookies.

PHP es un lenguaje de programación de estilo clásico, con variables, sentencias condicionales, bucles, funciones. No es un lenguaje de marcas como podría ser HTML, XML o WML. El resultado es normalmente una página HTML pero igualmente podría ser una página WML.

PHP es un lenguaje con asequible estructura de programación, la facilidad de llevar a cabo sentencias SQL embebidas, además de permitir la posibilidad de correr en diferentes tipos de servidores, entre ellos Apache.

Quizás la característica más potente y destacable de PHP es su soporte para una gran cantidad de bases de datos. Escribir una interfaz vía web para una base de datos es una tarea simple con PHP.

También soporta el uso de otros servicios que usen protocolos como IMAP, SNMP, NNTP, POP3,

HTTP y derivados. También se pueden abrir sockets de red directos (raw sockets) e interactuar con otros protocolos. [9]

SQL

El **Lenguaje de consulta estructurado** (SQL, **Structured Query Language**) no es más que un lenguaje estándar de comunicación con bases de datos. Es, por tanto, un lenguaje normalizado que permite trabajar con cualquier tipo de lenguaje (ASP o PHP) en combinación con cualquier tipo de base de datos (MS Access, SQL Server, MySQL).

1.13 ¿Qué es un IDE?

Integrated Development Environment (Entorno de Desarrollo Integrado), es decir un editor de código que además puede servirnos para depurar y facilitarnos las diferentes tareas necesarias en el desarrollo de cualquier tipo de aplicación. Hay que aclarar que es posible que un mismo IDE pueda funcionar con diferentes lenguajes de programación, un buen ejemplo de ello es Eclipse.

1.13.1 Algunos IDE's para programar en PHP

Zend Studio

Son muchos los desarrolladores que trabajan con Zend Studio, es posiblemente uno de los mejores IDE del momento. Se trata de un programa de la casa Zend, uno de los mayores impulsores de PHP, orientada a desarrollar aplicaciones web en PHP.

Zend Studio es un editor de texto para páginas PHP que proporciona un buen número de ayudas desde la creación y gestión de proyectos hasta la depuración del código.

Zend Studio o **Zend Development Environment** es un completo entorno integrado de desarrollo para el lenguaje de programación PHP. Está escrito en Java, y está disponible para las plataformas Microsoft Windows, Mac OS X y GNU/Linux.

Zend Studio fue diseñado para usarse con el lenguaje PHP; sin embargo ofrece soporte básico para otros lenguajes Web, como HTML, Javascript y XML.

PDT (PHP Development Tools, Eclipse)

Este proyecto ha tenido una gran respuesta entre los desarrolladores de PHP, y entre las características en la versión actual (1.0) se encuentran:

- Editor sensible al contexto, el cual provee resaltado de código, asistente de código y autocompletado de código y soporte para el debug incremental del código de PHP.

- Integración con el modelo del proyecto Eclipse, que permite para inspeccionar el uso de las vistas del contorno del fichero y del proyecto, así como la nueva vista PHP Explorer.
- Extensos frameworks y APIs que permiten a los desarrolladores e ISVs (vendedores de software independientes) fácilmente extender PDT para crear nuevas e interesantes herramientas orientadas al desarrollo de PHP.

Php Designer

Es un completo entorno de desarrollo y programación especialmente diseñado para desarrolladores de PHP, aunque también permite trabajar con comodidad en otros lenguajes de programación como HTML, XHTML, CSS y SQL.

Ofrece toda una serie de asistentes y diálogos integrados que facilitan en todo momento tu tarea, además de acceso directo a librerías de código o scripts de uso habitual, utilidades diversas y toda suerte de herramientas, todo ello en una interfaz de diseño sencillo y elegante que puedes personalizar con nada menos que dieciocho temas distintos.

PhpEdit

Ofrece un entorno de trabajo para todos aquellos programadores especializados en PHP, pero que también permite trabajar con HTML, XML y TXT; un entorno integrado para Windows y con una variada colección de herramientas.

Entre dichas herramientas se incluyen plantillas de teclado, marcadores de texto, un "debugger" integrado para detectar errores en el código y un código de color para comandos de sintaxis que hace más sencilla e intuitiva la lectura. [10]

1.14 Sistema Gestor de Bases de Datos

Un **Sistema Gestor de Bases de Datos** –SGBD- (Data Base Management System DBMS) consiste en una colección de datos interrelacionados y un conjunto de programas para acceder a esos datos. El Objetivo primordial de un SGBD es proporcionar un entorno que sea a la vez conveniente y eficiente para ser utilizado al extraer y almacenar información de la base de datos.

“El sistema de gestión de la base de datos (SGBD) es una aplicación que permite a los usuarios definir, crear y mantener la base de datos, y proporciona acceso controlado a la misma.”

1.14.1 Funcionalidades básicas de un SGBD

- **Control de la redundancia:** La redundancia de datos tiene varios efectos negativos (duplicar el trabajo al actualizar, desperdicia espacio en disco, puede provocar inconsistencia de datos) aunque a veces es deseable por cuestiones de rendimiento.
- **Restricción de los accesos no autorizados:** cada usuario ha de tener unos permisos de

acceso y autorización.

- **Cumplimiento de las restricciones de integridad:** el SGBD ha de ofrecer recursos para definir y garantizar el cumplimiento de las restricciones de integridad.

1.14.2 Tipos de SGBD

PostgreSQL es un sistema de gestión de base de datos relacional orientada a objetos de software libre, publicado bajo la licencia BSD.

PostgreSQL está ampliamente considerado como el sistema de bases de datos de código abierto más avanzado del mundo. Posee muchas características, que tradicionalmente sólo se podían ver en productos comerciales de alto calibre.

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

Firebird es una base de datos relacional que ofrece muchas características de SQL ANSI estándar y que funciona en Linux, Windows, MacOSX y una variedad de plataformas UNIX. Firebird ofrece una concurrencia excelente, alto rendimiento y un poderoso lenguaje de procedimientos almacenados y disparadores. Ha estado usándose en producción bajo varios nombres desde 1981. [11]

SQLite es una librería escrita en lenguaje C que implementa un manejador de base de datos SQL embebido. Sus desarrolladores destacan, que su principal característica, es su completo soporte para tablas e índices en un único archivo por base de datos, soporte transaccional, rapidez (unas 2 veces mas veloz que MySQL y PostgreSQL), escaso tamaño (unas 25 mil líneas de código C) y su completa portabilidad.

MySQL es la base de datos Open Source más popular y, posiblemente, mejor del mundo. Su continuo desarrollo y su creciente popularidad está haciendo de MySQL un competidor cada vez más directo de gigantes en la materia de las bases de datos como Oracle.

Es un sistema de administración de bases de datos (*Database Management System, DBMS*) para bases de datos relacionales.

También es muy destacable, la condición de Open Source de MySQL, que hace que su utilización sea gratuita e incluso se pueda modificar con total libertad, pudiendo descargar su código fuente. Esto ha favorecido muy positivamente en su desarrollo y continuas actualizaciones, para hacer de MySQL una de las herramientas más utilizadas por los programadores orientados a Internet. [12]

1.13.3 ¿Por qué usar MySQL?

MySQL es un potente servidor de base de datos SQL, ligero y preparado para manejar grandes cantidades de datos a la mayor velocidad posible. Es muy utilizado en aplicaciones web como Drupal o phpBB, en plataformas (Linux/Windows/Apache/MySQL/PHP/Perl/Python), y por herramientas de seguimiento de errores como Bugzilla. Su popularidad como aplicación web está muy ligada a PHP, que a menudo aparece en combinación con MySQL. En aplicaciones web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones.

MySQL es un sistema de administración relacional de bases de datos. Una base de datos relacional archiva datos en tablas separadas en vez de colocar todos los datos en un gran archivo. Esto permite velocidad y flexibilidad. Las tablas están conectadas por relaciones definidas que hacen posible combinar datos de diferentes tablas sobre pedido.

Además MySQL es software de fuente abierta, esto significa que es posible para cualquier persona usarlo y modificarlo. Cualquier persona puede bajar el código fuente de MySQL y usarlo sin pagar. Cualquier interesado puede estudiar el código fuente y ajustarlo a sus necesidades. MySQL usa el GPL (GNU General Public License) para definir que puede hacer y que no puede hacer con el software en diferentes situaciones. Si usted no se ajusta al GPL o requiere introducir código MySQL en aplicaciones comerciales, se puede comprar una versión comercial licenciada.

1.14 Servidores Web

Un servidor web es un programa que implementa el protocolo HTTP (*hypertext transfer protocol*). Este protocolo pertenece a la capa de aplicación del modelo OSI y está diseñado para transferir lo que se denomina hipertextos, páginas web o páginas HTML (*hypertext markup language*): textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de música.

Básicamente, un servidor web sirve contenido estático a un [navegador](#), carga un archivo y lo sirve a través de la red al [navegador](#) de un usuario. Este intercambio es mediado por el navegador y el servidor que hablan el uno con el otro mediante [HTTP](#). Se pueden utilizar varias tecnologías en el servidor para aumentar su potencia más allá de su capacidad de entregar páginas HTML; éstas incluyen scripts CGI, seguridad SSL y páginas activas del servidor (ASP).

De esta manera se hace necesario realizar un estudio de los diferentes servidores web que se utilizan para este tipo de aplicaciones.

1.14.1 Algunos tipos de servidores web

Servidor HTTP Cherokee

Es un servidor web libre, multiplataforma, abierto bajo la licencia GPL. Cherokee es un servidor web rápido, flexible y fácil de configurar, compatible con las principales tecnologías existentes en la actualidad (FastCGI, SCGI, PHP, CGI, TLS, SSL, etc.). Está escrito completamente en C, es escalable y puede usarse como un sistema integrado. También soporta registro y autenticación de usuarios Cherokee que puede también realizar redirecciones y soporta la configuración de Servidores Virtuales.

ISS

Internet Information Services (IIS) es un componente software que integra una serie de herramientas para la creación, configuración y administración de sitios Web, además de incluir otros servicios de Internet para la plataforma Windows. Entre estos servicios se encuentra un servicio Web, además de los siguientes: FTP, servicio para la transferencia de ficheros por red. SMTP, servicio para la transferencia de correo electrónico. NNTP, servicio para la transferencia de noticias.

Apache

El servidor HTTP Apache es un servidor HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etcétera), Windows y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual. El servidor Apache se desarrolla dentro del proyecto HTTP Server (httpd) de la Apache Software Foundation. Presenta entre otras características mensajes de error altamente configurables, bases de datos de autenticación y negociado de contenido.

Apache tiene amplia aceptación en la red: en el 2005, Apache es el servidor HTTP más usado, siendo el servidor HTTP del 70% de los sitios Web en el mundo y creciendo aún su cuota de mercado (estadísticas históricas y de uso diario proporcionadas por Netcraft).

La arquitectura del servidor Apache es muy modular. El servidor consta de un sección core y mucha de la funcionalidad que podría considerarse básica para un servidor Web es provista por módulos.

1.14.2 ¿Por qué utilizar Apache?

Después de haber estudiando diferentes tipos de servidores web, la utilización de Apache viene dada porque es un potente servidor web que satisface perfectamente los requerimientos de la aplicación a desarrollar debido a su arquitectura modular, sus mensajes de error altamente configurables, sus bases de datos de autenticación y negociado de contenido.

1.15 El lenguaje de modelado de Software

A lo largo de los años, el desarrollo de los proyectos de software causa bastantes confusiones y malas interpretaciones en los requerimientos de los clientes y usuarios, en parte debido a la

abundancia de notaciones, metodologías y conceptos que hace que los desarrolladores de sistemas no se pongan de acuerdo en que es lo que realmente están elaborando. En un esfuerzo para estandarizar las notaciones y procesos a utilizar, se conformó un consorcio liderado por la empresa Rational y por las principales empresas del mundo de la industria de la informática, entre ellas, Microsoft, Oracle, Sun Microsystems, Intellicorp, IBM, AMD y otras, quienes desarrollaron una notación llamada UML y el proceso de desarrollo RUP.

1.15.1 Lenguaje Unificado de Modelado (UML)

Lenguaje Unificado de Modelado (UML), por sus siglas en inglés, *Unified Modeling Language*) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el OMG (Object Management Group). Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. UML ofrece un estándar para describir un "plano" del sistema (modelo), 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.

Es importante resaltar que UML es un "lenguaje" para especificar y no para describir métodos o procesos. Se utiliza para definir un sistema, para detallar los artefactos en el sistema y para documentar y construir. En otras palabras, es el lenguaje en el que está descrito el modelo.

Se puede aplicar en el desarrollo de software entregando gran variedad de formas para dar soporte a una metodología de desarrollo de software (tal como el Proceso Unificado Racional o RUP), pero no especifica en sí mismo qué metodología o proceso usar.

UML no puede compararse con la programación estructurada, pues UML significa (Lenguaje de Modelación Unificada), no es programación, solo se diagrama la realidad de una utilización en un requerimiento. Mientras que, programación estructurada, es una forma de programar como lo es la orientación a objetos, sin embargo, la orientación a objetos viene siendo un complemento perfecto de UML, pero no por eso se toma UML sólo para lenguajes orientados a objetos

UML cuenta con varios tipos de diagramas, los cuales muestran diferentes aspectos de las entidades representadas.

1.16 Herramientas de modelado de Software

Visual Paradigm para UML es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación.

Entre algunas de sus características se encuentran:

- Soporte de UML versión 2.1
- Diagramas de Procesos de Negocio - Proceso, Decisión, Actor de negocio, Documento
- Modelado colaborativo con CVS y Subversion.
- Ingeniería de ida y vuelta
- Ingeniería inversa: Código a modelo, código a diagrama
- Generación de código - Modelo a código, diagrama a código
- Editor de Detalles de Casos de Uso: Entorno todo-en-uno para la especificación de los detalles de los casos de uso, incluyendo la especificación del modelo general y de las descripciones de los casos de uso.
- Diagramas de flujo de datos.
- Generación de bases de datos: Transformación de diagramas de Entidad-Relación en tablas de base de datos.
- Ingeniería inversa de bases de datos: Desde Sistemas Gestores de Bases de Datos (DBMS) existentes a diagramas de Entidad-Relación.
- Distribución automática de diagramas: Reorganización de las figuras y conectores de los diagramas UML.

Rational Rose está orientado hacia el Lenguaje Unificado de Modelado (UML). Es una herramienta de diseño de software destinado a la modelación y los componentes visuales de empresas de aplicaciones de software. Rational Rose va guardando el diagrama que se está construyendo y, a continuación, genera el código que seleccione el diseñador, que puede ser C + +, Visual Basic, Java, Oracle o CORBA. Dos características de Rational Rose son: su capacidad para proporcionar el desarrollo iterativo y de ingeniería de ida y vuelta, y además permite a los diseñadores tomar ventaja de desarrollo iterativo (a veces llamada desarrollo evolutivo), ya que la nueva aplicación se puede crear por etapas con la salida de una iteración de convertirse en la entrada a la siguiente. (Esto es en contraste con cascada donde el desarrollo se haya completado todo el proyecto de principio a fin antes de que un usuario llegue a probarlo.) Rational Rose puede realizar lo que se llama "ingeniería de ida y vuelta" por regresar y actualizar el resto del modelo para garantizar que el código sigue siendo coherente.

En el capítulo se realizó un estudio de algunas aplicaciones utilizadas en Cuba y el resto del mundo que brindan servicios como la aplicación que se pretende desarrollar. Se hizo un estudio de las principales herramientas y tecnologías utilizadas para el desarrollo de aplicaciones como la que se pretende desarrollar, seleccionando aquellas que mejor se adaptaron a los objetivos de la investigación y por consiguiente a las necesidades del cliente.

CAPÍTULO 2: MODELAMIENTO DEL SISTEMA

El capítulo desarrollado a continuación aborda los principales procesos a implementar mediante un modelo de negocio; haciendo énfasis en los conceptos más importantes y analizándose el software a desarrollar. Se definen además las funcionalidades del sistema, describiéndose detalladamente cómo debe trabajar el mismo a partir de sus requerimientos funcionales y no funcionales. El presente capítulo permitirá obtener una concepción general del sistema.

2.1 Descripción de los procesos del negocio

Actualmente la Dirección de Información de la Universidad de las Ciencias Informáticas cuenta con un servicio de impresión de documentos a estudiantes. Para que un estudiante pueda imprimir un documento debe tener una autorización escrita del jefe de su brigada con la cantidad de hojas que puede imprimir, después tiene que ir al local de impresión en la mañana (donde se va atendiendo a cada estudiante según el orden en que van llegando) con la información guardada en un dispositivo de almacenamiento extraíble. La secretaria de la oficina de impresión comprueba la autorización del jefe de brigada y verifica en la base de datos la cantidad de hojas disponibles que tiene el grupo al cual pertenece el estudiante, si existen hojas disponibles entonces copia el documento y confirma que la cantidad de páginas no sobrepase la cantidad de hojas que está autorizado a imprimir. Por último la secretaria imprime el documento, disminuye el número de hojas disponibles del grupo en la base de datos y finalmente entrega el documento impreso al estudiante.

2.2 Modelo de Negocio

Un **modelo de negocio** (también llamado diseño de negocio) es el mecanismo por el cual un [negocio](#) trata de generar [ingresos](#) y [beneficios](#). Es un resumen de cómo una compañía planifica servir a sus [clientes](#). Implica tanto el concepto de estrategia como el de implementación.

2.2.1 Actor del Negocio

El Actor del Negocio es una entidad externa al negocio y que interactúa con él.

Actor	Descripción
Estudiante	Persona que solicita el servicio de impresión de documentos.

2.2.2 Trabajador del Negocio

Un Trabajador del Negocio representa a un ser humano, software o hardware que desempeña un rol dentro de las realizaciones del Caso de Uso del Negocio. Este trabajador interactúa con entidades y otros trabajadores para que el negocio funcione. [14]

Trabajador	Descripción
Secretaria de oficina de impresión	Persona encargada de prestar el servicio de impresión de documentos.

2.2.3 Diagrama de Casos de Uso del Negocio

Muestra los casos de uso del negocio, trabajadores del negocio, actores del negocio y las interacciones entre ellos relacionadas con los procesos del negocio que se encuentran dentro de la organización y dentro del alcance del sistema que se está planeando realizar.



2.2.4 Casos de Uso del Negocio

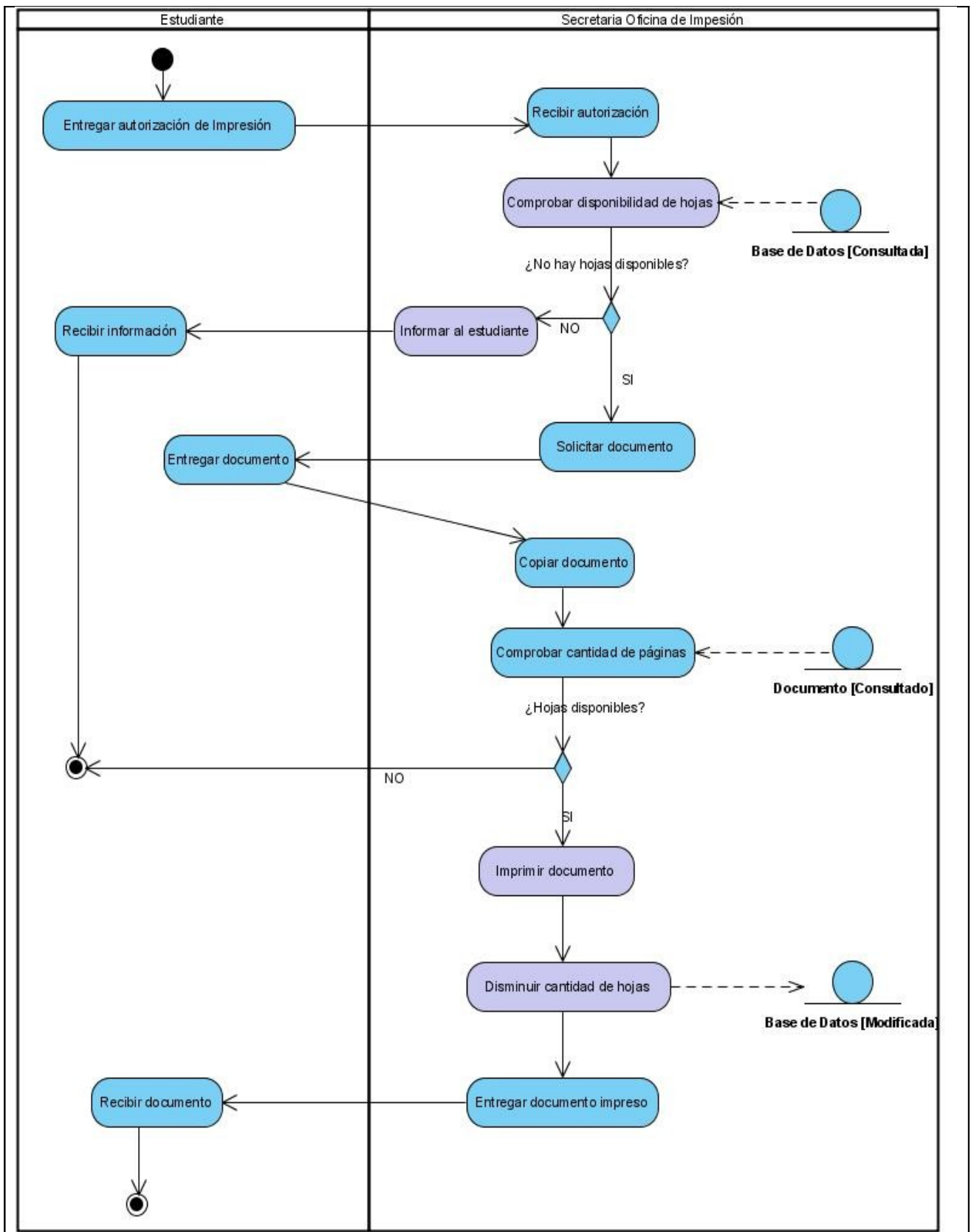
Caso de Uso del Negocio	Solicitar servicio de impresión
Actores	Estudiante
Trabajadores	Secretaria
Resumen	El caso de uso se inicia cuando el estudiante, presentando una autorización previa del jefe de brigada, solicita el servicio de impresión de un documento a la secretaria del local de impresión. La secretaria solicita al estudiante la autorización del jefe de brigada y confirma la disponibilidad de hojas del grupo para proceder a la impresión y luego entrega del documento impreso.
Flujo normal de eventos	

Acción del actor		Respuesta del proceso de negocio
1- El estudiante entrega la autorización del jefe de brigada para solicitar la impresión de un documento.		1.1 La secretaria introduce en la aplicación el número del grupo y comprueba que existen hojas disponibles. (Ver Flujo Altern01) 1.2 La secretaria solicita el dispositivo para copiar el documento a imprimir.
2- El estudiante le entrega el dispositivo de almacenamiento.		2.1 La secretaria copia el documento en la computadora. 2.2 La secretaria comprueba que la cantidad de hojas a imprimir no exceda el total de hojas disponibles. (Ver flujo Altern02) 2.3 La secretaria introduce el nombre del estudiante en la aplicación e imprime el documento. 2.4 La secretaria introduce la cantidad de hojas utilizadas en la base de datos y disminuye del total. 2.5 La secretaria entrega el dispositivo y el documento impreso.
Flujos alternos		
Acción del actor		Respuesta del proceso de negocio
2- El estudiante le entrega el dispositivo de almacenamiento.		2.6 La secretaria comprueba que no hay hojas disponibles o que imprimir exceden el total de hojas disponibles y le informa al estudiante que no se puede imprimir el documento. 2.7 Se termina el caso de uso.
Prioridad	Crítico	

Tabla 1: Descripción del Caso de Uso Solicitar impresión.

2.2.5 Diagrama de actividades del Caso de Uso Solicitar servicio de impresión.

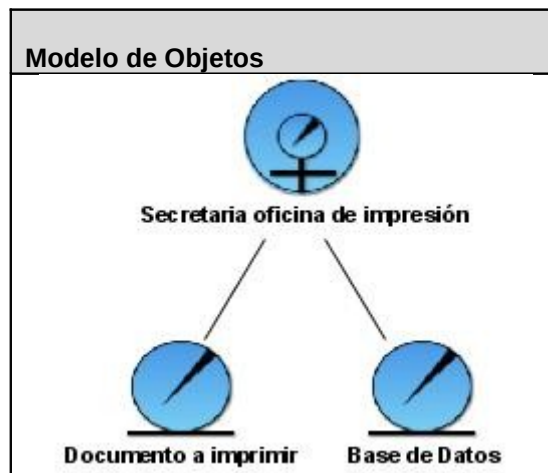
Diagrama de actividades del Caso de Uso Solicitar servicio de impresión.



2.2.6 Modelo de Objetos

Este diagrama permite representar las entidades con las que se relaciona la secretaria de la

oficina de impresión (trabajador del negocio) que son el documento a imprimir y una base de datos que contiene toda la información referente a la cantidad de hojas disponibles que tiene cada grupo.



2.3 Definición de los actores del sistema

Actor	Justificación
Administrador del sistema	Persona encargada de la gestión de contenidos y de usuarios, roles y permisos del sistema (crear, eliminar, modificar)
Secretaria de oficina de impresión	Es la encargada de dar respuesta a las solicitudes hechas al servicio de impresión (imprimir o eliminar el documento)
Estudiante	Persona que solicita el servicio de impresión.
Jefe de entidad	Es el responsable de autorizar las solicitudes de impresión realizadas por los estudiantes (puede ser un jefe de brigada o un presidente de la FEU).

2.4 Especificación de requisitos

La Especificación de Requisitos Software (ERS) es una descripción completa del comportamiento del sistema que se va a desarrollar. Los requisitos no funcionales son requisitos que imponen restricciones en el diseño o la implementación (Como por ejemplo restricciones en el diseño o estándares de calidad).

La captura de requisitos es una parte esencial del proceso de Ingeniería de Software, evita cambios posteriores en el sistema y facilita el entendimiento con el cliente. Para la captura de requisitos se analiza iterativamente el problema documentando los resultados y probando la exactitud del conocimiento adquirido.

2.4.1 Requisitos funcionales del sistema.

R1- Gestionar usuario

R1.1 Adicionar usuario

R1.2 Modificar datos de usuario

R1.3 Eliminar usuario

R1.4 Buscar usuario

R2- Gestionar rol

R2.1 Crear rol

R2.2 Eliminar rol

R2.3 Asignar permisos

R2.4 Cambiar permisos

R3- Autenticar usuario.

R4- Solicitar autorización.

R5- Autorizar impresión.

R6- Mostrar información.

R7- Enviar solicitud de impresión.

R8- Imprimir documentos.

R9- Gestionar jefes de entidad.

R10- Asignar cuota.

R11- Gestionar entidades.

R12- Consumo de hojas.

R13- Listado de consumo.

R14- Mostrar pedidos.

R15- Buscar información.

R16- Enviar notificación.

2.4.2 Requisitos no funcionales del sistema.

Apariencia o interfaz externa:

- El sistema debe contar con una navegación sencilla.
- Diseño sencillo, con pocas entradas, permitiendo que no sea necesario mucho entrenamiento para utilizar el sistema.
- Diseño encuadrado para la resolución 800x600, y preparado para verse en otras resoluciones.

Usabilidad:

- El sistema podrá ser usado por cualquier usuario que posea conocimientos básicos en el manejo de computadoras y de ambientes Web.

Soporte:

- Servidor Web Apache v2
- SGBD MySQL 5.0
- Versión de PHP 5.2.0
- CMS Drupal 5.2
- Por parte del cliente se requiere un navegador capaz de interpretar JavaScript y CSS.
- El sistema debe dar la posibilidad de ser mejorado, así como de incorporarle nuevos servicios en caso de ser necesarios.
- Las pruebas realizadas al sistema deben permitir evaluar sus ventajas y funcionalidades, además de detectar los errores que presenta.

Portabilidad:

- Necesidad de que el sistema sea multiplataforma.

Rendimiento:

- Garantizar velocidad de respuesta.
- Tener base de datos normalizada, para garantizar la integridad de la información y reducir los tiempos de respuesta.
- Permitir numerosas conexiones simultáneas.

Seguridad:

- Autenticar al usuario antes de que pueda realizar cualquier acción sobre el contenido del portal.
- Garantizar que las funcionalidades del sistema se muestren de acuerdo al rol del usuario que este activo.
- Denegar acciones no autorizadas que puedan afectar la integridad de los datos y el sistema.

Confiabilidad:

- La información contenida en el sistema debe ser totalmente confiable.
- La herramienta de implementación a utilizar tiene soporte para recuperación ante fallos y errores.

Legales:

- El sistema pertenece al proyecto Informatización de la Dirección de Información, el cual luego será implantado en la Universidad de las Ciencias Informáticas. La plataforma escogida para el desarrollo de la aplicación está basada en licencia de software libre GNU/ GPL.

2.5 Listado de Casos de Uso del Sistema.

CU - 1	Gestionar usuario
Actor	Administrador del sistema
Descripción	Con este caso de uso el administrador interactúa con el sistema para adicionar, eliminar, modificar o buscar algún usuario del sistema.
Referencia	R1

CU - 2	Gestionar rol
Actor	Administrador del sistema
Descripción	El administrador accede al sistema para realizar las opciones adicionar, eliminar, modificar o buscar roles al cual pertenecen cada uno de los usuarios del sistema.
Referencia	R2

CU - 3	Autenticar usuario
Actores	Administrador del sistema, Estudiante, Secretaria de oficina de impresión, Jefe de entidad.
Descripción	El caso de uso se inicia cuando un usuario es decir, cualquiera de los actores del sistema intenta acceder a este. El sistema les muestra el formulario para entrar los datos de acceso (usuario y contraseña).
Referencia	R3

CU - 4	Solicitar autorización
Actor	Estudiante
Descripción	El estudiante realiza la solicitud al jefe de entidad que desea imprimir un documento y la cantidad de hojas que necesita que este autorice.
Referencia	R4

CU - 5	Autorizar impresión
Actor	Jefe de entidad
Descripción	El jefe de entidad revisa las solicitudes de impresión de los estudiantes y autoriza la cantidad de hojas que estime a cada estudiante de acuerdo a la cantidad de hojas disponibles con las que cuenta el grupo o la FEU.

Referencia	R5
-------------------	-----------

CU - 6	Enviar solicitud de Impresión
Actor	Estudiante
Descripción	El estudiante, cuando ya está autorizado a imprimir sube al sitio el documento que desea con el formato en que quiere que este sea impreso, así como la cantidad de copias, colocándose este en el listado de documentos en espera de impresión.
Referencia	R5, R7

CU - 7	Mostrar información
Actor	Jefe de entidad
Descripción	El jefe de entidad selecciona la opción mostrar información y el sistema le muestra los datos relacionados con el consumo mensual de hojas de su entidad.
Referencia	R6

CU - 8	Imprimir Documentos
Actor	Secretaria de oficina de impresión.
Descripción	La secretaria de la oficina de impresión accede al listado de documentos por imprimir y descarga el documento para ser impreso, si este tiene el formato correcto entonces lo imprime, si no lo elimina de la lista de documentos a imprimir.
Referencia	R7, R8

CU - 9	Gestionar jefe de entidad
Actor	Administrador del sistema
Descripción	El administrador accede al sistema y de esta manera podrá adicionar o modificar los jefes de entidad.
Referencia	R9

CU - 10	Gestionar entidad
Actor	Administrador del sistema
Descripción	El administrador del sistema puede crear modificar o eliminar las diferentes entidades a las que pertenecerán los usuarios del sistema.
Referencia	R11

CU - 11	Asignar cuota
Actor	Administrador del sistema

Descripción	El administrador del sistema podrá establecer el consumo mensual de hojas para las entidades.
Referencia	R10

CU - 12	Consumo de hojas
Actor	Administrador del sistema
Descripción	El sistema muestra una serie de estadísticas relacionadas con el consumo anual y mensual de las hojas.
Referencia	R12

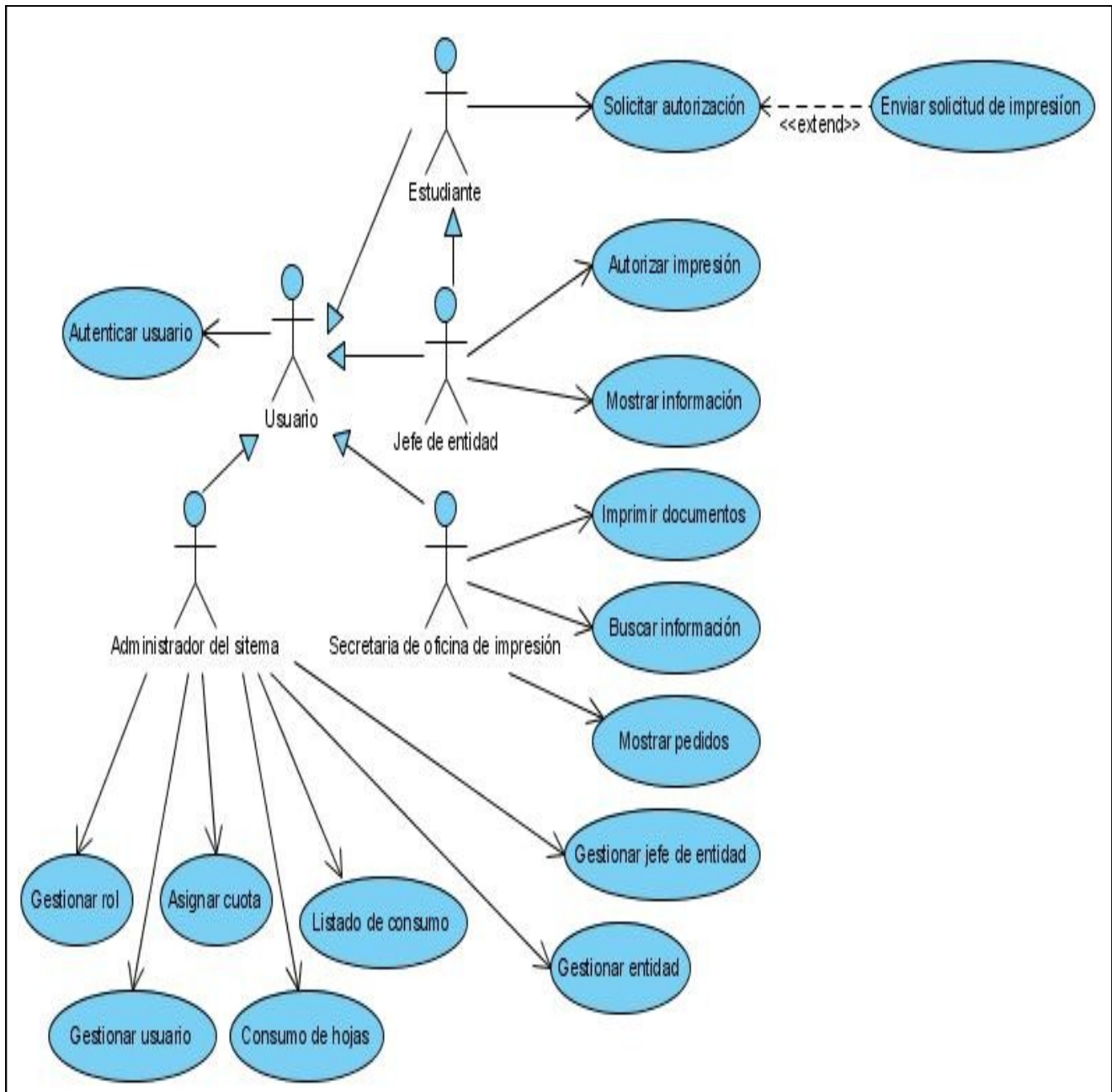
CU - 13	Listado de consumo
Actor	Administrador del sistema
Descripción	El sistema muestra un listado con el consumo mensual de las entidades por meses, representando el consumo y el % que representa.
Referencia	R13

CU - 14	Mostrar pedidos
Actor	Secretaria de oficina de impresión.
Descripción	El caso de uso se inicia cuando la secretaria desea ver el listado de los pedidos de impresión (impresiones realizadas) hasta la fecha.
Referencia	R14

CU - 18	Buscar información
Actor	Secretaria de oficina de impresión.
Descripción	El caso de uso se inicia cuando la secretaria solicita al sistema saber la cantidad de hojas impresas y las que faltan por imprimir de cada grupo.
Referencia	R10

2.5.1 Modelo de Casos de Uso de Sistema

Diagrama de Casos de Uso de Sistema



2.6 Descripción de los principales Casos de Usos.

En esta sección se mostrará la descripción de los casos de uso principales, describiéndose detalladamente la secuencia de eventos que los actores del sistema utilizan para llevar a cabo un proceso determinado a través del sistema. Para ver el resto de los casos de uso ver Anexo 1.

2.6.1 Gestionar usuario

Caso de Uso:	Gestionar usuario
Actores:	Administrador del sistema

Propósito:	Poder tener distintos tipos de usuario dentro del sistema.
Resumen:	El caso de uso se inicia cuando el administrador interactúa con el sistema para adicionar, eliminar, modificar o buscar algún usuario del sistema.
Precondiciones:	El administrador del sistema deberá estar autenticado previamente para la realización de algunas de estas acciones.
Referencias	R1
Prioridad	Crítico

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1-El administrador selecciona la opción Gestionar usuario.	1.1 El sistema le muestra las diferentes funcionalidades para la gestión de los usuarios.
2- El administrador selecciona la opción para realizar la operación deseada: <ul style="list-style-type: none"> • Adicionar Usuario • Modificar Usuario • Eliminar Usuario • Buscar Usuario 	2.1- Si el administrador selecciona la opción: <ul style="list-style-type: none"> - Adicionar Usuario ver la sección “Adicionar Usuario” - Modificar Usuario ver la sección “Modificar Usuario” - Eliminar Usuario ver la sección “Eliminar Usuario” - Buscar Usuario ver la sección “Buscar Usuario”

Sección “Adicionar Usuario”

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1- El administrador introduce los datos del usuario. <ul style="list-style-type: none"> • usuario • rol • permisos 	1.1-El sistema comprueba que el usuario no exista en la base de datos. En dicho caso procede a insertar el usuario.

Flujos Alternos

Acción del Actor	Respuesta del Negocio
1- El administrador introduce los datos del usuario. Usuario, rol y permisos.	1.2- El sistema notifica que el usuario ya existe en la base de datos. Termina el caso de uso.

Sección “Modificar Usuario”

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1- El administrador selecciona el usuario al que desea modificar los datos.	1.1-El sistema muestra los datos del usuario.
2- El administrador modifica los datos deseados.	2.1-El sistema actualiza los datos del usuario.

Sección "Eliminar Usuario"	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1-El administrador selecciona el usuario que desea eliminar	1.1-El sistema elimina el usuario de la base de datos.
Sección "Buscar Usuario"	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1-El administrador introduce los datos para realizar la búsqueda.	1.1-El sistema realiza la búsqueda en la base de datos y muestra los datos del usuario
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
1-El administrador introduce los datos para realizar la búsqueda.	1.2-El sistema no encuentra el usuario en la base de datos y notifica que ese usuario no existe.
Postcondiciones	Los datos del usuario quedan actualizados.

2.6.2 Gestionar rol

Caso de Uso:	Gestionar rol
Actores:	Administrador del sistema
Propósito:	Una vez creados los usuarios, poder asignarle diferentes permisos de acceso y configuración a cada tipo.
Resumen:	El caso de uso se inicia cuando el administrador accede al sistema para realizar las opciones adicionar, eliminar, modificar o buscar roles al cual pertenecen cada uno de los usuarios del sistema.
Precondiciones:	El administrador del sistema deberá estar autenticado previamente para la realización de algunas de estas acciones.
Referencias	R2
Prioridad	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1-El administrador selecciona la opción Gestionar Rol.	1.1 El sistema le muestra las diferentes funcionalidades para la gestión de los roles de usuarios.
2- El administrador selecciona la opción para realizar la operación deseada: - Crear Rol. - Eliminar Rol.	2.1- Si el administrador selecciona la opción: -Crear Rol ver la sección "Adicionar Rol" -Eliminar Rol ver la sección "Eliminar Rol" -Asignar Permisos ver la sección "Asignar Permisos"

- Asignar Permisos - Cambiar Permisos	-Cambiar Permisos ver la sección "Cambiar Permisos"
Sección "Crear Rol"	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
2- El administrador accede a la opción Crear Rol.	2.1 El sistema muestra el formulario con los datos (nombre, permisos del rol) a llenar.
3- El administrador llena el formulario y envía los datos (nombre, permisos del rol).	3.1 El sistema comprueba que los campos del formulario (nombre, permisos del rol) se hayan llenado correctamente. 3.2 El sistema crea el rol.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
3- El administrador no envía los datos del formulario (nombre, permisos del rol) y presiona la opción "Cancelar".	3.3 a El sistema muestra un mensaje "Los datos del formulario Crear Roles no fueron llenados satisfactoriamente". Retorna a la acción 2.
Sección "Eliminar Rol"	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
2- El administrador accede a la opción "Listar Roles"	2.1 El sistema muestra un listado de los roles existentes.
3- El administrador selecciona el rol que desea eliminar.	3.1 El sistema muestra un mensaje para confirmar que se desea eliminar el rol seleccionado.
4- El administrador presiona la opción "Eliminar"	4.1 El sistema elimina el rol. 4.2 El sistema muestra el listado de los roles actualizado.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
5- El administrador presiona la opción "Cancelar"	4.3 El sistema redirecciona al administrador hacia la página donde se encuentran las opciones de los roles. Se termina el caso de uso
Sección "Asignar Permisos"	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
2- El administrador selecciona la opción "Asignar Permisos".	2.1 El sistema comprueba que ya existen roles y muestra el listado de roles.
3- El administrador selecciona el rol deseado para realizar la opción "Asignar Permisos".	3.1 El sistema le muestra los permisos que pueden ser asignados a los roles.

4- El administrador marca los permisos que serán asignados al rol y presiona la opción "Asignar"	4.1 El sistema asigna los permisos.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
2- El administrador selecciona la opción "Asignar Permisos".	2.2 El sistema comprueba que no existen roles y muestra un mensaje.
Sección "Cambiar Permisos"	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
2- El administrador selecciona la opción "Cambiar Permisos"	2.1 El sistema muestra el listado de los roles y los permisos definidos.
3- El administrador realiza los cambios en los permisos y presiona la opción "Aceptar"	3.1 El sistema cambia los permisos.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
Postcondiciones	Los roles y permisos son actualizados.

2.6.3 Autenticar usuario

Caso de Uso:	Autenticar usuario
Actores:	Administrador del sistema, estudiante, secretaria de oficina de impresión, jefe de grupo.
Propósito:	Permite a los usuarios autenticarse en el sistema; esto permitirá acceder a las opciones a las que puede acceder cada usuario
Resumen:	El caso de uso se inicia cuando un usuario es decir, cualquiera de los actores del sistema intenta acceder al sistema. El sistema les muestra el formulario para entrar los datos de acceso(usuario, contraseña)
Precondiciones:	El sistema debe estar funcionando.
Referencias	R3
Prioridad	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1- El usuario accede al sistema poniendo su nombre de usuario y su contraseña.	1.1 El sistema verifica que el nombre de usuario y la contraseña son correctamente escritos.

	1.2 El sistema permite al actor acceder a las funcionalidades a las que está autorizado según su rol.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
1- El usuario accede al sistema poniendo su nombre de usuario y su contraseña.	1.3 El sistema verifica que el nombre de usuario y la contraseña están escritos incorrectamente 1.4 El sistema envía una notificación informando que el nombre de usuario o la contraseña está incorrecto. 1.5 El sistema solicita que sean introducidos nuevamente los datos (usuario, contraseña).
Postcondiciones	El usuario se ha autenticado y se habilitan las funcionalidades según los permisos asignados.

2.6.4 Solicitar autorización.

Caso de Uso:	Solicitar autorización
Actores:	Estudiante
Propósito:	Que el estudiante sea autorizado a imprimir un documento.
Resumen:	El Caso de Uso se inicia cuando el estudiante accede al sistema y envía una solicitud al jefe de entidad (jefe de brigada o presidente de la FEU) con la cantidad de hojas que desea imprimir.
Precondiciones:	El estudiante debe estar autenticado en el sistema
Referencias	R4
Prioridad	Crítico

Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1- El estudiante accede al sistema.	1.1 El sistema muestra las diferentes vías por las que puede realizar la solicitud (Brigada o FEU).
2- El estudiante selecciona una de las vías para realizar la solicitud.	2.1 El sistema le muestra la cantidad de hojas que están disponibles para imprimir.
3- El estudiante solicita la cantidad de hojas que desea imprimir.	3.1 El sistema envía la solicitud de impresión solicitada al Jefe de entidad correspondiente (Jefe de brigada o presidente de la FEU).
Flujos Alternos	
Acción del Actor	Respuesta del Sistema

Postcondiciones	Solicitud de impresión enviada al Jefe de entidad

2.6.5 Autorizar impresión

Caso de Uso:	Autorizar impresión
Actores:	Jefe de entidad
Propósito:	Autorizar la cantidad de hojas que va a imprimir el estudiante.
Resumen:	El jefe de entidad accede al sistema para autorizar al estudiante a imprimir.
Precondiciones:	El jefe de entidad debe estar autenticado en el sistema.
Referencias	R5
Prioridad	Crítico

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1- El jefe de entidad accede al sistema.	1.1 El sistema le muestra la cantidad de hojas disponibles y la cantidad de hojas que el estudiante solicitó para imprimir.
2- El jefe de entidad selecciona el nombre del estudiante que desea autorizar a imprimir.	2.1 El sistema permite cambiar la cantidad de hojas que desea autorizar a imprimir.
3- El jefe de entidad autoriza la cantidad de hojas solicitadas.	3.1 El sistema pone en espera de impresión la cantidad de hojas indicada.

Flujos Alternos

Acción del Actor	Respuesta del Sistema
4- El jefe de grupo cancela la solicitud.	4.1 El sistema elimina la solicitud de la lista de solicitudes que tiene el jefe de brigada. Termina el Caso de Uso.
Postcondiciones	La solicitud de impresión queda registrada y la cantidad de hojas añadidas en espera de impresión.

2.6.6 Enviar solicitud de impresión

Caso de Uso:	Enviar solicitud de impresión.
Actores:	Estudiante

Propósito:	Que el estudiante pueda enviar a imprimir un documento.	
Resumen:	El Caso de Uso se inicia cuando el estudiante (previamente autorizado por el Jefe de entidad) accede al sistema y envía a imprimir un documento.	
Precondiciones:	El estudiante debe estar autenticado en el sistema y debe haber sido autorizado por el jefe de entidad.	
Referencias	R5, R7	
Prioridad	Crítico	
Flujo Normal de Eventos		
	Acción del Actor	Respuesta del Sistema
	1- El estudiante accede al sistema.	1.1 El sistema muestra la cantidad de hojas que le han sido autorizadas a imprimir por el Jefe de entidad, el formato en que desea imprimir el documento y la cantidad de copias que desea copiar.
	2- El estudiante sube el documento y selecciona las opciones de formato y copias.	3.1 El sistema ingresa el documento a la lista de documentos por imprimir. 3.2 El sistema elimina la cantidad de hojas disponibles del grupo al que pertenece el estudiante.
Flujos Alternos		
	Acción del Actor	Respuesta del Sistema
	1- El estudiante accede al sistema.	1.2 El sistema no le muestra las opciones para mandar a imprimir el documento. Termina el caso de uso.
Postcondiciones	Documento añadido a la lista de documentos a imprimir	

2.6.7 Imprimir documentos.

Caso de Uso:	Imprimir documentos
Actores:	Secretaria de oficina de impresión
Propósito:	Imprimir documentos a los estudiantes.
Resumen:	El caso de uso se inicia cuando la secretaria verifica la lista de documentos que están por imprimir y procede a imprimir de uno en uno empezando por el primero de la lista.
Precondiciones:	Deben existir documentos en la lista de documentos a imprimir.
Referencias:	R8,R16
Prioridad:	Crítico

Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1-La secretaria de la oficina de impresión accede al sistema.	1.1 El sistema le muestra una lista con los documentos a imprimir, dándole la posibilidad de poder descargar el documento que se va a imprimir.
2-La secretaria de la oficina de impresión descarga el documento a imprimir, revisa el formato y si este es correcto imprime el documento y selecciona la opción enviar.	2.1 El sistema envía una notificación al estudiante que solicitó la impresión diciendo que su documento ya está impreso. 2.2 El sistema elimina el documento de la lista de documentos por imprimir y lo pasa a la lista de documentos impresos.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
2-La secretaria de la oficina de impresión descarga el documento a imprimir, revisa el formato y si este es incorrecto elimina el documento de la lista de documentos por imprimir.	2.4 El sistema envía una notificación al estudiante que solicitó el servicio de impresión diciendo que no se pudo imprimir el documento. 2.5 El sistema vuelve a poner en la cantidad de hojas disponibles la cantidad solicitada por dicho estudiante.
Postcondiciones	Se imprime el documento y es notificado el estudiante.

En este capítulo se planteó la propuesta de solución del sistema a partir del análisis de los procesos del negocio. Se especificaron los requisitos funcionales y no funcionales de los cuales dependerá el mismo y se realizó una descripción detallada de sus funcionalidades. Finalmente se describieron paso a paso todas las acciones de los actores del sistema con los casos de uso con los que interactuaban. De esta manera se pretende una mejor comprensión de cómo funcionará el sistema.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

El presente capítulo tiene como principal objetivo modelar los artefactos que tienen lugar durante el flujo de trabajo de análisis y diseño. Se describirá detalladamente la propuesta del sistema a implementar a partir de los requisitos funcionales mediante el análisis y el diseño. De esta forma se obtiene un modelo de clases del análisis por cada caso de uso significativo, y se realizará el diseño para perfeccionar los modelos obtenidos del análisis partiendo también de los requisitos no funcionales, representado en diagramas de clases del diseño y diagramas de interacción respectivamente.

3.1 ¿Qué es el modelo de análisis?

Cuando se analiza un problema se estudian detalladamente las características del mismo antes de comenzar a plantear posibles soluciones mediante una aplicación informática. En la etapa de análisis se expresa la estructura de solución, centrándose en las necesidades del usuario. El modelo de análisis es una aproximación del modelo de diseño. En él se refinan los requisitos, con el objetivo de comprenderlos perfectamente, sin embargo no se tiene en cuenta el lenguaje en el que se implementará la aplicación.

A continuación, se describen las clases que serán utilizadas en la realización de los diagramas de clases del análisis de cada caso de uso.

CI_< Nombre de la clase >: estas clases modelan la interacción entre los actores y el sistema.

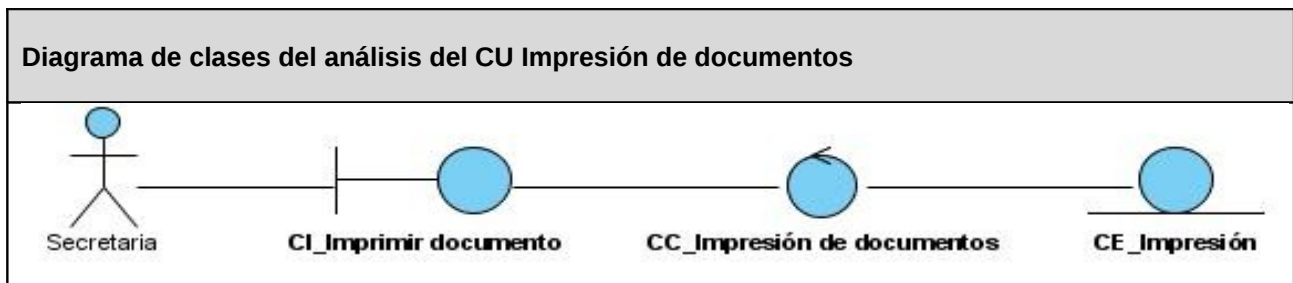
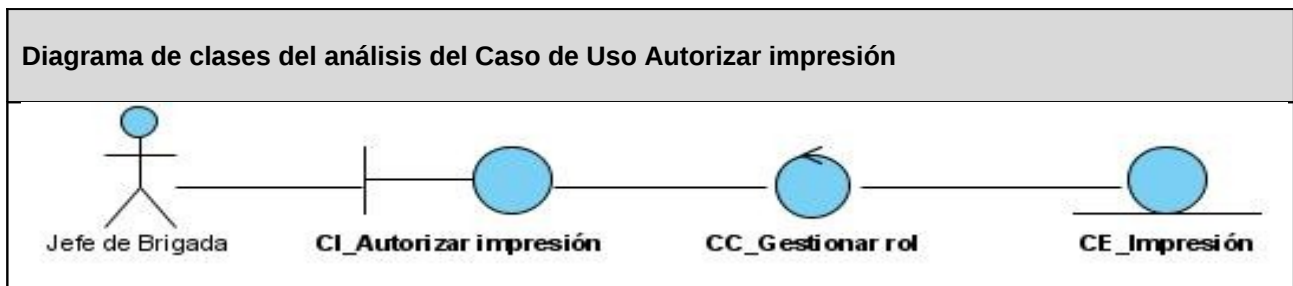
CC_< Nombre de la clase >: estas clases coordinan la realización de los casos de uso y además, coordinan el trabajo de las clases interfaz y entidad.

CE_< Nombre de la clase >: estas clases modelan información que posee larga vida y que es a menudo persistente.

3.2 Diagramas de clases del análisis

A continuación se presentan los principales diagramas de clases del análisis, para ver el resto de los diagramas ver el Anexo 2.





3.3 Modelo del diseño.

El modelo de diseño es un proceso en el que se aplican técnicas y principios con el objetivo de definir un sistema con suficientes detalles, de manera que pueda ser interpretado e implementado posteriormente. En él se crean las estructuras de datos necesarios para implementar el software, se define la relación entre cada uno de los elementos estructurales del programa. Después del modelo, en el diseño se encuentra la forma para que el sistema soporte todos los requisitos, incluyendo los no funcionales y las diferentes restricciones.

3.3.1 Diagramas de clases del diseño

En el siguiente epígrafe se muestran los diagramas de clases del diseño de los casos de uso más críticos. Para ver el resto de los diagramas ver el Anexo 3.

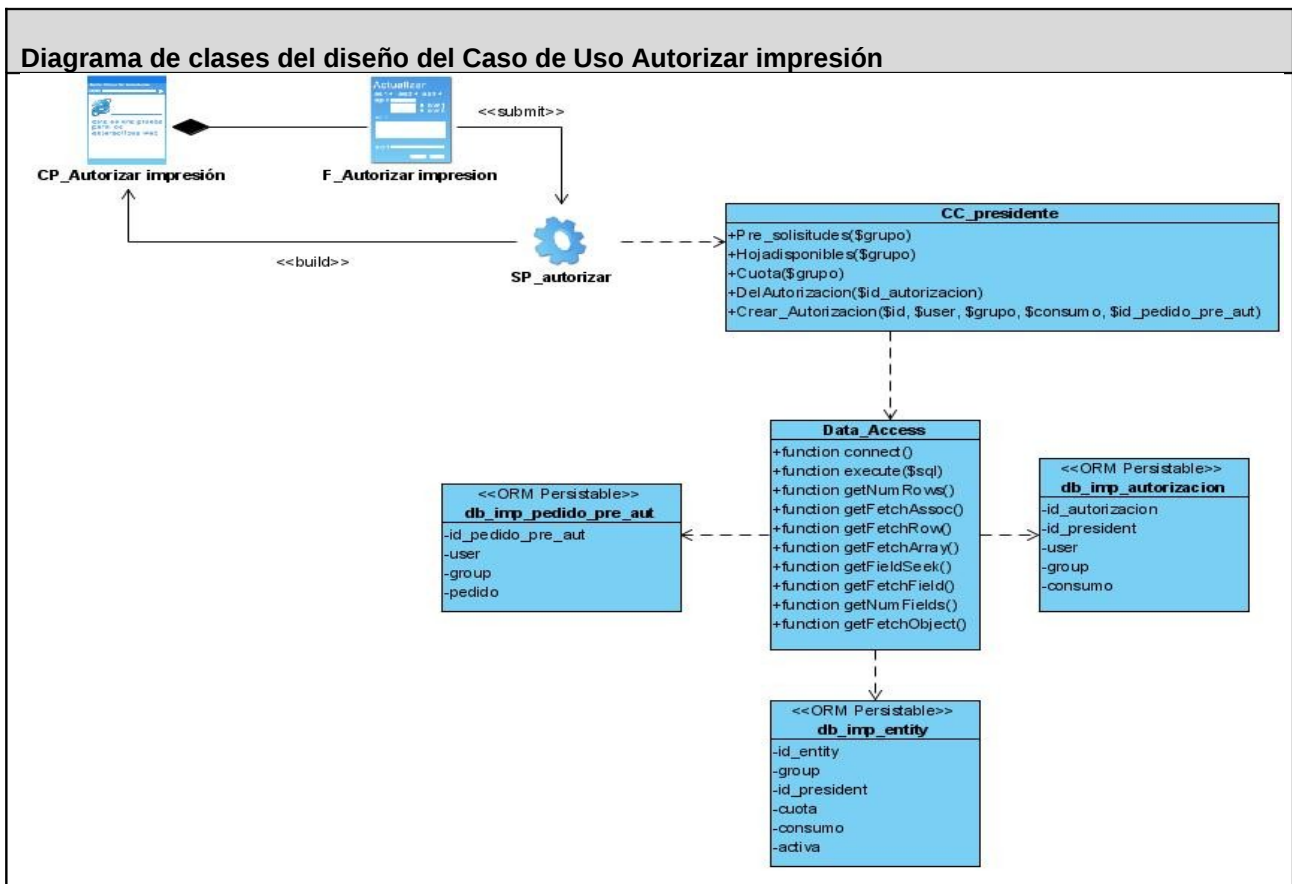
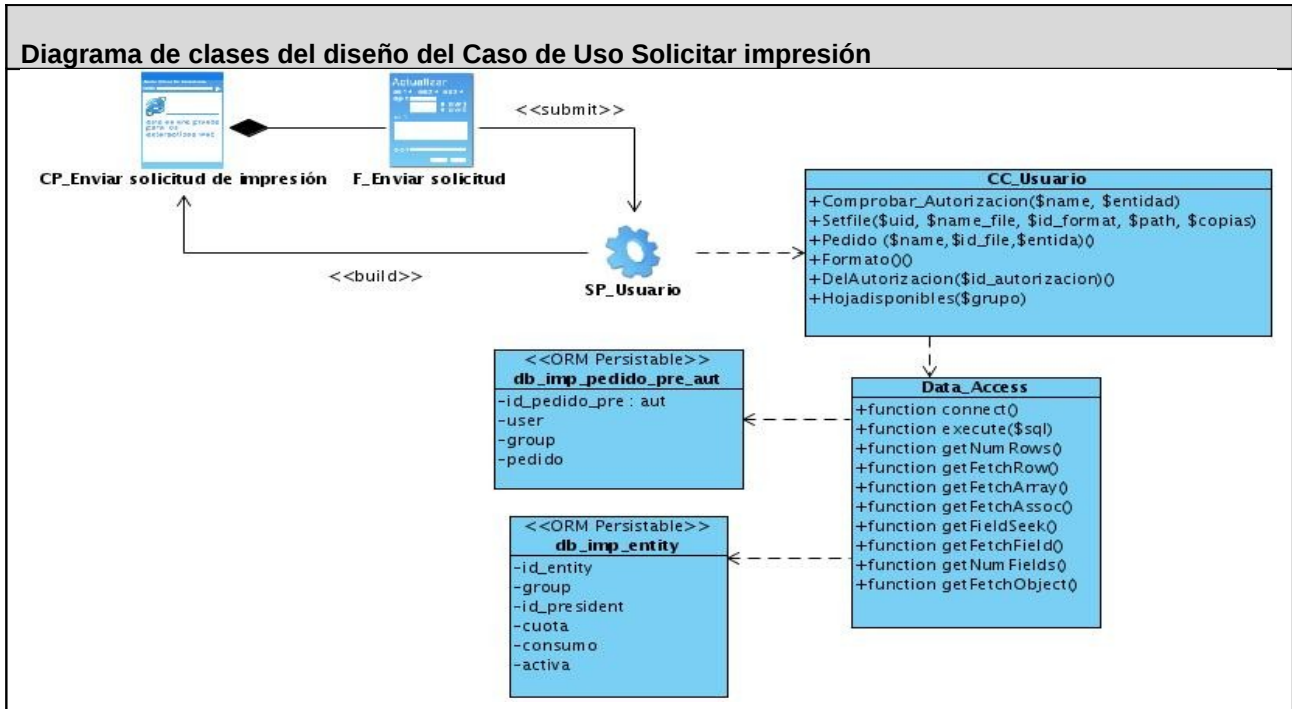
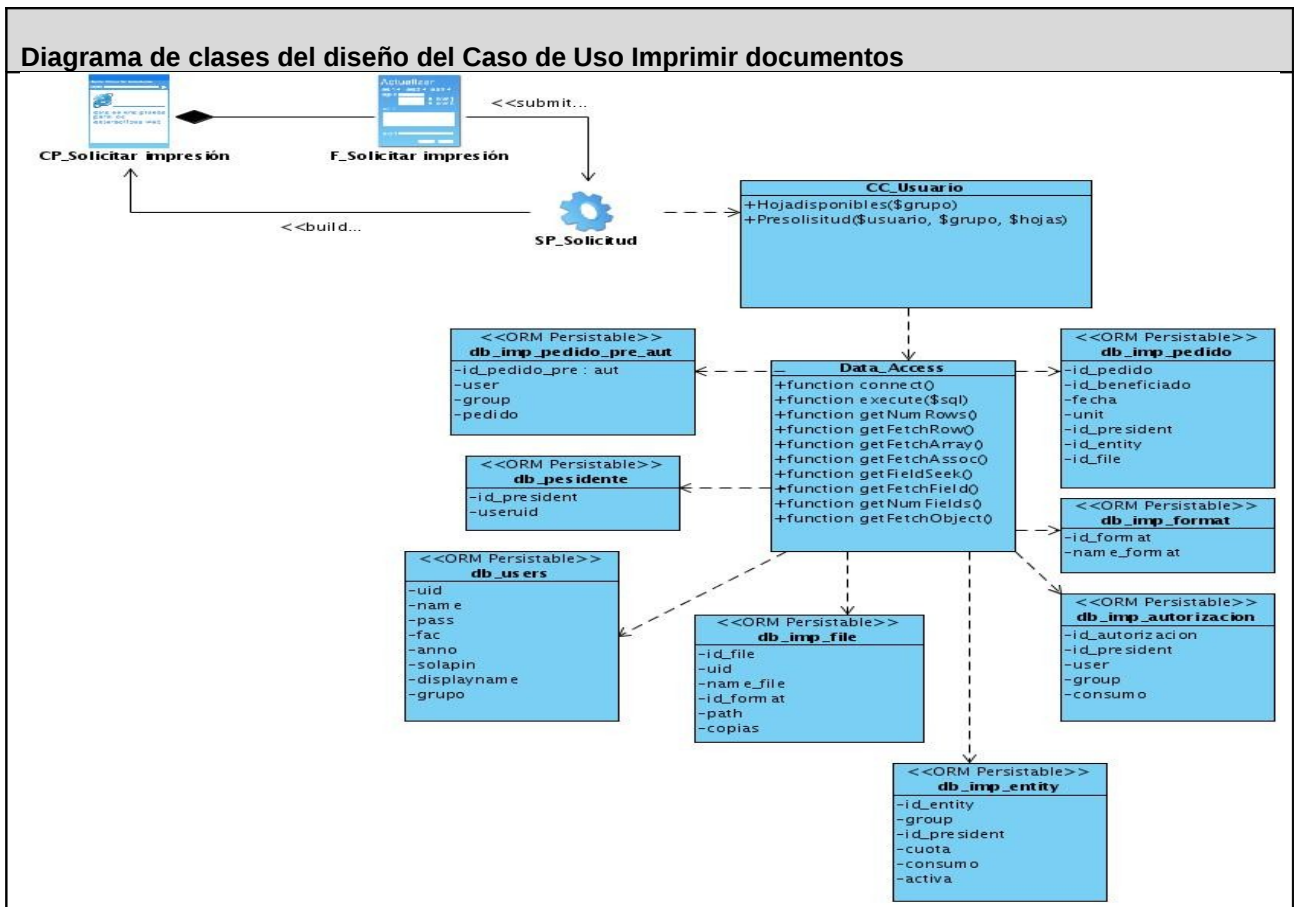
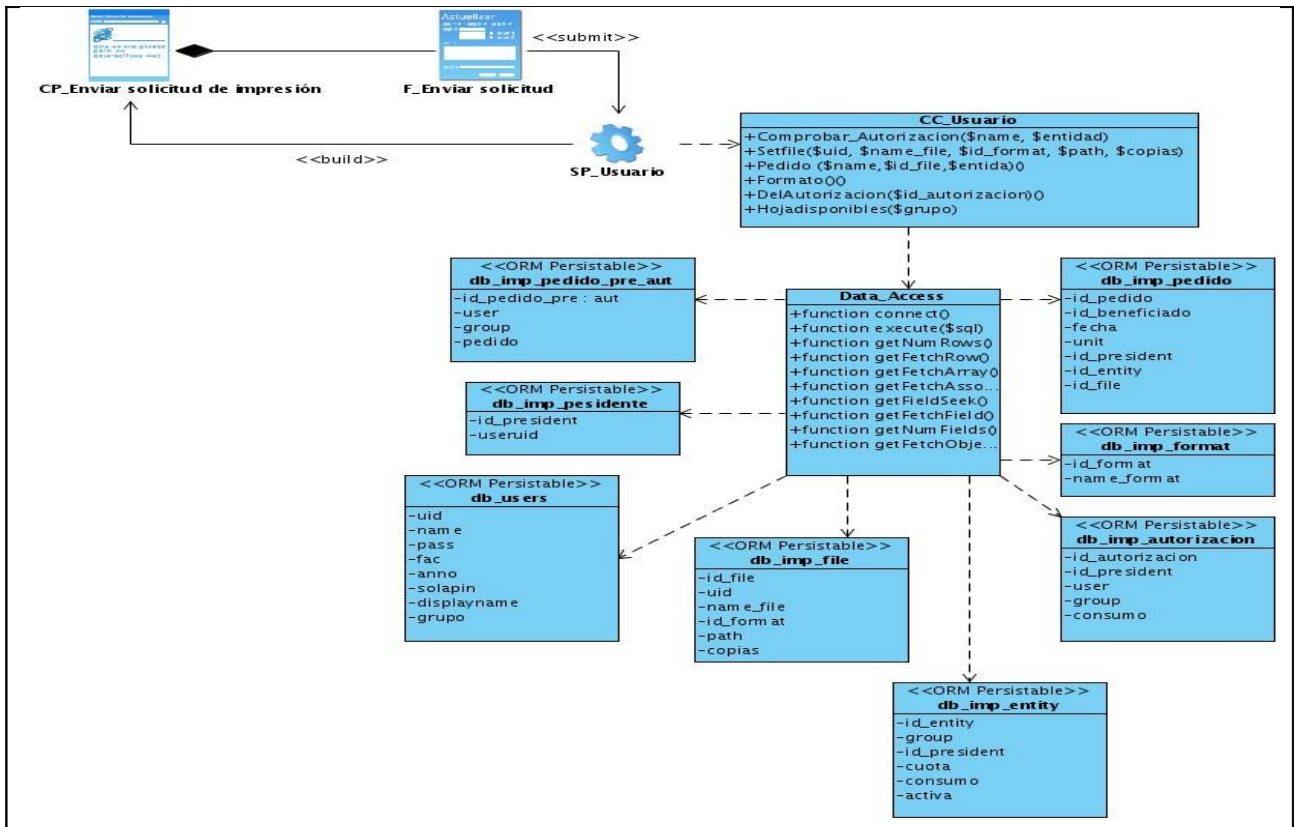


Diagrama de clases del diseño del Caso de Uso Enviar solicitud de impresión



3.3.2 Diagramas de Interacción.

Los diagramas de interacción se utilizan para modelar los aspectos dinámicos de un sistema. Pueden servir para visualizar, especificar, construir y documentar los aspectos dinámicos de una sociedad particular de objetos, o pueden ser usados para modelar un flujo particular de control de un caso de uso. A continuación se presentan los principales diagramas de interacción, para más detalles ver el Anexo 4.

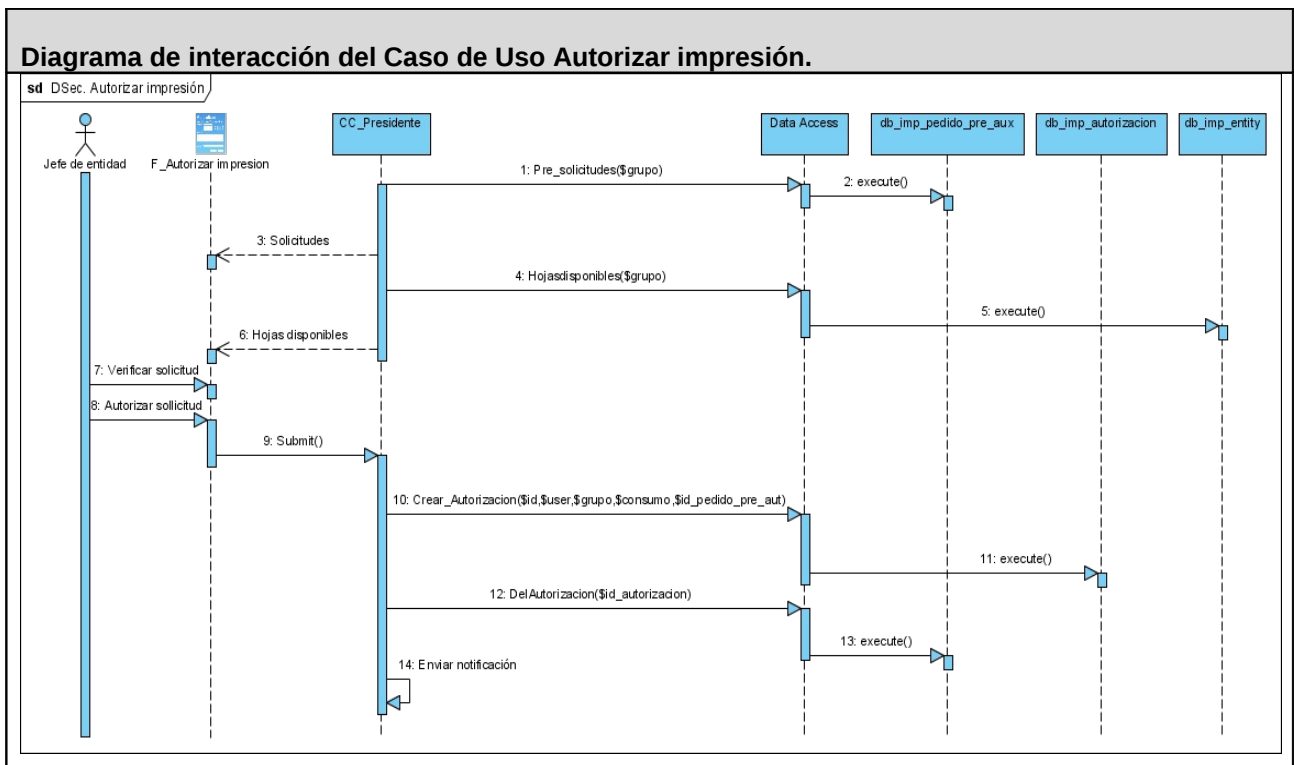
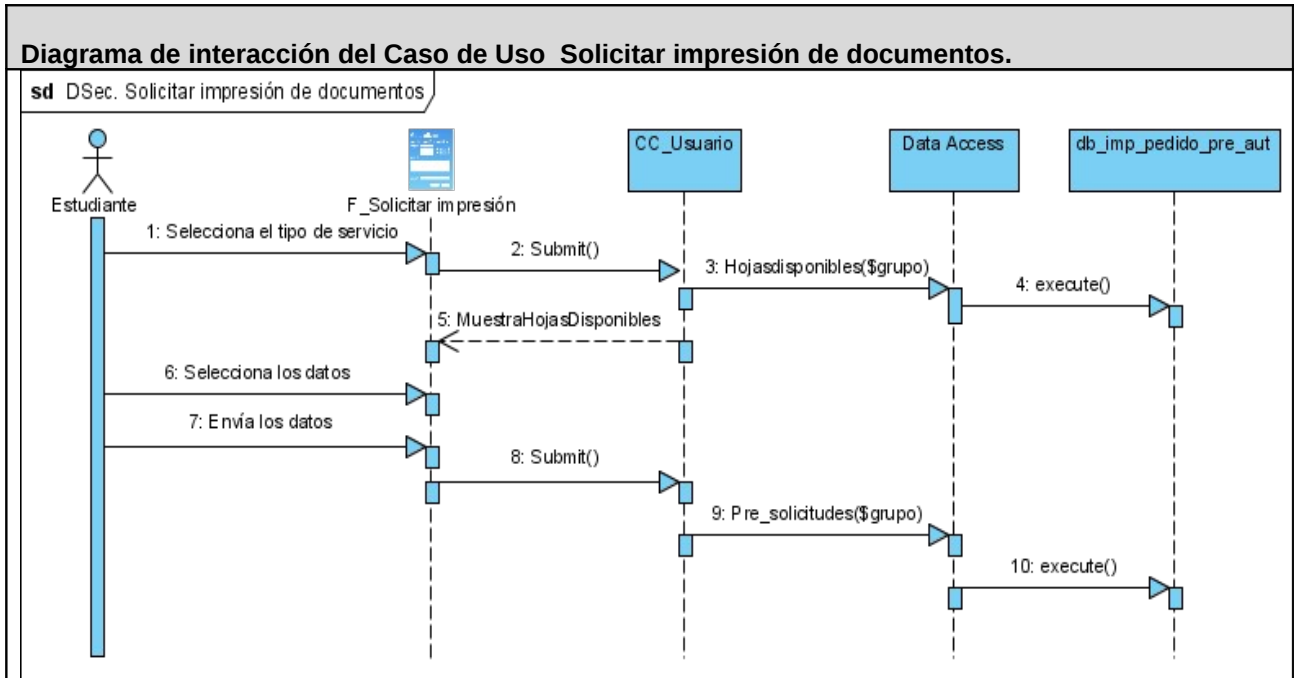
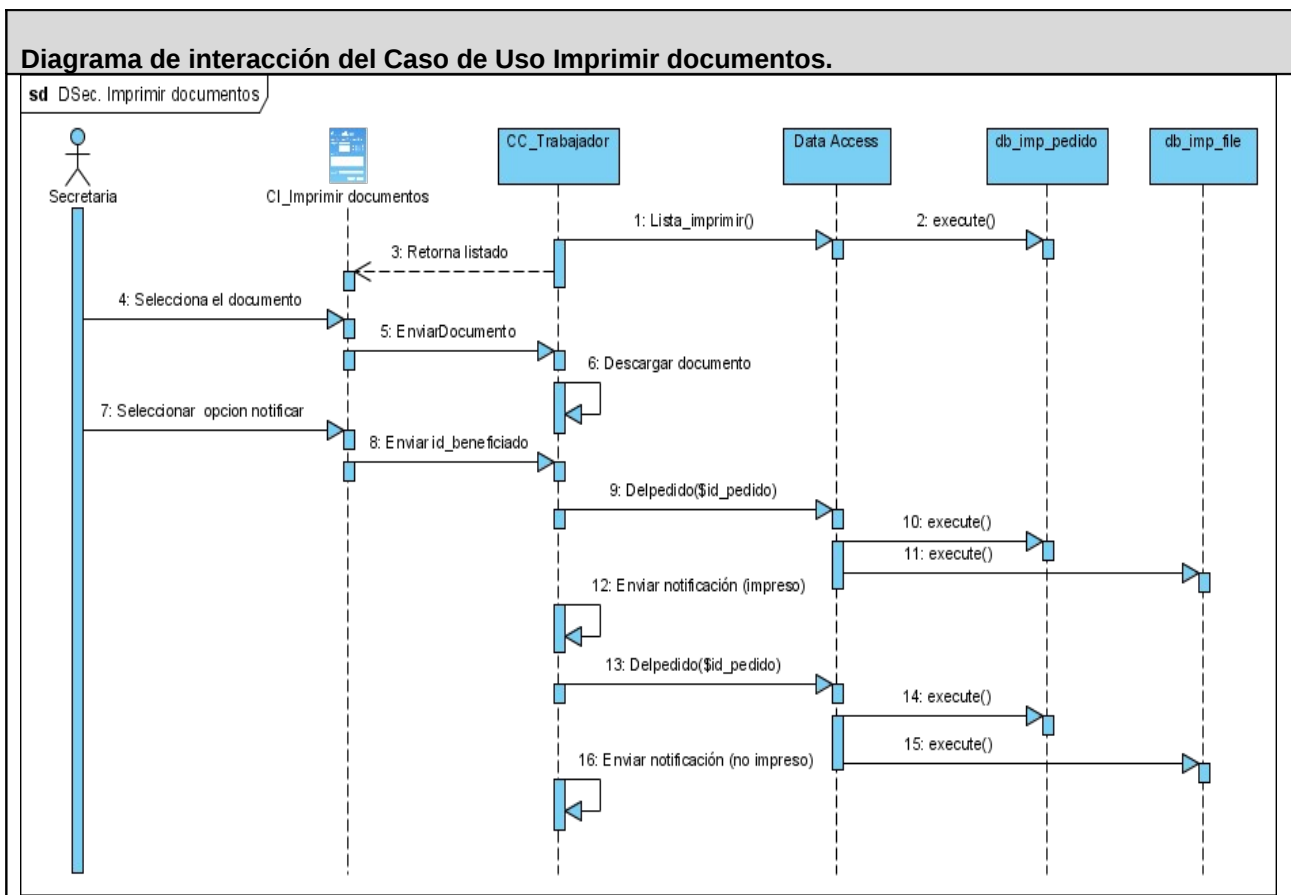
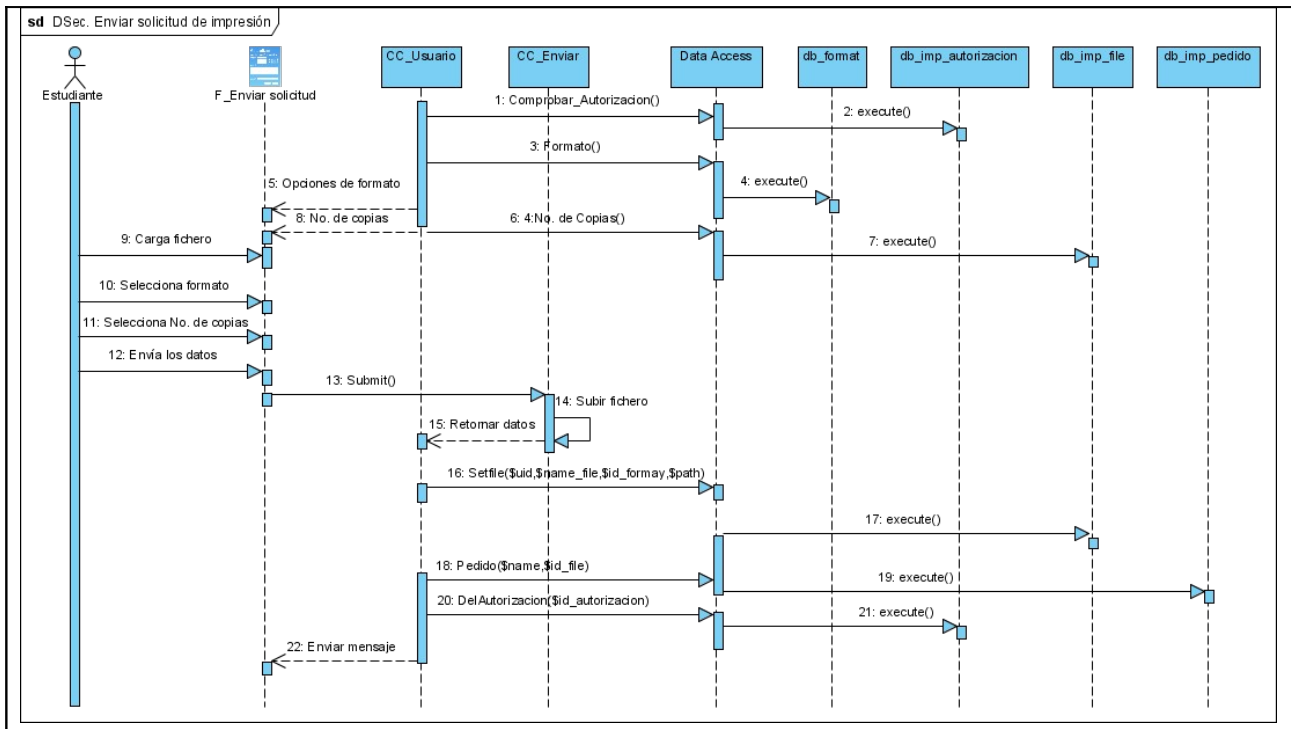


Diagrama de interacción del Caso de Uso Enviar solicitud de impresión.

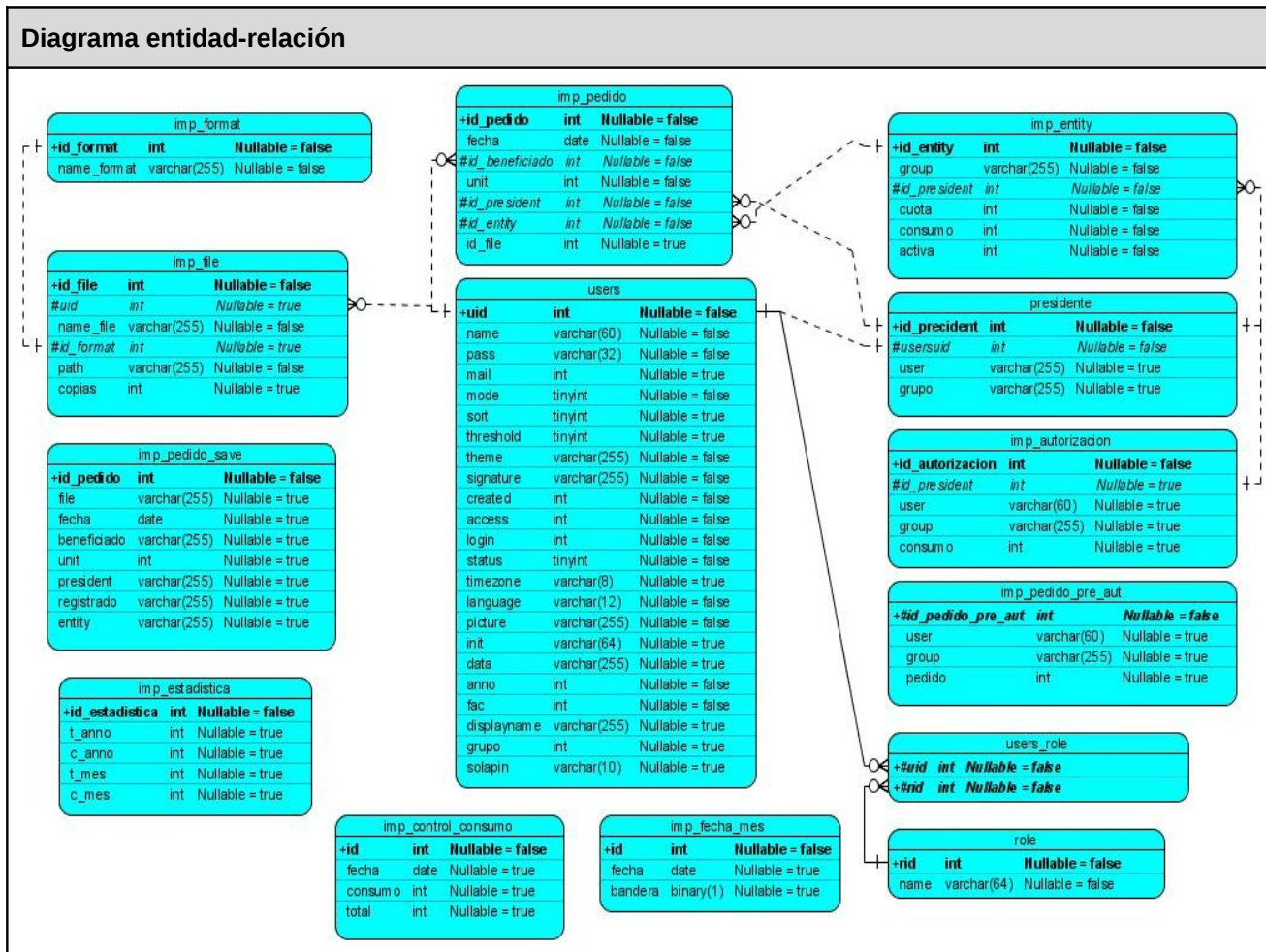


3.4 Diseño de la Base de Datos.

Uno de los procesos más importantes en la construcción de una aplicación que maneje base de

datos, es sin duda, el diseño de la misma. El modelo de datos del caso específico tiene una complejidad relativamente baja, debido a su desarrollo en Drupal. La mayor parte de las entidades generadas las maneja el CMS por lo que no se contempla en el modelo.

La base de datos para los procesos correspondientes al ciclo de vida para el módulo de impresión queda de la siguiente forma:



3.4.1 Descripción de las tablas principales.

En la descripción de las tablas principales se hace referencia al nombre y tipo de clase, una pequeña síntesis de la funcionalidad de las clases y las variables con las que cuenta cada una, especificando su tipo. Para ver una breve descripción de las tablas principales ver el Anexo 5.

3.5 Arquitectura de software

La arquitectura de software es la organización fundamental de un sistema encarnada en sus componentes, las relaciones entre ellos y su ambiente, y los principios que orientan su diseño y

evolución.

Generalmente en los sistemas se emplean un conjunto de estilos arquitectónicos tales como: **Arquitectura Orientada a Servicios** (en inglés Service-Oriented Architecture o SOA), es un concepto de arquitectura de software que define la utilización de servicios para dar soporte a los requerimientos de software del usuario. SOA proporciona una metodología y un marco de trabajo para documentar las capacidades de negocio y puede dar soporte a las actividades de integración y consolidación. En un ambiente SOA, los nodos de la red hacen disponibles sus recursos a otros participantes en la red como servicios independientes a los que tienen acceso de un modo estandarizado. La mayoría de las definiciones de SOA identifican la utilización de Servicios Web (empleando SOAP y WSDL) en su implementación, no obstante se puede implementar una SOA utilizando cualquier tecnología basada en servicios, por ejemplo, REST o XML-RPC.

Arquitectura orientada a objetos: Los componentes del sistema encapsulan datos y operaciones que deben utilizarse para manipular dichos datos. La comunicación y coordinación entre componentes se realiza mediante envío de mensajes. En esencia es un sistema donde se enfatiza el empaquetamiento entre datos y operaciones que permiten manipular y acceder a dichos datos

Arquitectura en capas: Se definen un conjunto de niveles o capas, cada nivel interno que se atraviesa se aproxima más al nivel del conjunto de instrucciones máquina.

Sistemas en capas puros.- cada capa sólo puede comunicarse con la vecinas. Esta solución aunque puede ser menos eficiente en algunos casos, facilita la portabilidad de los diseños.

Modelo Vista Controlador (MVC) es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.

3.5.1 Patrón MVC

Para realizar el módulo del servicio de impresión se utiliza el Modelo Vista Controlador debido a que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos, donde la vista es la página HTML y el código que provee de datos dinámicos a la página; el modelo son las clases que acceden a los datos en el [Sistema de Gestión de Base de Datos](#) y el controlador representa la [Lógica de negocio](#). Además, se representa la base de datos como fuente de datos, no siendo esta una capa debido a que en el sistema no se tienen procedimientos almacenados, por lo que no hay procesamiento lógico.

3.5.2 Descripción del patrón MVC

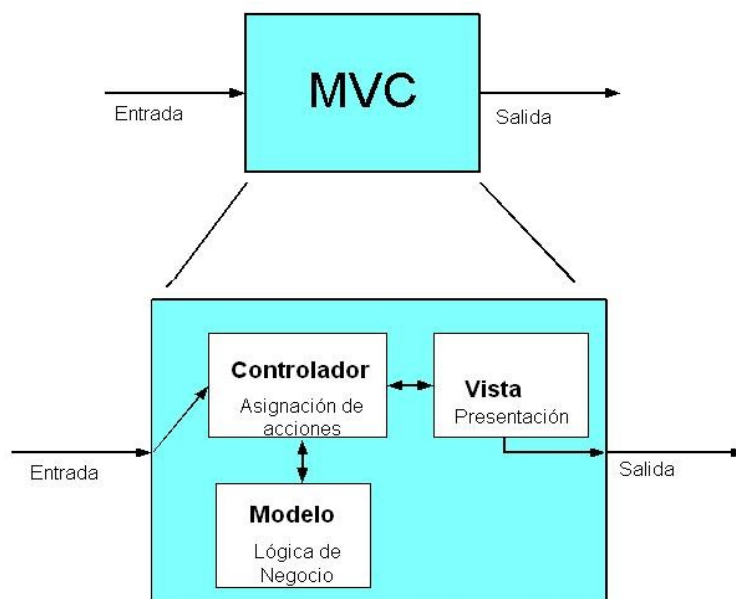
- **Modelo:** componente encargado del acceso a datos. Modela los datos y el comportamiento detrás de los procesos de negocio. A través de este componente se

realizan las consultas a la base de datos, se ejecutan los cálculos de los procesos de negocio y se procesan las diferentes órdenes. Se encapsulan los datos y el comportamiento que son independientes de la presentación. Para el acceso a los datos se presenta una interfaz denominada Data Access que implementa el patrón Factoría.

- **Vista:** en la vista se despliega la información de acuerdo al tipo de cliente, así como los resultados de la lógica de negocios (modelo). Una vista puede ser una página web o una parte de una página.
- **Controlador:** sirve como una conexión lógica entre la interacción de usuario y los servicios de negocio disponibles. En otras palabras, el controlador actúa como intermediario entre el modelo, la vista y cualquier otro recurso necesario para generar una página.

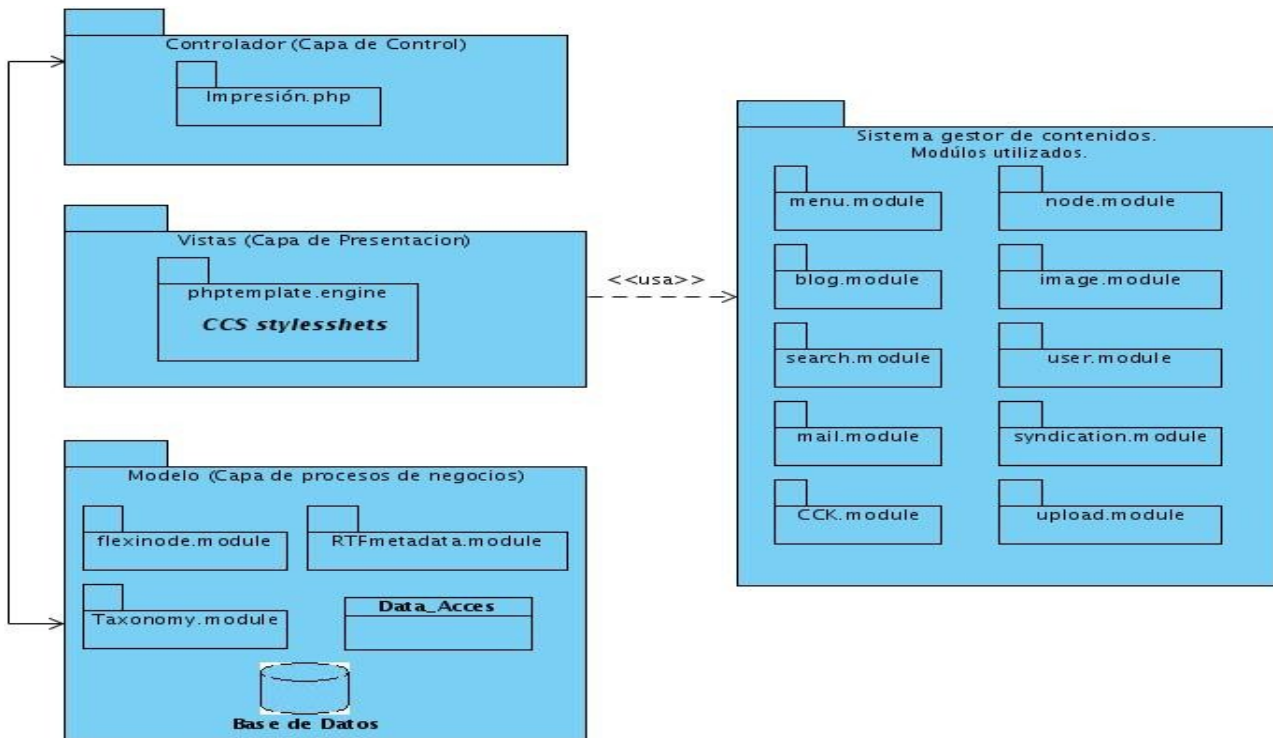
3.5.3 Procesamiento del patrón MVC.

El procesamiento se lleva a cabo entre sus tres componentes de la siguiente manera: El controlador recibe una orden y decide quién la lleva a cabo en el modelo. Una vez que el modelo (la lógica de negocio) termina sus operaciones devuelve el flujo al controlador y este envía el resultado a la capa de presentación. Este proceso se puede ver más claro en la siguiente figura:



El CMS Drupal, el cual será utilizado para el desarrollo del sistema, utiliza el modelo-vista-controlador como estilo de arquitectura, con la única diferencia que implementa este patrón arquitectónico de manera diferente al sistema propuesto. Esta diferencia está enmarcada en que

la lógica de negocio propuesta por el sistema a desarrollar es bastante compleja e impide programar para este gestor de contenido. Por lo que es más factible desarrollar la lógica del negocio haciendo uso de algunos módulos que posee Drupal. A continuación se especifica la relación que existe entre el sistema a desarrollar y el CMS a utilizar:



3.6 Tratamiento de errores.

Con el desarrollo de un software se trata de que sea funcional para los usuarios, si esto no se cumple, se dice que el software contiene errores. Estos errores son los detectables por el usuario, lo que exige al desarrollador del software trabajar exhaustivamente la calidad externa de la aplicación, lo que guarda una relación directa con la calidad interna del mismo, es decir con su estructura y codificación.

Drupal brinda la posibilidad de tratar los errores del sistema de manera flexible y dinámica. Cuando este se encuentra en presencia de una excepción el cliente es re-direccionado a una página de error. Los formularios existentes en cada una de las páginas son los encargados de validar los datos que manejan; evitando así errores de concepto. Además se utilizan mensajes de confirmación en el caso de acciones irreversibles como eliminar.

Se destaca además que los mensajes de error se imprimen en la misma página; de esta forma es más fácil corregirlos.

Con el desarrollo del presente capítulo se expusieron elementos que modelaron la construcción del sistema; teniendo en cuenta las clases de análisis y diseño. Durante esta etapa de análisis y diseño se realizó un análisis del sistema ya en términos de solución, quedando definida la

estructura del sistema en el análisis, donde se suministran una comprensión detallada de los requisitos con los cuales tiene que cumplir el sistema. Mediante el diseño se comprendió la lógica general del sistema. Se realizó el diagrama de clases persistentes de la base de datos, el cual contiene la información física que se utilizó para construir la aplicación, el tipo de arquitectura utilizado para el desarrollo del sistema y por último se expuso el tema relacionado al tratamiento de errores.

CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA

La implementación comienza con el resultado del diseño y se implementa el sistema en términos de componentes, es decir, ficheros de código fuente, scripts, ficheros de código binario, ejecutables y similares.

En el flujo de trabajo de diseño se propone crear un plano del modelo de implementación, por lo que sus últimas actividades están vinculadas a la creación del modelo de despliegue. El flujo de trabajo de implementación describe cómo los elementos del modelo del diseño se implementan en términos de componentes y cómo estos se organizan de acuerdo a los nodos específicos en el modelo de despliegue.

En este capítulo se presentan los diagramas de despliegue y componentes, que son artefactos generados durante dicho flujo de trabajo, conformando lo que se conoce como un modelo de implementación al describir los componentes a construir y su organización y dependencia entre nodos físicos en los que funcionará a aplicación.

Los principales propósitos de la implementación son:

- Definir la organización del código en términos de Subsistemas de Implementación organizados en capas.
- Implementar los elementos de diseño en términos de elementos de implementación (ficheros fuentes, binarios, ejecutables y otros).
- Probar los componentes desarrollados independientemente como unidades.
- Integrar los resultados producidos por desarrolladores independientes o equipos en un sistema ejecutable.

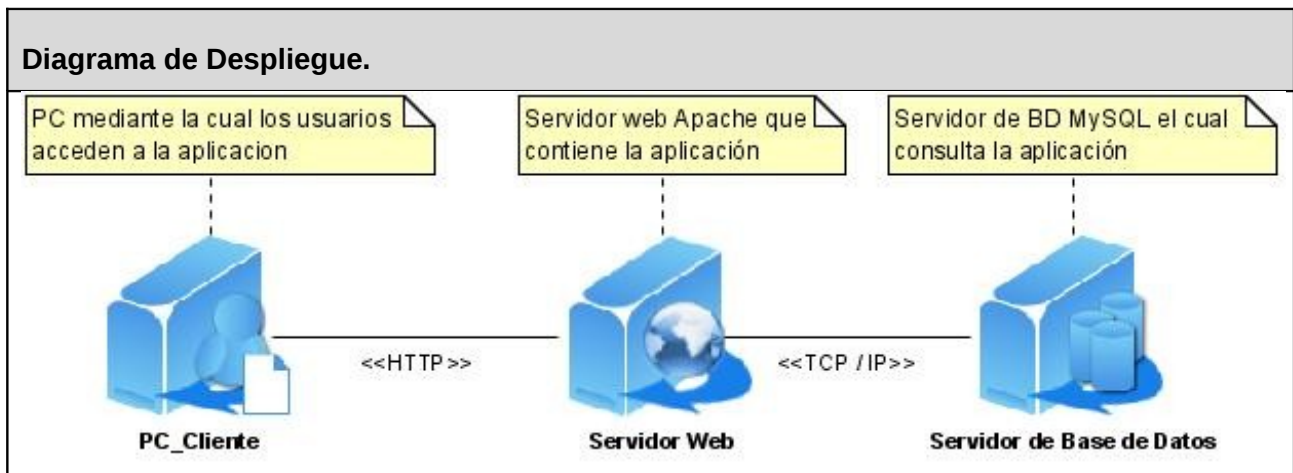
Finalmente, en este capítulo se definen algunas pruebas de software cuyo principal objetivo es asegurar la calidad del software desarrollado y como revisión final del cumplimiento de las especificaciones del diseño y de la codificación.

4.1 Diagrama de despliegue

Un diagrama de Despliegue muestra cómo y dónde se desplegará el sistema. Las máquinas físicas y los procesadores se representan como nodos, y la construcción interna puede ser representada por nodos o artefactos embebidos. Como los artefactos se ubican en los nodos para modelar el despliegue del sistema, la ubicación es guiada por el uso de las especificaciones de despliegue.

El diagrama de despliegue muestra las relaciones físicas entre los componentes *hardware* y

software en el sistema final, es decir, la configuración de los elementos de procesamiento en tiempo de ejecución y los componentes *software* (procesos y objetos que se ejecutan en ellos).



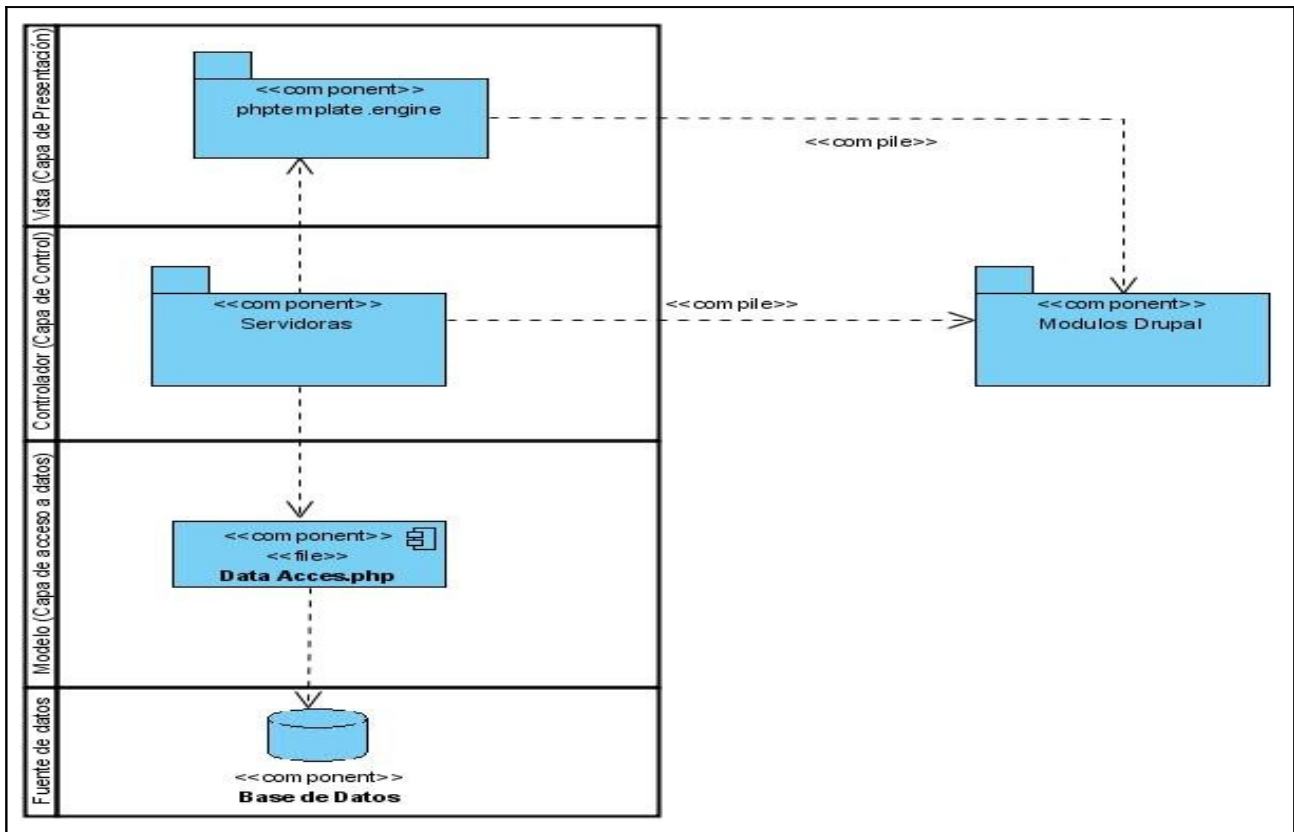
4.2 Diagrama de componentes

El diagrama de componentes muestra las organizaciones y dependencias lógicas entre componentes software, sean éstos componentes de código fuente, binarios o ejecutables. Desde el punto de vista del diagrama de componentes se tienen en consideración los requisitos relacionados con la facilidad de desarrollo, la gestión del software, la reutilización, y las restricciones impuestas por los lenguajes de programación y las herramientas utilizadas en el desarrollo. Los elementos de modelado dentro de un diagrama de componentes serán componentes y paquetes. En cuanto a los componentes, sólo aparecen tipos de componentes, ya que las instancias específicas de cada tipo se encuentran en el diagrama de despliegue.

Un diagrama de componentes se representa como un grafo de componentes software unidos por medio de relaciones de dependencia (generalmente de compilación). Puede mostrar también que un componente software contiene una interfaz, es decir, la soporta.

El diagrama mostrado a continuación se realiza con el objetivo de modelar la distribución lógica de los componentes por capas, mostrando las capas de la arquitectura definida y la fuente de datos, donde se almacenan los mismos.





4.3 Etapa de pruebas

El principal objetivo de la etapa de pruebas es evaluar la calidad del producto que se está desarrollando a través de las diferentes fases por las cuales este pasa, mediante la aplicación de pruebas concretas para validar que las suposiciones hechas en el diseño y los requerimientos se estén cumpliendo satisfactoriamente, esto quiere decir que se verifica que el producto funcione como se diseñó y que los requerimientos son satisfechos cabalmente. La realización de pruebas debe estar presente en todo el ciclo de vida del desarrollo del sistema para ir refinándolo constantemente y no al final del mismo.

Los principales aspectos a ser evaluados en un producto software son la Funcionalidad (hace lo que debe), la Fiabilidad (resistente a fallos), y el Rendimiento (lleva a cabo su trabajo de manera efectiva). Las pruebas pueden hacerse a diferentes niveles dependiendo del objetivo de los mismos, entre algunos tenemos: Pruebas de unidad (se prueban las unidades mínimas por separado, y normalmente se hace durante la implementación misma), de integración (varias unidades juntas), de sistema (sobre la aplicación o sistema completo) y de aceptación (realizado sobre el sistema global por los usuarios o terceros).

Entre los casos de pruebas se puede distinguir dos tipos comúnmente utilizados: las llamadas pruebas de “caja negra” y las de “caja blanca”.

Las pruebas de caja negra se llevan a cabo sobre la interfaz del software. Su principal objetivo es demostrar que las funciones del software son operativas, que las entradas se aceptan de forma

adecuada y se produce un resultado correcto, y que la integridad de la información externa se mantiene (no se ve el código).

En las pruebas de caja blanca se comprueban los caminos lógicos del software. En estas se puede examinar el estado del programa en varios puntos para determinar si el estado real coincide con el esperado (sobre el código).

Para comprobar la calidad del producto realizado, se propone realizar una prueba de caja negra a cada caso de uso para verificar que los casos de uso cumplen con las precondiciones y postcondiciones especificadas para cada uno. En la realización de estas pruebas, se tendrá en cuenta la técnica de la Partición de Equivalencia, la cual permite examinar los valores válidos e inválidos de las entradas existentes en el software, además, descubre de forma inmediata una clase de errores que, de otro modo, requerirían la ejecución de muchos casos antes de detectar el error genérico.

4.3.1 Descripción de los casos de prueba

Los casos de prueba se realizan con el objetivo de comprobar que la aplicación cumple con las funcionalidades previstas, en las siguientes tablas se describen las entradas de datos para cada caso de uso, así como las condiciones que debe cumplir cada uno para alcanzar los resultados esperados.

CP1: Caso de Uso Configurar sistema

CP1.1 Adicionar jefe de entidad

Entrada	Resultados esperados	Condiciones
El administrador del sistema deja todos los campos del formulario vacíos.	El sistema muestra un mensaje de error diciendo que debe llenar todos los campos	El administrador del sistema debe llenar todos los campos del formulario.
El administrador escribe incorrectamente el nombre de usuario del jefe de entidad.	El sistema muestra un mensaje de error diciendo que ese usuario no existe en la base de datos.	El administrador del sistema debe escribir un nombre de usuario válido de un estudiante.
El administrador llena todos los campos correctamente.	El sistema adiciona al usuario seleccionado a la tabla de presidentes de brigada.	El usuario es un estudiante de la UCI.
Escribe caracteres no válidos en cualquiera de los campos.	El sistema muestra un mensaje de error diciendo que debe completar	El administrador debe llenar los campos del formulario

	los campos correctamente.	correctamente.
--	---------------------------	----------------

CP1.2 Modificar jefe de entidad

Entrada	Resultados esperados	Condiciones
El administrador del sistema deja todos los campos del formulario vacíos y selecciona la opción modificar.	El sistema muestra un mensaje de error diciendo que debe llenar todos los campos	El administrador del sistema debe llenar todos los campos del formulario.
El administrador escribe incorrectamente el nombre de usuario del jefe de entidad.	El sistema muestra un mensaje de error diciendo que ese usuario no existe en la base de datos.	El administrador del sistema debe escribir un nombre de usuario válido de un estudiante.
El administrador llena todos los campos correctamente.	El sistema modifica el jefe de entidad en la tabla correspondiente.	El usuario es un estudiante de la UCI.
Escribe caracteres no válidos en cualquiera de los campos.	El sistema muestra un mensaje de error diciendo que debe completar los campos correctamente.	El administrador debe llenar los campos del formulario correctamente.
El administrador decide cancelar la modificación del jefe de entidad.	El sistema vacía los campos que habían sido completados.	El administrador no desea modificar ese jefe de entidad.

CP1.3 Asignar hojas

Entrada	Resultados esperados	Condiciones
El administrador del sistema no selecciona una cantidad determinada de hojas para establecer la cuota.	El sistema muestra un mensaje de error diciendo que debe seleccionar un número de hojas para realizar la asignación.	El administrador del sistema debe seleccionar un número de hojas para establecer la cuota.
El administrador del sistema selecciona una cantidad determinada de hojas para establecer la cuota.	El sistema asigna dicha cantidad de hojas a la entidad seleccionada.	El administrador del sistema asigna un número válido de hojas.

CP2: Caso de Uso Realizar solicitud de impresión

Entrada	Resultados esperados	Condiciones
El estudiante no selecciona en el formulario una cantidad de hojas específicas para realizar la solicitud.	El sistema muestra un mensaje de error diciendo que debe seleccionar un número de hojas para realizar la solicitud.	El estudiante debe seleccionar un número de hojas para realizar la solicitud.
El estudiante selecciona un número específico de hojas para realizar la solicitud.	El sistema elimina dicha cantidad de hojas de las hojas disponibles del grupo y añade la solicitud a la lista de solicitudes de impresión por autorizar al jefe de entidad.	Existen hojas disponibles para realizar la solicitud.

CP3: Caso de Uso Enviar solicitud de impresión

Entrada	Resultados esperados	Condiciones
El estudiante no selecciona ningún documento para imprimir y selecciona la opción enviar.	El sistema muestra un mensaje de error diciendo que debe seleccionar un documento para imprimir.	El estudiante debe seleccionar el documento que desea imprimir.
Después de seleccionar el documento no se selecciona el tipo de formato con el que se va a imprimir y se escoge la opción enviar.	El sistema muestra un mensaje de error diciendo que debe seleccionar el formato de impresión del documento.	El estudiante debe haber seleccionado un formato para imprimir el documento.
Se añade el documento y se selecciona el formato de impresión, el número de copias y escoge la opción enviar.	El sistema añade el documento a la lista de documentos por imprimir y muestra un mensaje notificando que su documento ha sido enviado con éxito.	
El estudiante decide cancelar el envío de solicitud de impresión.	El sistema muestra un mensaje notificando que su documento ha sido enviado con éxito.	Haber solicitado la cancelación de la solicitud de impresión.

CP4: Caso de Uso Imprimir documentos

Entrada	Resultados esperados	Condiciones
La secretaria envía la notificación de que el documento se ha impreso.	El sistema envía un correo notificando al estudiante que solicitó el servicio de impresión que ya su documento ha sido impreso y añade el documento a la lista de documentos impresos.	La secretaria debe haber impreso el documento.
La secretaria elimina el documento del listado de documentos por imprimir.	El sistema elimina el documento de la lista de documentos por imprimir y envía un correo al estudiante notificando que su documento no ha sido impreso.	El documento no cumple con los requisitos establecidos por la oficina de impresión.

CP5: Caso de Uso Buscar información

Entrada	Resultados esperados	Condiciones
La secretaria deja los dos campos del formulario vacíos.	El sistema muestra un mensaje de error diciendo que debe llenar al menos un campo.	La secretaria debe poner el grupo para realizar la búsqueda.
La secretaria escribe el número del grupo.	El sistema muestra la cantidad de hojas consumidas y disponibles de dicho grupo.	El grupo seleccionado debe estar registrado en la base de datos.
La secretaria escribe caracteres no válidos en alguno de los campos del formulario.	El sistema muestra un mensaje de error diciendo que debe llenar los campos correctamente.	La secretaria debe poner un número de grupo válido para realizar la búsqueda.

En este capítulo se presentó la distribución física por nodos de la aplicación en el diagrama de despliegue, y mediante el diagrama de componentes se representaron las dependencias entre los mismos. Se realizó un estudio de los diferentes tipos de pruebas que se pueden realizar a la aplicación para comprobar su funcionamiento, realizándose la descripción de los casos de prueba de integración para los casos de uso más significativos, que aunque no aseguran la no existencia de fallos, si dan una alta confiabilidad, y un nivel de calidad de la aplicación lista para pasar a una etapa de pruebas de aceptación por parte del cliente.

CONCLUSIONES GENERALES

En el presente trabajo se propuso la realización de una aplicación para facilitar el proceso de impresión de la Dirección de Informatización de la UCI, logrando que el mismo se desarrolle de manera más eficiente. Con este:

- Se logra una completa automatización del servicio de impresión.
- Permite controlar de manera más rigurosa el consumo de hojas.
- Se agiliza el proceso de impresión tanto para los estudiantes que solicitan el servicio como para los trabajadores del local.
- Provee a los trabajadores del local de un mejor control sobre las autorizaciones realizadas para imprimir, asegurando de esta manera su veracidad.
- Facilita en gran medida el trabajo en la oficina de impresión.

RECOMENDACIONES

- Continuar investigando acerca de las tecnologías de desarrollo y otros sistemas similares con el objetivo de garantizar mejoras del sistema en futuras versiones del mismo.
- Que el presente trabajo se siga desarrollando para implementar nuevas funcionalidades de acuerdo al surgimiento de nuevas necesidades y mejorar su calidad y robustez.
- Poner en funcionamiento la aplicación para que pueda ser utilizada por los estudiantes de la universidad.

REFERENCIAS BIBLIGRÁFICAS

- [1] Travesía. *El portal en internet de las bibliotecas públicas españolas*. [En línea] [Citado el: 27 de noviembre de 2008.] Disponible en: <http://travesia.mcu.es/documentos/bp-def.asp>.
- [2] Universidad de Alicante. [En línea] [Citado el: 4 de febrero de 2009.] Disponible en: http://www.ua.es/es/bibliotecas/servicios/otros_servicios/impr_digital/imp_dig.html.
- [3] IDICT. *Instituto de Información Científica y Tecnológica*. [En línea] [Citado el: 4 de febrero de 2009.] Disponible en: <http://www.idict.cu/servicios/poligraficos>.
- [4] REYES HERNÁNDEZ, Kenia. *Diseño del Módulo OPAC para la Biblioteca Nacional José Martí*. Trabajo de Diploma para optar por el título de Ingeniería en Ciencias Informáticas. Ciudad de la Habana, 2007.
- [5] desarrollo web.com. *Arquitectura cliente-servidor*. [En línea] [Citado el: 6 de febrero de 2009.] Disponible en: <http://www.desarrolloweb.com/articulos/arquitectura-cliente-servidor.html>
- [6] Kioskea.net. *Entorno cliente/servidor*. [En línea][Citado el: 6 de febrero de 2009.] Disponible en: <http://es.kioskea.net/contents/cs/csintro.php3>
- [7] XperimentoS. *Experiencias Profesionales... Gestión de contenidos*. [En línea] [Citado el: 6 de febrero de 2009.] Disponible en: <http://www.xperimentos.com/2007/03/02/gestion-de-contenidos/>
- [8] BananaTools.com. *Ventajas de utilizar Drupal*. [En línea] [Citado el: 6 de febrero de 2009.] Disponible en: <http://www.banatatools.com/drupal/conceptos-basicos/18/caracteristicas-generales-de-drupal>
- [9] Universidad Nacional de Colombia. *Lenguaje PHP*. [En línea] [Citado el: 6 de febrero de 2009.] Disponible en: <http://www.virtual.unal.edu.co/cursos/sedes/manizales/4060029/lecciones/cap11-2.html>
- [10] TuFuncion. [En línea] Los mejores IDEs para PHP. [Citado el: 6 de febrero de 2009.] <http://www.tufuncion.com/ide-php>
- [11] Firebird en español. [En línea] [Citado el: 7 de febrero de 2009.] <http://www.firebird.com.mx/modules/news/>
- [12] esepestudio. *¿Qué es MySQL?* [En línea] [Citado el: 7 de febrero de 2009.] Disponible en: <http://www.esepestudio.com/articulo/desarrollo-web/bases-de-datos-mysql/Que-es-MySQL.htm>
- [13] GÁMEZ LAMBERT, Aliuska (Alfonso, 2009). *Análisis y Diseño del Portal del Graduado*. Trabajo de Diploma para optar por el título de Ingeniería en Ciencias Informáticas. Ciudad de la Habana, 2007.
- [14] Gobierno Bolivariano de Venezuela. *Me Rinde*. [En línea]. Trabajador del Negocio. [Citado el: 25 de febrero de 2009.] Disponible en: http://merinde.rinde.gob.ve/index.php?option=com_content&task=view&id=128&Itemid=294

BIBLIOGRAFÍA

- Universitat de València. [En línea] [Citado el: 4 de febrero de 2009.] Disponible en: http://biblioteca.uv.es/castellano/bibliotecas/de_campus/b_ciencias/b_ciencias7.php.
- USB. *Universidad Simón Bolívar*. [En línea] [Citado el: 4 de febrero de 2009.] Disponible en: <http://www.bib.usb.ve/fotocopiado.htm>.
- Tecnológico de Monterrey. [En línea] [Citado el: 4 de febrero de 2009.] Disponible en: <http://biblioteca.mty.itesm.mx/node/649>.
- Biblioteca Nacional Ernesto J. Castellero R. [En línea] [Citado el: 4 de febrero de 2009.] Disponible en: <http://www.binal.ac.pa/visitenos/servicios.htm>.
- GNU Operating System. *La Definición de Software Libre* [En línea] [Citado el: 4 de febrero de 2009.] Disponible en: <http://www.gnu.org/philosophy/free-sw.es.html>
- Luanet. *Soluciones para su negocio*. [En línea] Gestión de contenidos. [Citado el: 6 de febrero de 2009.] Disponible en: http://www.luanet.com/gestion_de_contenidos.php
- Drupal Web [En línea] [Citado el: 6 de febrero de 2009.] Disponible en: <http://www.drupalweb.com/>
- investic. *Drupal CMS*. [En línea] [Citado el: 6 de febrero de 2009.] Disponible en: http://www.investig.net/servicios/soluciones_web/drupal
- informatizate. *Metodologías de desarrollo de software*. [En línea] [Citado el: 6 de febrero de 2009.] Disponible en: http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html
- Gaviota. *El lenguaje HTML*. [En línea] [Citado el: 6 de febrero de 2009.] Disponible en: <http://www.juntadeandalucia.es/averroes/iesgaviota/informatica/html.html>
- Maestros de la web. *¿Qué es Javascript?* [En línea] [Citado el: 6 de febrero de 2009.] Disponible en: <http://www.maestrosdelweb.com/editorial/%C2%BFque-es-javascript/>
- desarrolloweb.com. *Qué es y para qué sirve el SQL*. [En línea] [Citado el: 6 de febrero de 2009.] Disponible en: <http://www.desarrolloweb.com/articulos/262.php>
- desarrolloweb.com. *Zend Studio* [En línea] [Citado el: 7 de febrero de 2009.] Disponible en: <http://www.desarrolloweb.com/articulos/1178.php>
- TuFuncion. *Zend Studio* [En línea] [Citado el: 7 de febrero de 2009.] Disponible en: <http://www.tufuncion.com/zend-studio>
- Grupove. *Qué es PostgreSQL*. [En línea] [Citado el: 7 de febrero de 2009.] Disponible en: <http://grupove.org.ve/postgresql>
- Somos libres. *¿Qué es SQLite?* [En línea]. [Citado el: 7 de febrero de 2009.] Disponible en: <http://www.somoslibres.org/modules.php?name=News&file=article&sid=183>

masadelante.com. *¿Qué es un servidor web (Web Servers)? - Definición de servidor web* [En línea] [Citado el: 7 de febrero de 2009.] Disponible en: <http://www.masadelante.com/faq-servidor-web.htm>

Multimedia. *Servidores web*. [En línea] [Citado el: 10 de febrero de 2009.] Disponible en: <http://http://mery-multi.blogspot.com/2006/12/servidores-web.html>

IngenieroSoftware. *UML: Diagramas UML. ¿Qué es UML?* [En línea] [Citado el: 10 de febrero de 2009.] Disponible en: <http://www.ingenierosoftware.com/analisisydiseno/uml.php>

Whatis.com. *What is Rational Rose*. [En línea] [Citado el: 10 de febrero de 2009.] Disponible en: http://searchcio-midmarket.techtarget.com/sDefinition/0,,sid183_gci516025,00.html

Gobierno Bolivariano de Venezuela. *Me Rinde*. [En línea]. Modelo de Casos de Uso del Negocio. [Citado el: 25 de febrero de 2009.] Disponible en: http://merinde.rinde.gob.ve/index.php?option=com_content&task=view&id=90&Itemid=294

ANEXOS

Anexo 1. Descripción de Casos de Uso

1.1 Mostrar pedidos.

Caso de Uso:	Mostrar pedidos.	
Propósito:	Mostrar todos los pedidos de impresión realizados hasta la fecha de una entidad determinada.	
Actores:	Secretaria de oficina de impresión	
Resumen:	El caso de uso se inicia cuando la secretaria desea ver las impresiones que se han realizado hasta la fecha.	
Precondiciones:	Debe haberse impreso alguna solicitud.	
Referencias	R14	
Prioridad	Secundario	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1- La secretaria solicita al sistema ver la lista de pedidos realizados en la opción "Mostrar listado de pedidos realizados"	1.1 El sistema muestra la lista de pedidos realizados hasta la fecha.	
Flujos Alternos		
Acción del Actor	Respuesta del Sistema	
1- La secretaria solicita al sistema ver la lista de pedidos realizados en la opción "Mostrar listado de pedidos realizados"	1.2 El sistema verifica que no se ha impreso ningún pedido y muestra un mensaje "No existen documentos impresos". Termina el caso de uso.	
Postcondiciones	Se muestra el listado de documentos por imprimir	

1.2 Gestionar jefe de entidad.

Caso de Uso:	Adicionar jefe de entidad
Actores:	Administrador del sistema
Propósito:	Agregar o modificar jefes de entidad y mantener el sistema actualizado.
Resumen:	El caso de uso se inicia cuando el administrador accede al sistema; de esta manera podrá agregar o modificar un jefe de entidad determinado.
Precondiciones:	El administrador debe estar autenticado en el sistema

Referencias	R9
Prioridad	Critico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1- El administrador accede a la opción Gestión de jefes de brigada.	1.1 El sistema muestra los campos con las opciones que puede realizar, o sea escribir el usuario y el grupo.
2- El administrador llena los campos y envía los datos.	2.1 El sistema comprueba que los campos se hayan llenado correctamente. 2.2 El sistema actualiza el listado de los jefes de entidad.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
2- El administrador no llena los campos correspondientes correctamente.	2.3 El sistema muestra un mensaje "Entre todos los datos". Retorna a la acción 2.
Postcondiciones	Jefe de entidad agregado o modificado en el listado de jefes de entidad.

1.3 Asignar cuota

Caso de Uso:	Asignar cuota
Actores:	Administrador del sistema
Propósito:	Establecer el consumo mensual de hojas para las entidades.
Resumen:	El caso de uso se inicia cuando el administrador accede al sistema y de esta forma puede establecer una cuota para de hojas mensual para el consumo de las entidades.
Precondiciones:	El administrador debe estar autenticado en el sistema
Referencias	R10
Prioridad	Secundario
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1- El administrador accede a la opción Asignar hojas	1.1 El sistema le brinda la opción de que escriba el número de hojas que desee establecer como cuota.
2- El administrador escribe la cantidad de hojas y envía los datos.	2.1 El sistema actualiza la cuota de hojas establecida para el consumo de las entidades.

Flujos Alternos	
Acción del Actor	Respuesta del Sistema
2- El administrador escribe la cantidad de hojas incorrectamente y envía los datos.	2.2 El sistema muestra un mensaje "Debe entrar una cantidad de hojas válida". Retorna a la acción
Postcondiciones	Nueva cuota de hojas asignada al consumo de las entidades.

1.4 Consumo de hojas

Caso de Uso:	Consumo de hojas	
Actores:	Administrador del sistema	
Propósito:	Ver el consumo mensual, anual y total de hojas y el % que este representa.	
Resumen:	El caso de uso se inicia cuando el administrador accede al sistema y de esta forma puede ver estadísticas relacionadas con el consumo de hojas.	
Precondiciones:	El administrador debe estar autenticado en el sistema	
Referencias	R12	
Prioridad	Secundario	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1- El administrador accede a la opción Consumo de hojas.	1.1 El sistema le muestra el consumo de hojas mensual, anual y total de hojas.	
Flujos Alternos		
Acción del Actor	Respuesta del Sistema	
Postcondiciones	El administrador puede mantener el control de las hojas en la oficina de impresión.	

1.5 Listado de consumo

Caso de Uso:	Listado de consumo
Actores:	Administrador del sistema

Propósito:	Conocer el consumo de hojas por meses y el % del total que representa.
Resumen:	El caso de uso se inicia cuando el administrador accede al sistema y de esta forma puede ver un listado por meses con el consumo de cada mes y el % que representa con el total de hojas asignadas para ese mes.
Precondiciones:	El administrador debe estar autenticado en el sistema
Referencias	R13
Prioridad	Secundario

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1- El administrador accede a la opción Listado de consumo.	1.1 El sistema le muestra un listado con los meses, el total de hojas asignadas, la cantidad de hojas consumidas y el % que estas representan.

Flujos Alternos

Acción del Actor	Respuesta del Sistema

Postcondiciones	El administrador puede mantener el control de la cantidad de hojas que se imprimen en cada mes.
------------------------	---

1.6 Gestionar entidad

Caso de Uso:	Gestionar entidad
Actores:	Administrador del sistema
Propósito:	Gestionar las entidades del sistema y mantener el sistema actualizado.
Resumen:	El caso de uso se inicia cuando el administrador accede al sistema; de esta manera podrá asignar, modificar o crear una nueva entidad.
Precondiciones:	El administrador debe estar autenticado en el sistema
Referencias	R11
Prioridad	Critico

"Sección Crear entidad"

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1- El administrador accede a la opción Crear entidad.	1.1 El sistema muestra los campos con las opciones que puede realizar, o sea escribir la

	entidad, el usuario del jefe y si es estudiante o trabajador.
2- El administrador llena los campos y envía los datos.	2.1 El sistema comprueba que los campos se hayan llenado correctamente. 2.2 El sistema actualiza el listado de las entidades.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
2- El administrador no llena los campos correspondientes correctamente.	2.3 El sistema muestra un mensaje "Entre todos los datos". Retorna a la acción 2.
"Sección Modificar entidad"	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1- El administrador accede a la opción Gestión de entidad.	1.1 El sistema muestra un listado con las entidades existentes.
2- El administrador selecciona la entidad que desea modificar y selecciona la opción modificar.	2.1 El sistema comprueba que los campos se hayan llenado correctamente. 2.2 El sistema actualiza el listado de las entidades.
"Sección Eliminar entidad"	
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
2- El administrador no llena los campos correspondientes correctamente.	2.3 El sistema muestra un mensaje "Entre todos los datos". Retorna a la acción 2.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1- El administrador accede a la opción Gestión de entidad.	1.1 El sistema muestra un listado con las entidades existentes.
2- El administrador selecciona la entidad que desea modificar y selecciona la opción modificar.	2.1 El sistema comprueba que los campos se hayan llenado correctamente. 2.2 El sistema actualiza el listado de las entidades.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
2- El administrador no llena los campos correspondientes correctamente.	2.3 El sistema muestra un mensaje "Entre todos los datos". Retorna a la acción 2.

Postcondiciones	Entidad adicionada al sistema.
------------------------	--------------------------------

1.7 Buscar información

Caso de Uso:	Buscar información
Actores:	Secretaria de oficina de impresión.
Propósito:	Conocer la cantidad de hojas impresas y sin imprimir de cada grupo.
Resumen:	El caso de uso se inicia cuando la secretaria solicita al sistema saber la cantidad de hojas impresas y las que faltan por imprimir de cada grupo.
Precondiciones:	La secretaria debe haberse autenticado previamente.
Referencias	R10
Prioridad	Secundario

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1- La secretaria accede al sistema y solicita el estado de cada grupo en la opción "Buscar Información"	1.1 El sistema le pide que entre el número del grupo.
2- La secretaria entra el número del grupo	2.1 El sistema verifica que el número del grupo es correcto y muestra la cantidad de hojas impresas de ese grupo y la cantidad de hojas que le quedan disponibles para imprimir.

Flujos Alternos

Acción del Actor	Respuesta del Sistema
2- La secretaria entra el número del grupo.	2.2 El sistema verifica que el número del grupo es incorrecto y envía una notificación diciendo que no existe ese grupo y pide a la secretaria que vuelva a escribir el número del grupo.
Postcondiciones	El sistema muestra el estado del grupo seleccionado.

1.8 Mostrar información

Caso de Uso:	Mostrar información
Actores:	Jefe de entidad
Propósito:	Conocer la cantidad de hojas impresas y sin imprimir de su grupo.

Resumen:	El caso de uso se inicia cuando el jefe de entidad solicita al sistema saber la cantidad de hojas impresas y las que faltan por imprimir de su grupo.	
Precondiciones:	El jefe de brigada debe haberse autenticado previamente.	
Referencias	R6	
Prioridad	Secundario	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1- El jefe de entidad accede al sistema y selecciona la opción Mostrar listado.	1.1 El sistema le muestra un listado con la cantidad de hojas impresas y los nombres de los estudiantes que las imprimieron.	
Flujos Alternos		
Acción del Actor	Respuesta del Sistema	
Postcondiciones	El sistema muestra el estado de la entidad al que pertenece el jefe de entidad.	

Anexo 2. Diagramas de clases del análisis

Diagrama de clases del análisis del Caso de Uso Autenticar usuario

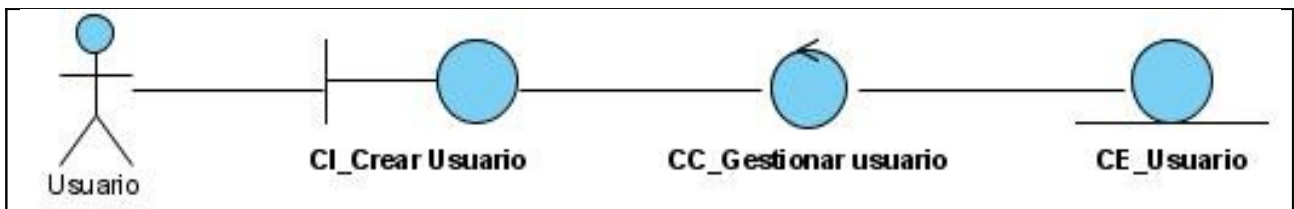


Diagrama de clases del análisis del Caso de Uso Gestionar rol

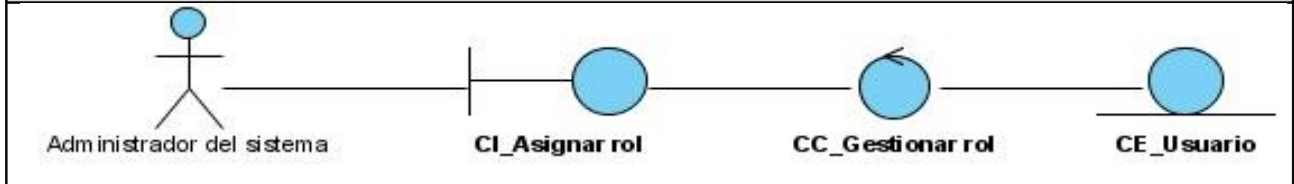


Diagrama de clases del análisis del Caso de Uso Gestionar usuario



Diagrama de clases del análisis del CU Gestionar jefe de entidad. Adicionar jefe de entidad

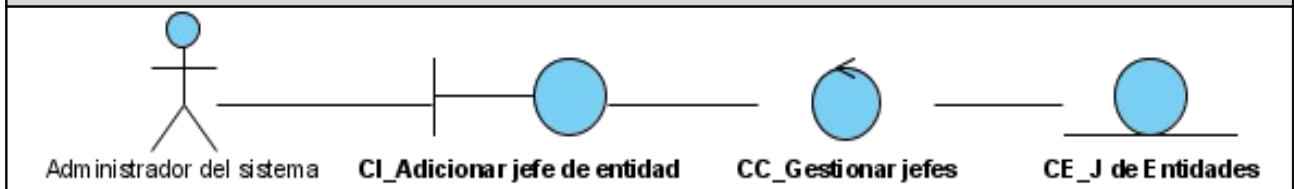


Diagrama de clases del análisis del CU Gestionar jefe de entidad. Modificar jefe de entidad

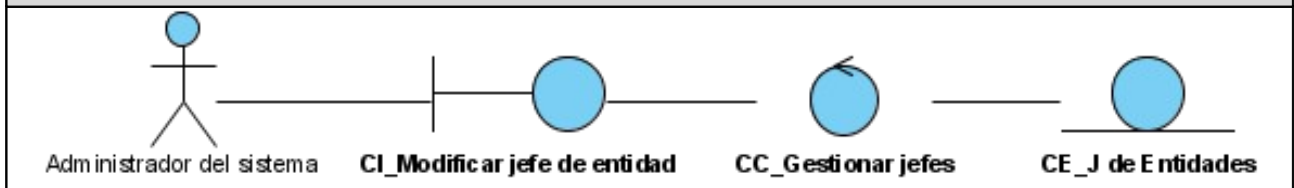
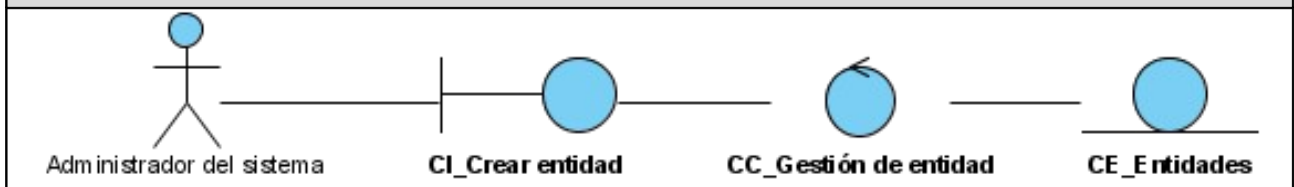
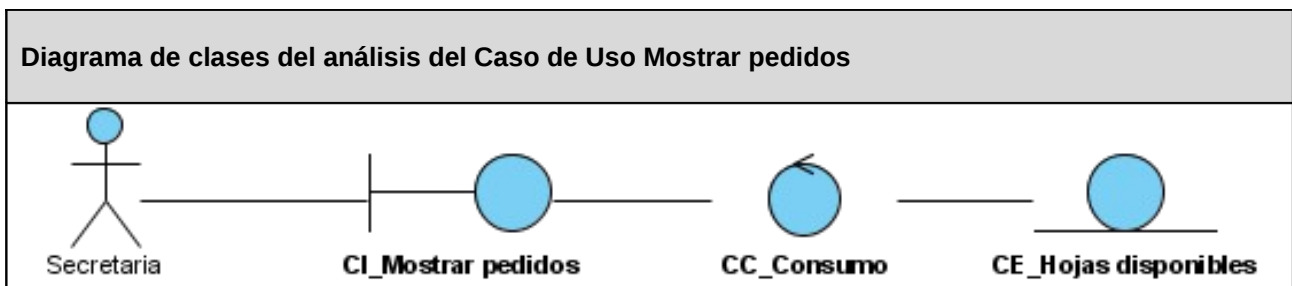
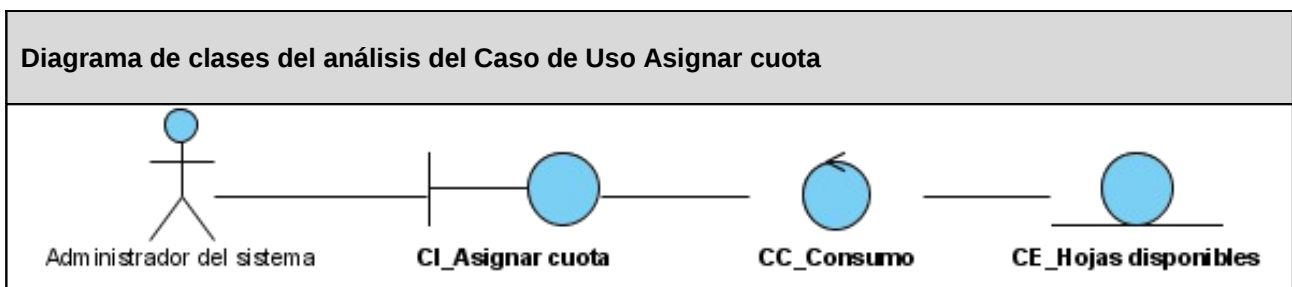
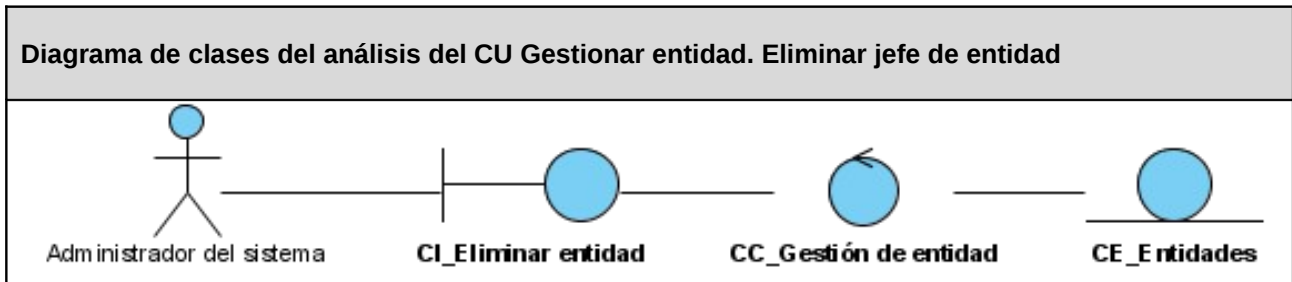
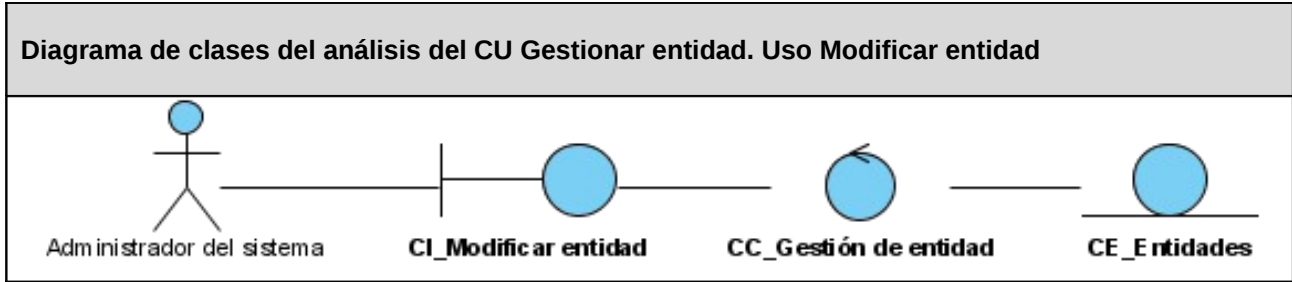
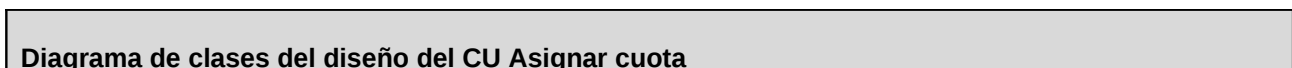


Diagrama de clases del análisis del CU Gestionar entidad. Crear entidad





Anexo 3. Diagramas de clases del diseño.



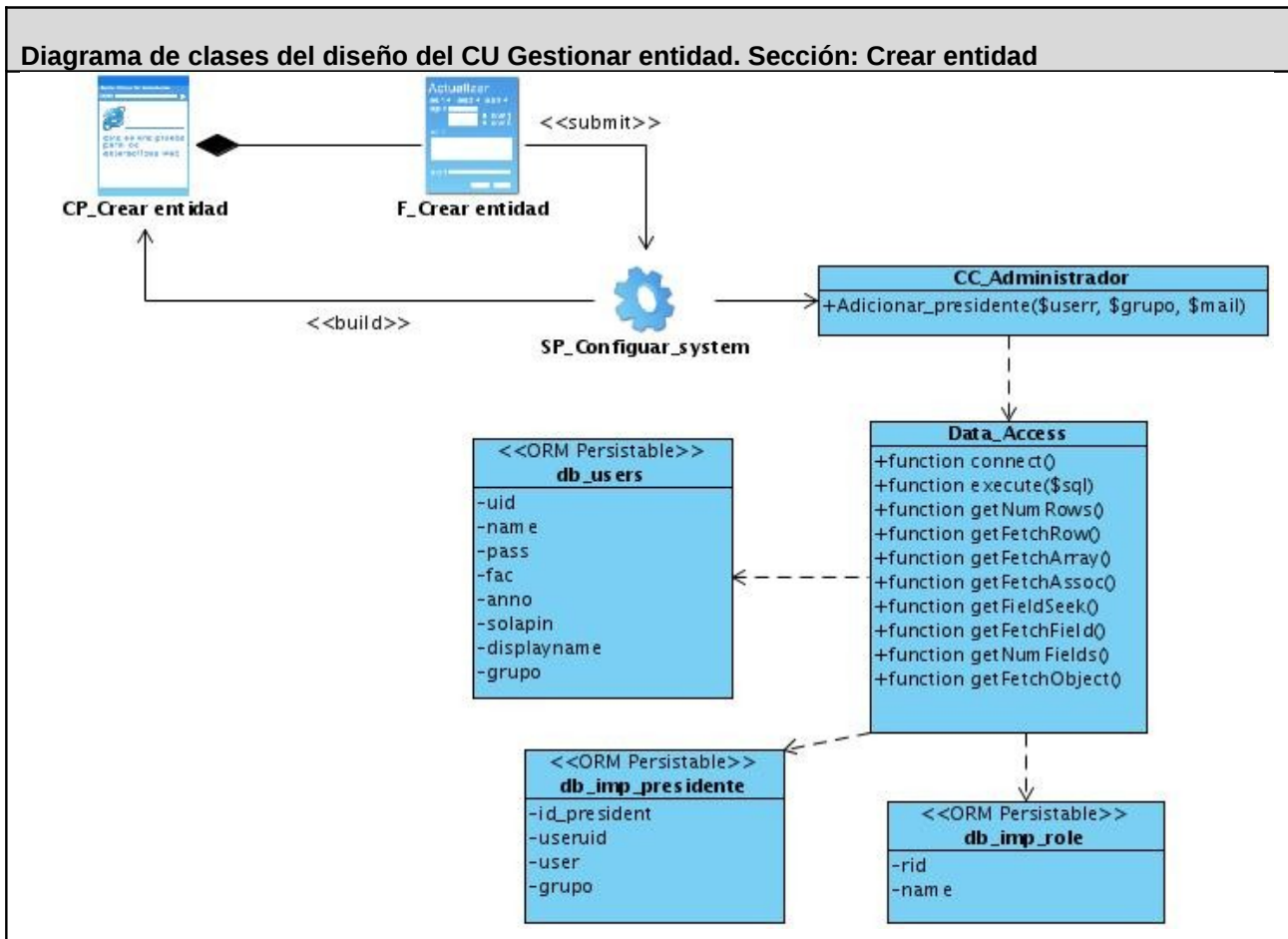
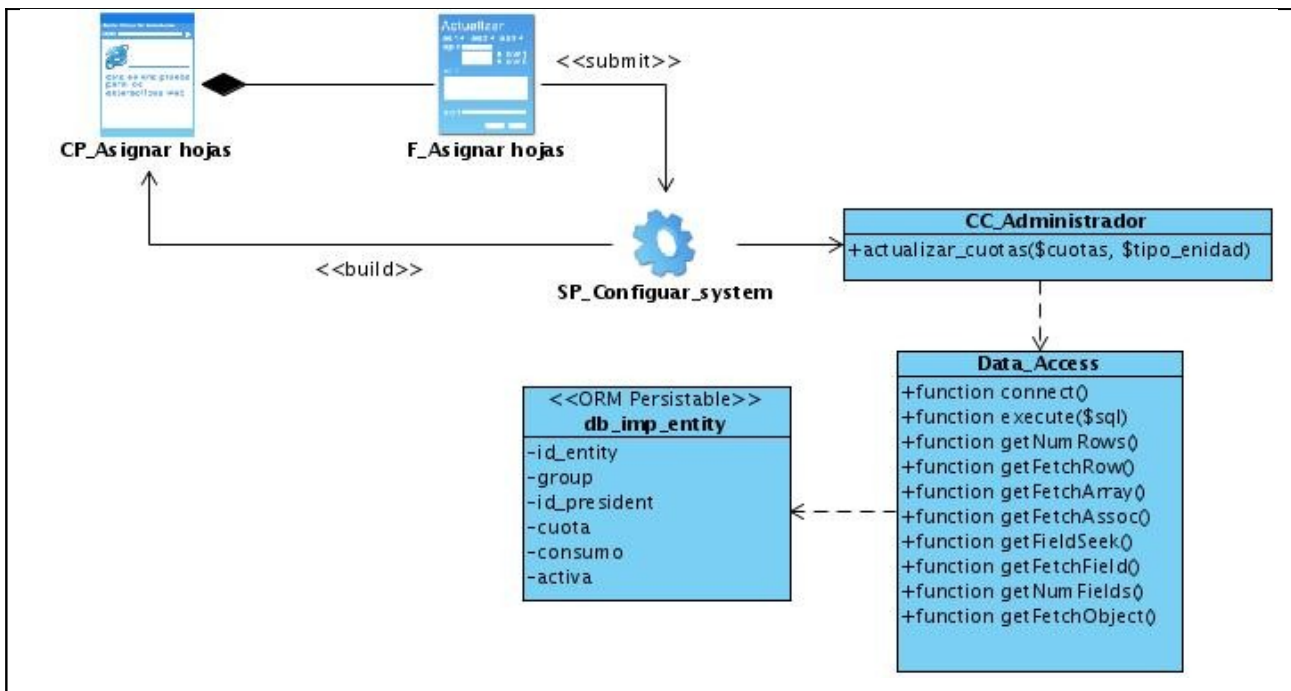


Diagrama de clases del diseño del CU Gestionar jefe de entidad. Sección: Adicionar jefe de entidad

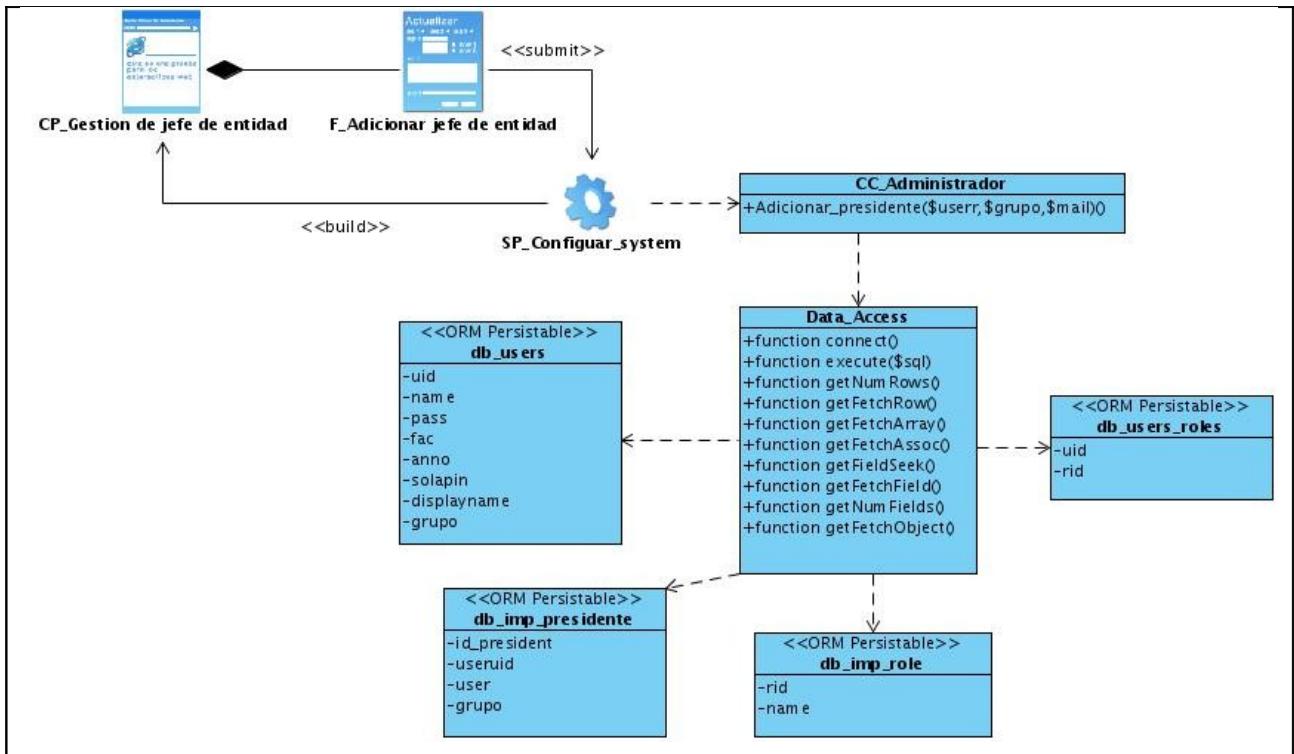


Diagrama de clases del diseño del CU Gestionar jefe de entidad. Sección: Modificar jefe de entidad

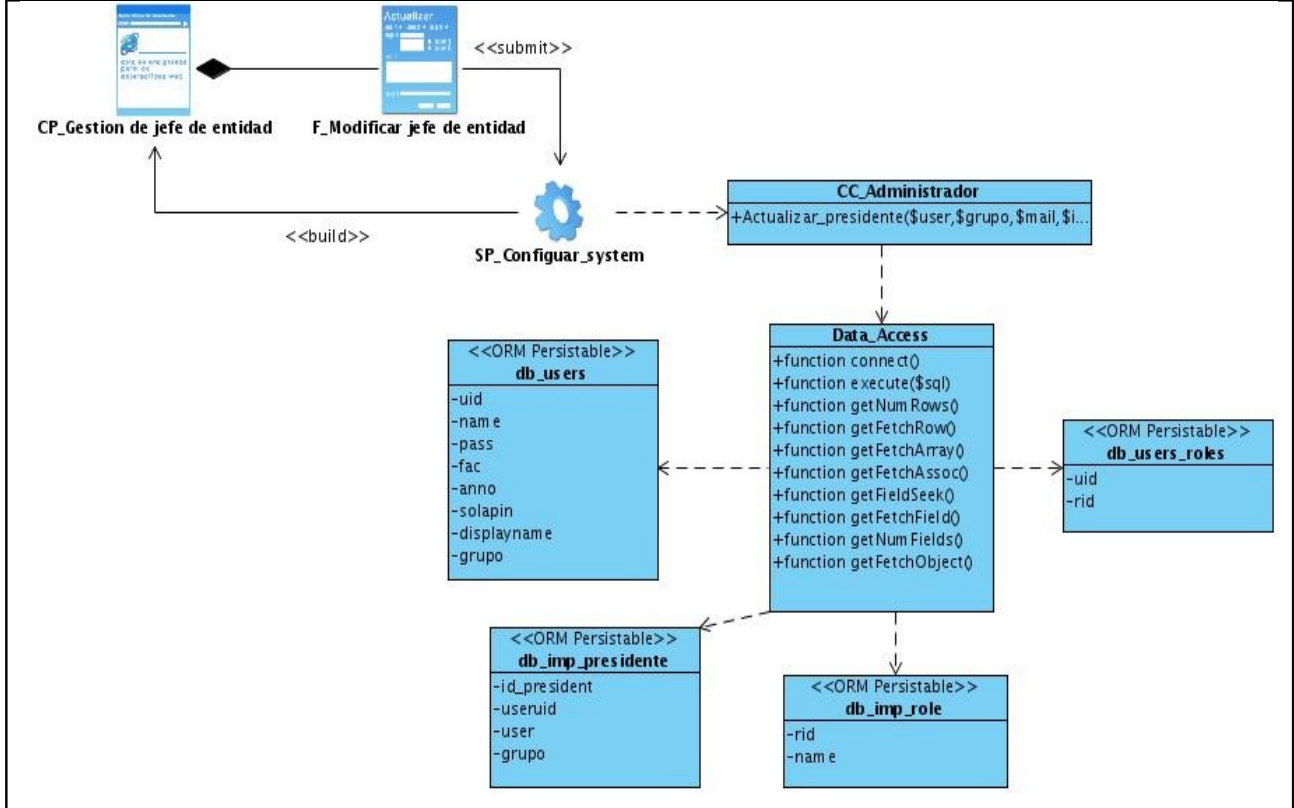
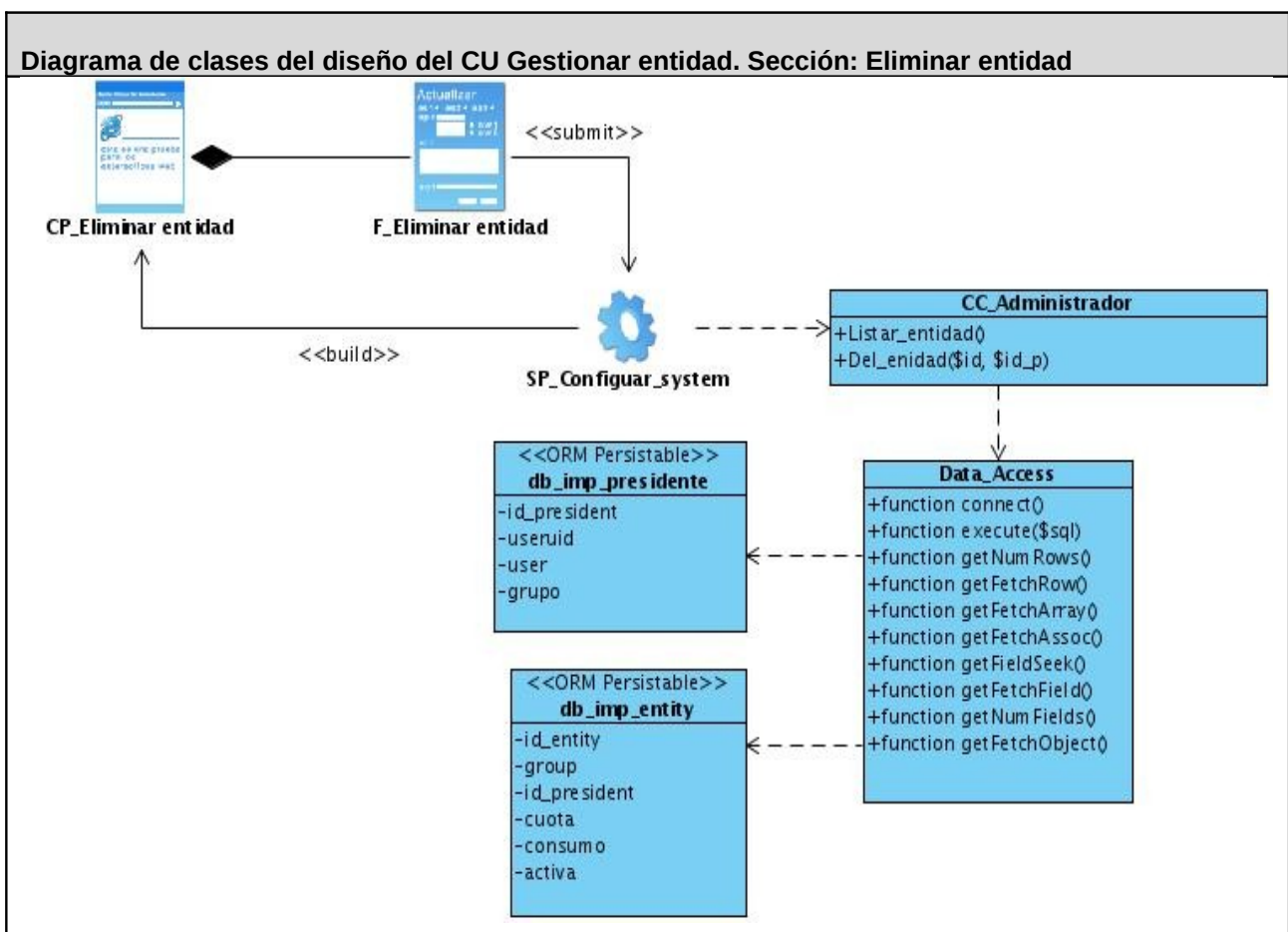
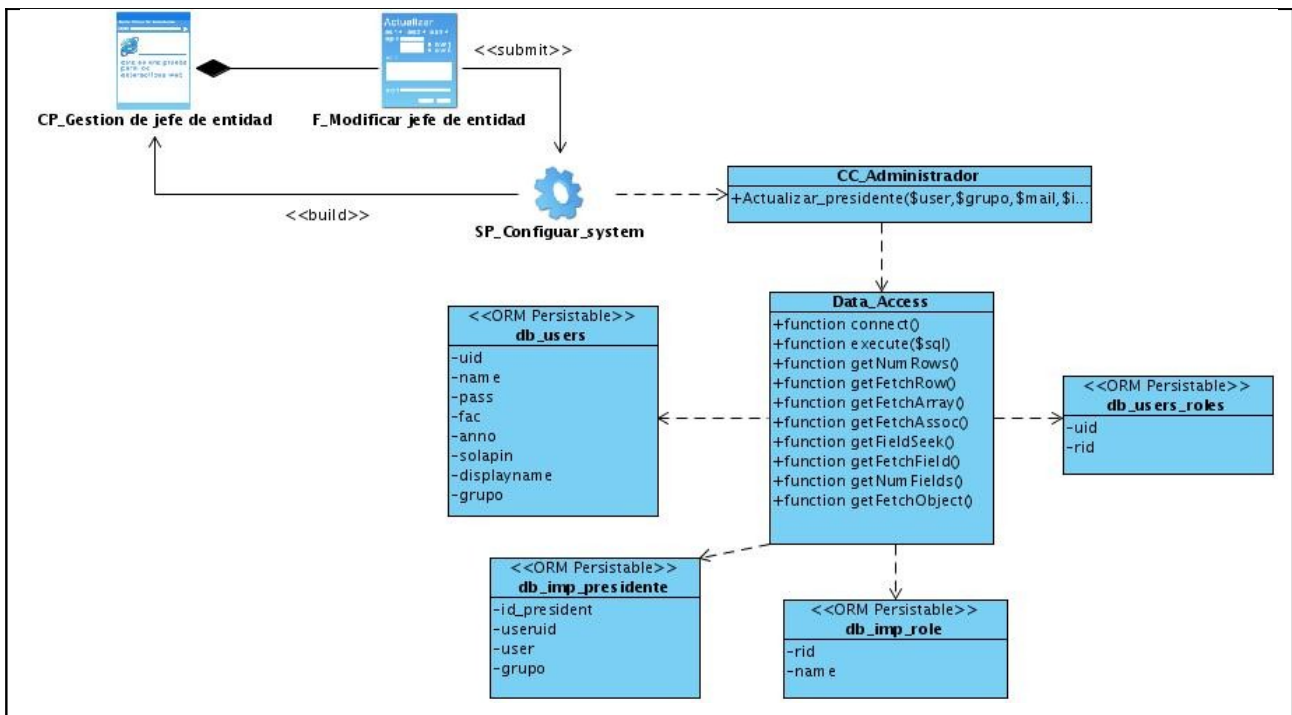


Diagrama de clases del diseño del CU Gestionar entidad. Sección: Modificar entidad



Anexo 4. Diagramas de interacción.

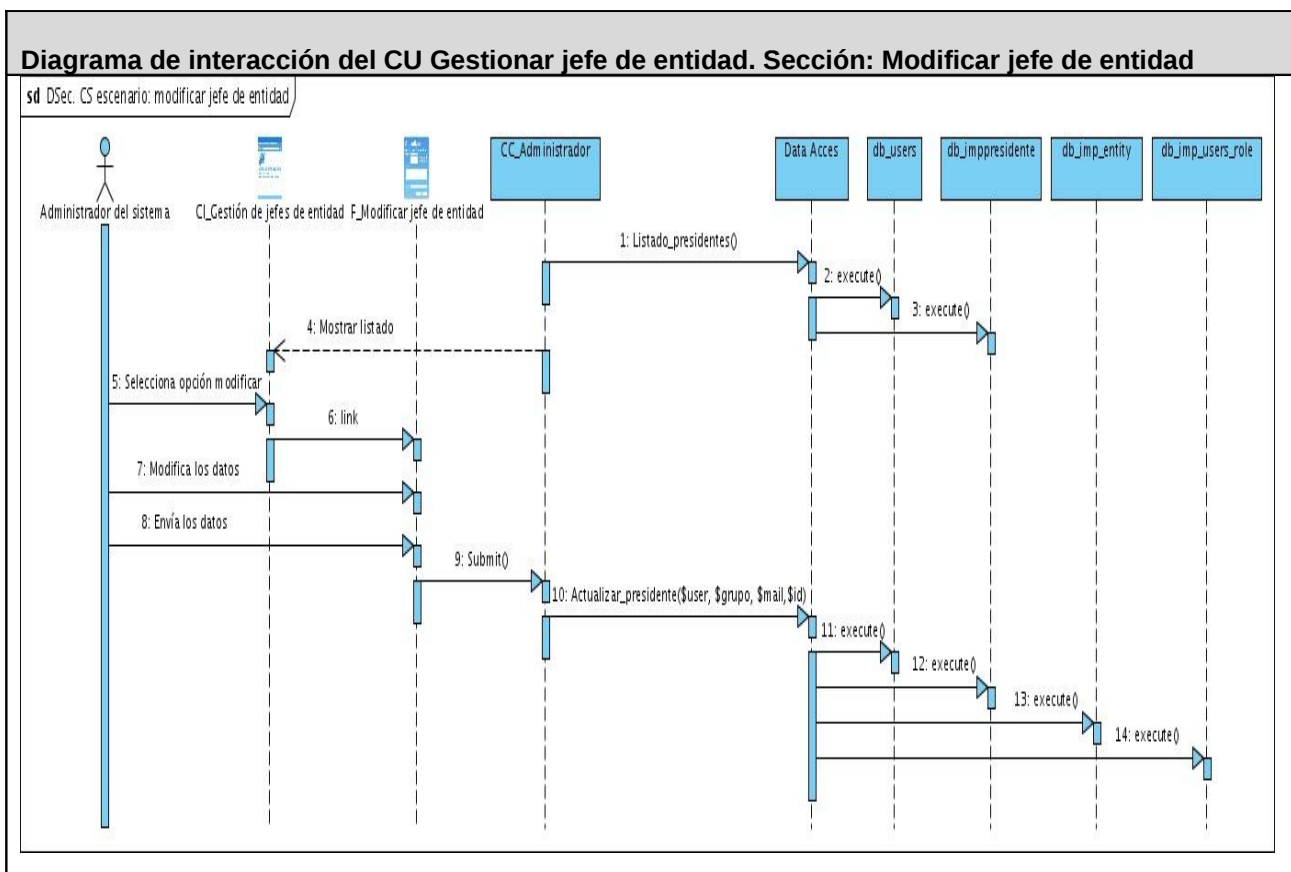
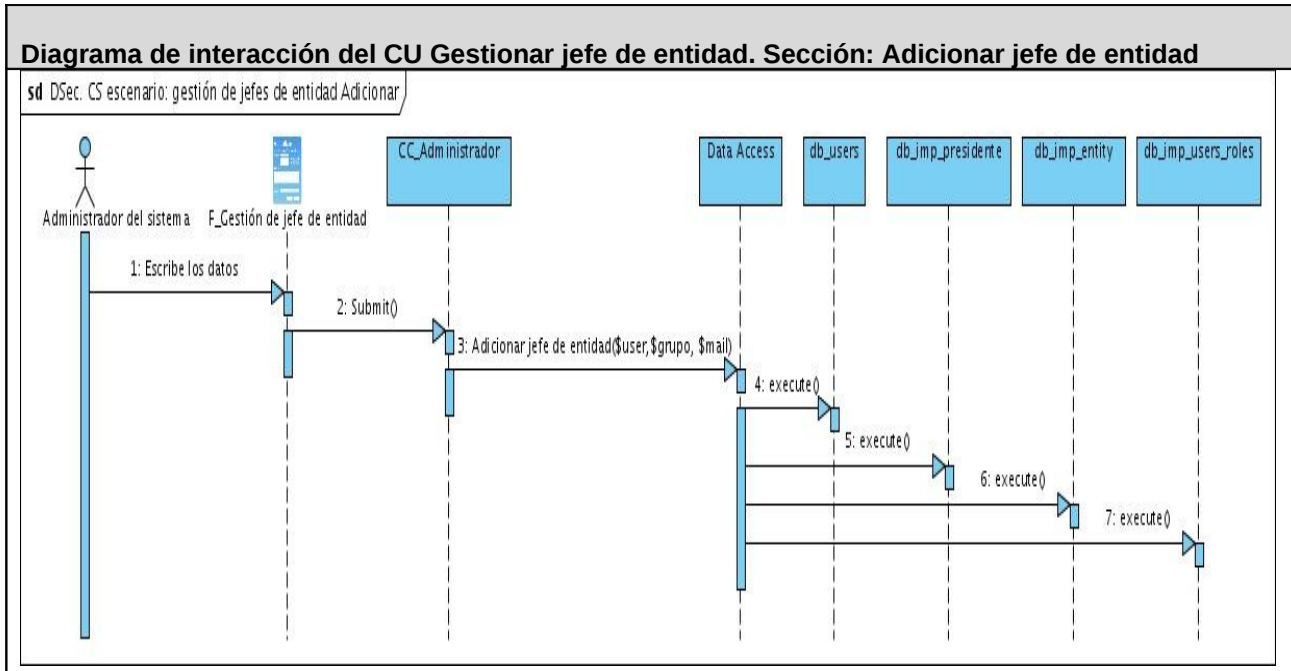


Diagrama de interacción del CU Asignar cuota

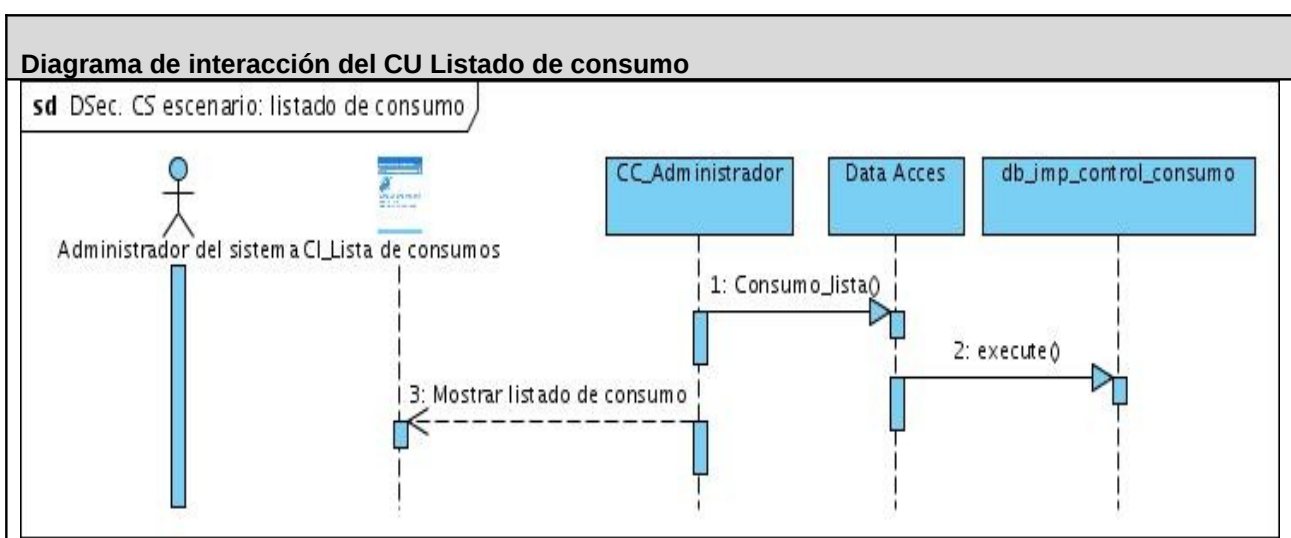
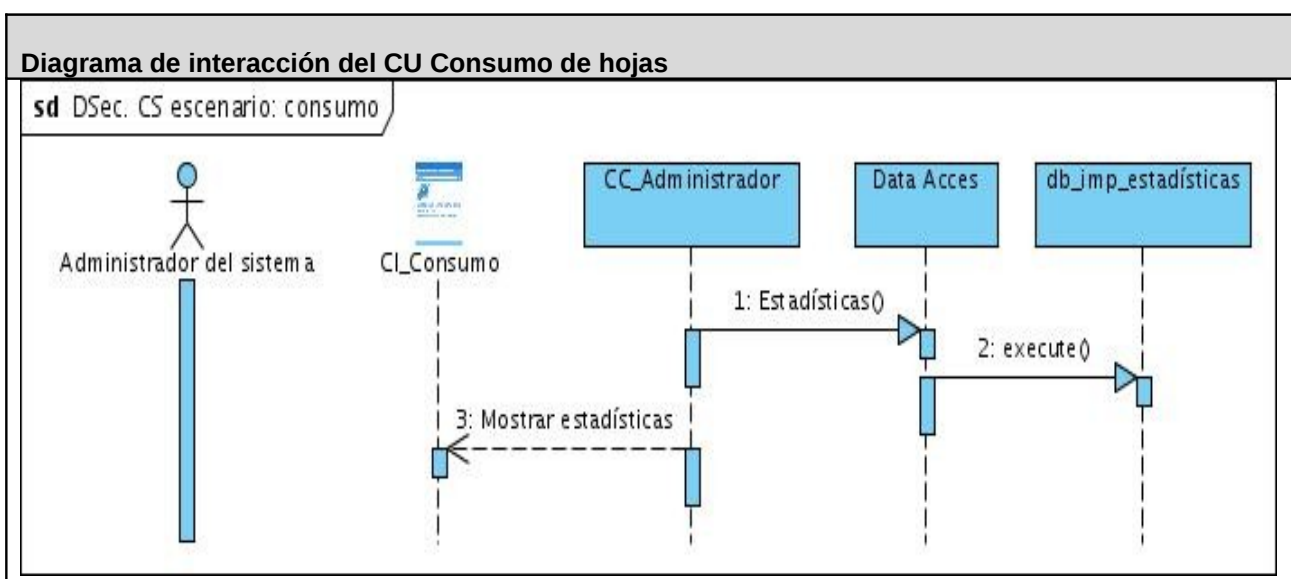
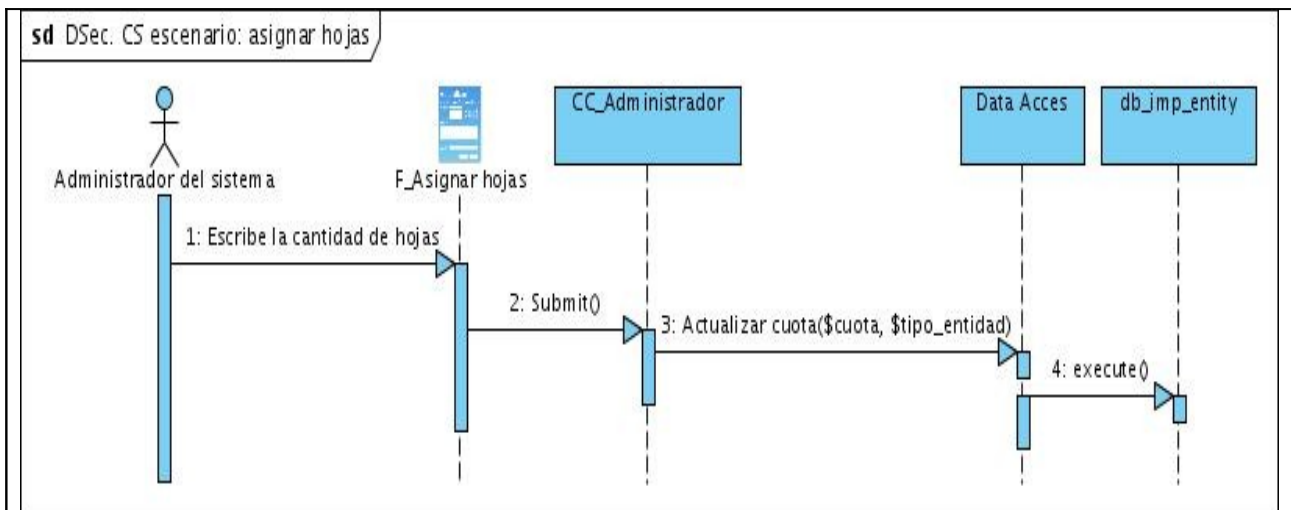
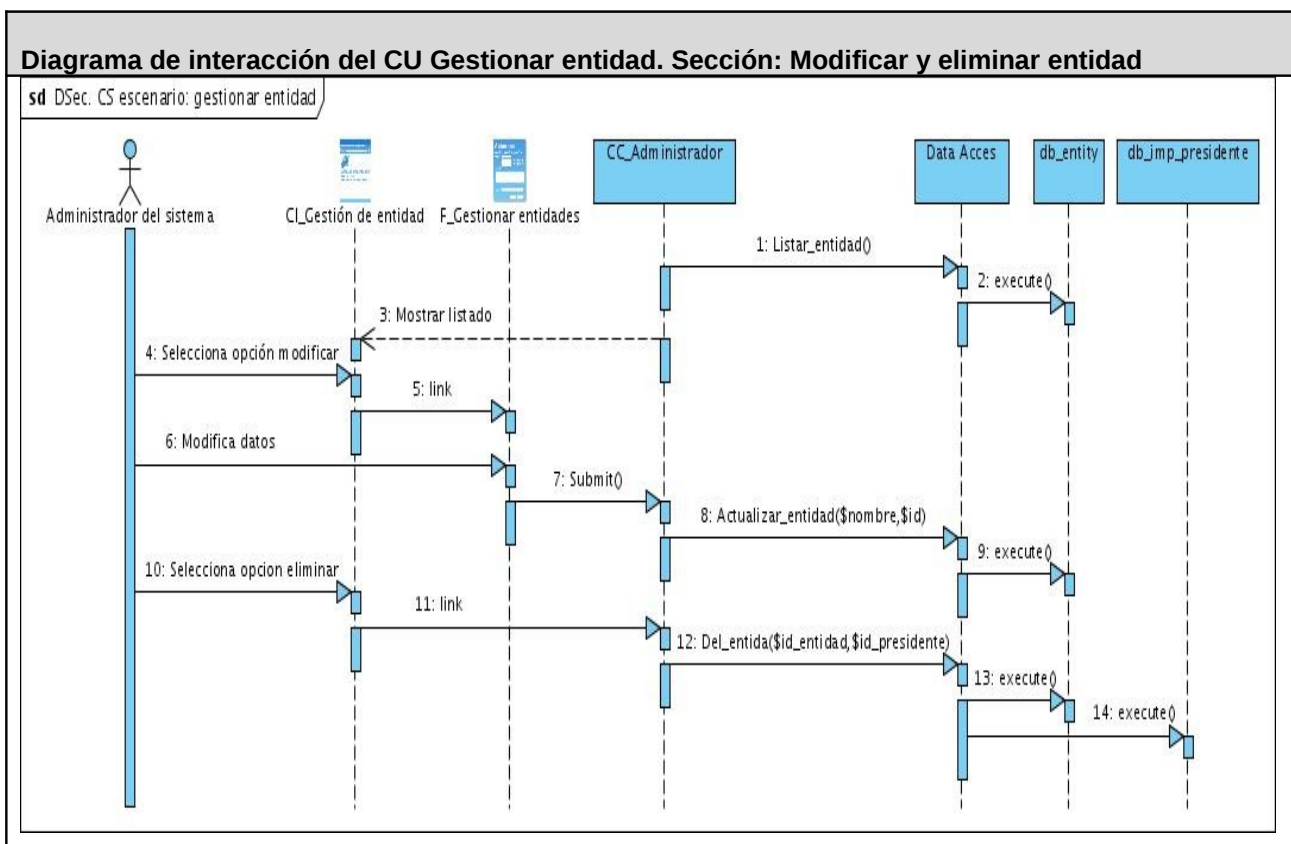
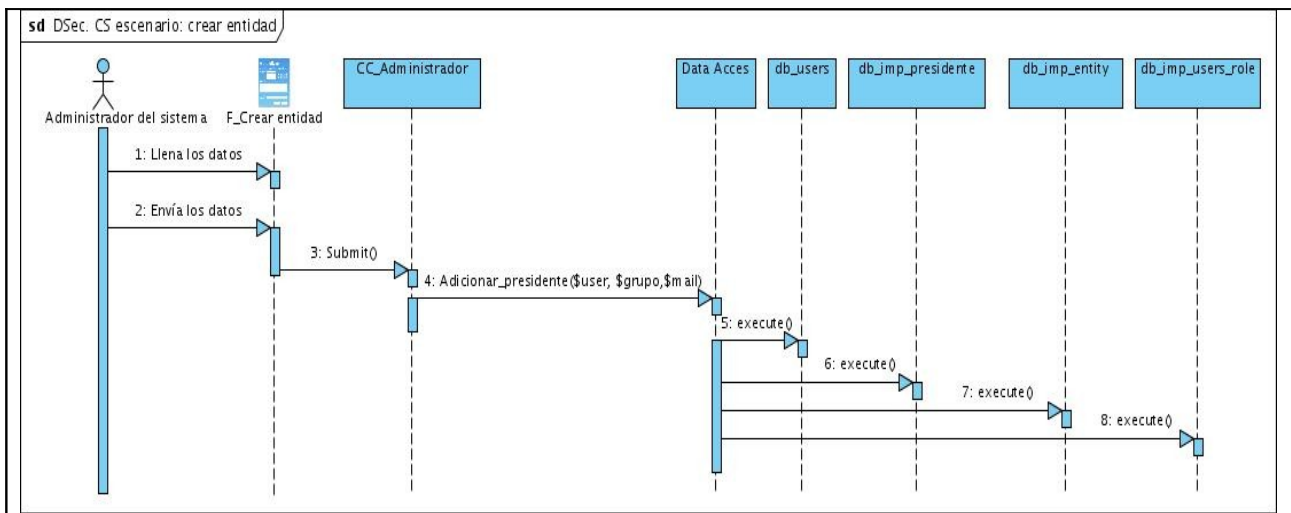


Diagrama de interacción del CU Gestionar entidad. Sección: Crear entidad



Anexo 5. Descripción de las tablas principales

Nombre: presidente
Tipo de Clase: entidad

Descripción: Se utiliza para conocer quién es presidente de brigada de un grupo determinado	
Atributo	Tipo
Id_president	int
usersuid	int
entity	int

Nombre: users	
Tipo de Clase: entidad	
Descripción: Guarda un registro de todos los usuarios que acceden al sistema.	
Atributo	Tipo
uid	int
name	varchar
pass	varchar
mail	int
mode	tinyint
sort	tinyint
threshold	tinyint
theme	varchar
signature	varchar
created	int
access	int
login	int
status	tinyint
timezone	varchar
lenguaje	varchar
picture	varchar
init	varchar
data	varchar
nombre_usuario	varchar
anno	int
fac	int
displayname	varchar
grupo	int
solapin	varchar

Nombre: users_role	
Tipo de Clase: entidad	
Descripción: Permite conocer el rol que juega cada usuario en el sistema.	
Atributo	Tipo
uid	int

rid	int

Nombre: role	
Tipo de Clase: entidad	
Descripción: Contiene los diferentes tipos de roles de usuario.	
Atributo	Tipo
rid	int
name	varchar

Nombre: imp_entity	
Tipo de Clase: entidad	
Descripción: Almacena un listado con los grupos, la cuota asignada a cada uno y el consumo.	
Atributo	Tipo
Id_entity	int
group	varchar
Id_president	int
cuota	int
consumo	int
activa	int

Nombre: imp_pedido	
Tipo de Clase: entidad	
Descripción: Guarda un listado con las solicitudes de impresión realizadas.	
Atributo	Tipo
Id_pedido	int
fecha	date
Id_beneficiario	int
unit	int
Id_president	int
Id_entity	int
Id_file	int

Nombre: imp_format	
Tipo de Clase: entidad	
Descripción: Almacena los tipos de formatos de impresión.	

Atributo	Tipo
Id_format	int
name_format	varchar

Nombre: imp_file	
Tipo de Clase: entidad	
Descripción: Guarda los ficheros que se van a imprimir.	
Atributo	Tipo
Id_file	int
uid	int
name_file	varchar
Id_format	int
path	varchar
copias	int

Nombre: imp_autorizacion	
Tipo de Clase: entidad	
Descripción: Almacena las autorizaciones realizadas por el jefe de entidad	
Atributo	Tipo
Id_autorizacion	int
Id_president	int
user	varchar
group	varchar
consumo	int

Nombre: imp_pedido_pre_aut	
Tipo de Clase: entidad	
Descripción: Guarda las peticiones de autorización realizadas por los estudiantes al jefe de brigada	
Atributo	Tipo
Id_pedido_pre_aut	int
user	varchar
group	varchar
pedido	int

Anexo 6. Prototipos de interfaz.

Prototipo de interfaz del Caso de Uso Solicitar impresión

Intranet | Correo-e | Jabber | Inter-nos

UCI Dirección de Información

Buscar

Servicios | Catálogos | Base de Datos | Biblioteca Digital | Tutoriales | Quiénes Somos | Mapa del Sitio | RSS

Principal

Impresion

Ver | Editar | Índice

Solicitud de impresión	Ayuda
<ul style="list-style-type: none"> • BRIGADA • FEU 	<ol style="list-style-type: none"> 1. Seleccionar el tipo de solicitud (FEU o BRIGADA) 2. Seleccionar la cantidad de hojas que desea imprimir 3. Enviar el documento para ser impreso

informacion@uci.cu | 835 8440

Intranet | Correo-e | Jabber | Inter-nos

UCI Dirección de Información

Buscar

Servicios | Catálogos | Base de Datos | Biblioteca Digital | Tutoriales | Quiénes Somos | Mapa del Sitio | RSS

Principal

Impresion

Ver | Editar | Índice

Solicitud de impresión	Ayuda
Solicitar autorización para imprimir : <input type="text" value="2"/> <input type="button" value="Solicitar"/>	<ol style="list-style-type: none"> 1. Seleccionar el tipo de solicitud (FEU o BRIGADA) 2. Seleccionar la cantidad de hojas que desea imprimir 3. Enviar el documento para ser impreso

informacion@uci.cu | 835 8440

Prototipo de interfaz del Caso de Uso Autorizar impresión

Intranet | Correo-e | Jabber | Inter-nos

UCI Dirección de Información

Servicios | Catálogos | Base de Datos | Biblioteca Digital | Tutoriales | Quiénes Somos | Mapa del Sitio | RSS

Principal

Impresion

Ver | Editar | Índice

Autorización de solicitud de impresión	Ayuda
Hojas Disponibles: 50 dirinfo <input type="text" value="2"/> <input type="button" value="autorizar"/> <input type="button" value="cancelar"/>	1. Autorizar o denegar las presolitudes pendientes

informacion@uci.cu | 835 8440

Prototipo de interfaz del Caso de Uso Enviar solicitud de impresión

Intranet | Correo-e | Jabber | Inter-nos

UCI Dirección de Información

Servicios | Catálogos | Base de Datos | Biblioteca Digital | Tutoriales | Quiénes Somos | Mapa del Sitio | RSS

Principal

Impresion

Ver | Editar | Índice

Solicitud de impresión	Ayuda
La cantidad de hojas autorizadas a imprimir es: 2 Elija el archivo y presione el boton Enviar. (1) Elegir Archivo: <input type="text" value="E:\Tesis.doc.bt"/> <input type="button" value="Examinar..."/> (2) Elegir formato: <input type="text" value="Libro"/> (3) Número de copias : <input type="text" value="2"/> (4) Enviar Archivo: <input type="button" value="Enviar"/> <input type="button" value="Cancelar"/>	1. Seleccionar el documento a imprimir 2. Seleccionar el formato <ul style="list-style-type: none"> o Simple o Doble cara o Libro 3. Seleccionar la cantida de copias deseadas 4. Enviar la solicitud y espera a ser autorizado por el Jefe de entidad (Presidente de la FEU o Jefe de BRIGADA)

informacion@uci.cu | 835 8440

Prototipo de interfaz del Caso de Uso Impresión de documentos

Intranet | Correo-e | Jabber | Inter-nos

UCI Dirección de Información

[Input] Buscar

Servicios | Catálogos | Base de Datos | Biblioteca Digital | Tutoriales | Quiénes Somos | Mapa del Sitio | RSS

Principal

Impresion

Ver | Editar | Índice

Imprimir documento | Mostrar listado de documentos impresos | Buscar Información

Fecha	Beneficiado	Entidad	Nºde Hojas	Formato	Nºde copias	Descargar	Notificar	Eliminar
2009-05-10	Randy Llanes Alfonso	10501	2	Simple	0	Tesis.doc	[Icon]	[X]
2009-05-10	Yunel Cabrera Garcia	10505	4	Doble cara	0	Tesis.pdf	[Icon]	[X]

informacion@uci.cu | 835 8440

Prototipo de interfaz del Caso de Uso Mostrar pedidos

Intranet | Correo-e | Jabber | Inter-nos

UCI Dirección de Información

[Input] Buscar

Servicios | Catálogos | Base de Datos | Biblioteca Digital | Tutoriales | Quiénes Somos | Mapa del Sitio | RSS

Principal

Impresion

Ver | Editar | Índice


Imprimir documento | Mostrar listado de documentos impresos | Buscar Información

Fecha	Beneficiado	Unidades	Registrado por
2009-05-10	Yunel Cabrera Garcia	4	administrador de sistema

informacion@uci.cu | 835 8440

Prototipo de interfaz del Caso de Uso Buscar información

Intranet | Correo-e | Jabber | Inter-nos



Buscar

Servicios | Catálogos | Base de Datos | Biblioteca Digital | Tutoriales | Quiénes Somos | Mapa del Sitio | RSS

Principal

Impresion

Grupo :

informacion@uci.cu | 835 8440

Prototipo de interfaz del Caso de Uso Administrar sistema. Escenario: gestión de jefes de entidad

Intranet | Correo-e | Jabber | Inter-nos



Buscar

Servicios | Catálogos | Base de Datos | Biblioteca Digital | Tutoriales | Quiénes Somos | Mapa del Sitio | RSS

Principal

Impresion

Usuario : Grupo :

Usuario	Nombre	Correo	Grupo	Modificar
ycabrerag	Yunel Cabrera Garcia	ycabrerag@estudiantes.uci.cu	Feu-10	
dirinfo	administrador de sistema	biblio@uci.cu	10501	
ambatista	Ana Maria batista	ambatista@estudiantes.uci.cu	08101	

informacion@uci.cu | 835 8440

Prototipo de interfaz del Caso de Uso Administrar sistema. Escenario: asignar hojas

Intranet | Correo-e | Jabber | Inter-nos

UCI Dirección de Información

Buscar

Servicios | Catálogos | Base de Datos | Biblioteca Digital | Tutoriales | Quiénes Somos | Mapa del Sitio | RSS

Principal

Impresion

Ver | Editar | Índice

Gestión de jefes de brigada	Asignar hojas	Consumo de hojas	Lista de Consumo de hojas	Crear entidad	Gestión de entidad
-----------------------------	---------------	------------------	---------------------------	---------------	--------------------

Asignacion de cuotas

Cuota:

BRIGADAS
 FEU

Asignar

informacion@uci.cu | 835 8440

Prototipo de interfaz del Caso de Uso Administrar sistema. Escenario: consumo de hojas

Intranet | Correo-e | Jabber | Inter-nos

UCI Dirección de Información

Buscar

Servicios | Catálogos | Base de Datos | Biblioteca Digital | Tutoriales | Quiénes Somos | Mapa del Sitio | RSS

Principal

Impresion

Ver | Editar | Índice

Gestión de jefes de brigada	Asignar hojas	Consumo de hojas	Lista de Consumo de hojas	Crear entidad	Gestión de entidad
-----------------------------	---------------	------------------	---------------------------	---------------	--------------------

Consumo Total	Consumo Anual	Consumo Mensual
Consumo: 11 Total: 345120 Porciento: 0.003187297172 %	Consumo: 16 Total: 345120 Porciento: 0.00463606861382 %	Consumo: 11 Total: 28760 Porciento: 0.038247566064 %

informacion@uci.cu | 835 8440

Prototipo de interfaz del Caso de Uso Administrar sistema. Escenario: lista de consumo de hojas

Intranet | Correo-e | Jabber | Inter-nos

UCI Dirección de Información

Buscar

Servicios | Catálogos | Base de Datos | Biblioteca Digital | Tutoriales | Quiénes Somos | Mapa del Sitio | RSS

Principal

Impresion

Ver | Editar | Índice

Gestión de jefes de brigada	Asignar hojas	Consumo de hojas	Lista de Consumo de hojas	Crear entidad	Gestión de entidad
-----------------------------	---------------	------------------	---------------------------	---------------	--------------------

Mes	Total	Consumo	Por ciento
abril - 2009	28760	16	0.0556 %
mayo - 2009	28760	37	0.1287 %

informacion@uci.cu | 835 8440

Prototipo de interfaz del Caso de Uso Administrar sistema. Escenario: crear entidad

Intranet | Correo-e | Jabber | Inter-nos

UCI Dirección de Información

Buscar

Servicios | Catálogos | Base de Datos | Biblioteca Digital | Tutoriales | Quiénes Somos | Mapa del Sitio | RSS

Principal

Impresion

Ver | Editar | Índice

Gestión de jefes de brigada	Asignar hojas	Consumo de hojas	Lista de Consumo de hojas	Crear entidad	Gestión de entidad
-----------------------------	---------------	------------------	---------------------------	---------------	--------------------

Entidad:

Usuario del jefe de entidad:

Estudiante :

Trabajador :

informacion@uci.cu | 835 8440

Prototipo de interfaz del Caso de Uso Administrar sistema. Escenario: crear entidad

Intranet | Correo-e | Jabber | Inter-nos

UCI Dirección de Información

Buscar

Servicios | Catálogos | Base de Datos | Biblioteca Digital | Tutoriales | Quiénes Somos | Mapa del Sitio | RSS

Principal

Impresion

Ver | Editar | Índice

Gestión de jefes de brigada	Asignar hojas	Consumo de hojas	Lista de Consumo de hojas	Crear entidad	Gestión de entidad
			Entidad	Modificar	Eliminar
			10505		
			10506		
			10301		
			10304		
			10102		
			09101		
			10501		
			10503		
			10407		
			Feu-10		
			08101		
			Feu-09		
			10104		
			09201		

informacion@uci.cu | 835 8440

GLOSARIO DE TÉRMINOS

UML: Lenguaje Unificado de Modelado (UML, por sus siglas en inglés, Unified Modelling Language) es el lenguaje de modelado de sistemas de software más conocido en la actualidad.

Drupal: Sistema de administración de contenidos utilizado para la implementación de la aplicación.

MySQL: Sistema gestor de base de datos utilizado para almacenar los datos manejados por la aplicación.

RUP: Proceso unificado de desarrollo de software. Metodología utilizada para elaborar la documentación del software.

PHP: Lenguaje de programación utilizado del lado del servidor para implementar la aplicación.

HTML: Lenguaje de etiquetado de hipertexto

SGBD: Sistema gestor de base de datos.

CMS: Sistema de manipulación de contenidos.

GPL: Licencia de software libre que está orientada principalmente a proteger la libre distribución, modificación y uso de software.

IDE: Entorno de desarrollo integrado o en inglés Integrated Development Environment (IDE). Es un programa compuesto por un conjunto de herramientas para un programador desde el que se pueden editar programas, compilarlos y depurarlos.

Libre: O Software Libre se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software.

XML: Es el acrónimo de eXtensible Markup Language (lenguaje de marcado extensible) desarrollado por el World Wide Web Consortium (W3C).

LAN (Local Area Network): En informática designa a una red de área local. Es una red de computadoras que generalmente se encuentran dentro de un edificio o un piso.

WAN (Wide Area Network): Red de Área Amplia, es un tipo de [red de computadoras](#) capaz de cubrir distancias desde unos 100 Km. hasta unos 1000 Km., brinda el servicio a un país o un continente.

Bugzilla: es una base de datos que sirve para que el usuario reporte todos los bugs o errores que encuentre en su GNOME.

CGI (Common Gateway Interfaces): Es un conjunto de reglas que describen de que manera un servidor de web se comunica con otros elementos de software en la misma computadora, y como el programa CGI “conversa” con el servidor de web.

FastCGI: es una evolución natural de CGI.

SSL (Secure Sockets Layer): Protocolo diseñado por la empresa Netscape para proveer comunicaciones encriptadas en internet.