



Universidad de las Ciencias Informáticas

FACULTAD 7

Título: Portal Web para la gestión de la
información en el proyecto productivo APS

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

Autores: Magdalena Tamayo Boza

Marcos Rodríguez Morales

Tutores: Ing. David Barreto Medina

Lic. Mirna Cabrera Hernández

Ciudad de La Habana, Julio 2008

“Año 50 de la Revolución”

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

Para que así conste firmamos la presente a los 9 días del mes de Julio del año 2008.

Autora
Magdalena Tamayo Boza

Autor
Marcos Rodríguez Morales

Tutor
Ing. David Barreto Medina

Tutora
Ing. Mirna Cabrera Hernández

Ing. David Barreto Medina:

Graduado de Ingeniería Industrial en la UHO. Profesor Instructor. Departamento Ciencias Básicas Facultad 7. Universidad de las Ciencias Informáticas.

Email: dbarreto@uci.cu

Ing. Mirna Cabrera Hernández:

Graduada de Ing. SAD Técnicos y Económicos en el año 1986 en el ISPJAE. Posee categoría docente de Profesor Auxiliar Adjunto y cursa la Maestría de Gestión de Proyectos Informáticos. Se desempeña como Líder del Proyecto APS en la Empresa Softel.

Email: mirna@softel.cu y mirna@infomed.sld.cu

*"Nos forjaremos en la acción cotidiana creando un
hombre nuevo con una nueva técnica."*

Ernesto Guevara de la Serna.

Magdalena Tamayo Boza:

A Fidel y la Revolución por darme la posibilidad de forjarme como ingeniera en la Universidad del futuro.

A toda mi familia por su apoyo, confianza y por haberme guiado por el camino correcto.

A todas las personas (estudiantes y profesores) que me han acompañado desde el primer año de la Universidad y en especial a los que han brindado su apoyo siempre que lo he necesitado.

A mi novio Yunier J. Trujillo por su amor, cariño y por darme todo el apoyo y ayuda que necesité. Gracias mi nene.

Marcos Rodríguez Morales:

A mi madre, que me dio libertad para tomar mis propias decisiones, te quiero. A mi hermano Miguel por todos los piñazos que nos dimos cuando niños y por demostrarme que en la vida hay que imponer respeto para llegar a ser alguien. A mis hermanitos Leidi Sandra y Alejandro, los quiero mucho aunque nos veamos una vez al año. A mi padre y a toda mi familia de Ciego de Ávila, todos formaron parte de mi educación y les debo mucho más que agradecimientos. A mi Padrastro Roque, que no tuvo la posibilidad de ser universitario pero supo guiarme en la vida, miles de años no alcanzarían para agradecerte lo que has hecho por nosotros. A mi abuelo Alfredo, papa pen, como siempre lo soñaste; tu primer nieto es hoy Teniente e Ingeniero. A mi abuela Marta por ser mi segunda madre. A Eusebio, mi tío; si yo tuviera que ser igual a alguien, trataría de ser como tú, porque tu cumples con todos los principios que para mí deben tener los hombres. A mi tía Tamara, a Loli y a todos mis familiares. Sepan que hoy estoy aquí gracias a ustedes.

Quiero agradecer además a mis amigos, especialmente a Jorge y Maikel por apoyarme en todos los sentidos y estar para mí cuando los necesité, ustedes son mis hermanos para toda la vida. A mis hermanos de la UCI, gracias por todos los momentos que compartimos juntos. A Ariel González (WALLACE). A la nene linda que me enseñó a aprovechar cada minuto; te quiero y eso nunca va a cambiar, gracias por tu ayuda. A todos mis amigos de Cienfuegos, Armando, el Zurdo, Yudiel; son muchos pero a todos los quiero. A esa persona que siempre me ha querido, pero las cosas hay que saber merecerlas. A todos los que hicieron posible que este sueño fuera realidad aportando sus conocimientos. A Jorge del Toro y a Magdalena; mi compañera de tesis. A los profesores que me apoyaron. A la Revolución, a las FAR y a Fidel por darme la posibilidad de estudiar en una Universidad tan bella.

A toda mi familia en especial a mis padres Daylys y William, por apoyarme en todo momento, confiar plenamente en mí y educarme de la mejor forma, por todo esto y muchas otras cosas, gracias, los quiero con la vida.

A mi hermana Odalís, por ser mi fuente de inspiración para la lucha diaria para ser alguien en la vida.

A mi sobrinita Jessica, para que se esfuerce y aprenda todos los días cosas nuevas.

Magdalena Tamayo Boza.

A mi madre, a todos mis amigos, a la Revolución.

Marcos Rodríguez Morales.

RESUMEN

Hoy en día, la gestión de proyecto cobra gran importancia para las empresas informáticas. Lo mismo ocurre en la Universidad de las Ciencias Informáticas, donde la producción está organizada por proyectos, en su mayoría de gran tamaño y complejidad por lo que se hace muy difícil controlar y gestionar toda la información que generan.

El objetivo principal del presente trabajo es la implementación de un Portal Web que permita la gestión de la información en el proyecto productivo APS, sustituyendo el entorno de desarrollo utilizado actualmente. El portal no solo tendrá en cuenta el factor informativo, sino que generará toda la información y los reportes necesarios.

Se exponen las tecnologías y herramientas utilizadas, siguiendo las políticas de la arquitectura trazadas por la facultad en la estrategia de la creación de Portales Web. En el desarrollo de la aplicación se logra la unión de algunas herramientas utilizadas en el proyecto como: el DotProject y el Subversión. Para el desarrollo del sistema son utilizados el CMS Joomla, con PHP como lenguaje de programación y MySQL como Sistema Gestor de Base de Datos.

El desarrollo de la aplicación propuesta permitirá al proyecto llevar un control de la asistencia a las actividades productivas, registrar los resultados obtenidos por los estudiantes en los diferentes cursos, así como mostrar la información y estructura del proyecto.

CONTENIDO

RESUMEN	VII
INTRODUCCIÓN.	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	3
1.1 Portales Web (Definición).	3
1.2 Sistemas de información para la salud en Cuba.	4
1.3 Patrones de arquitectura y diseño.	4
1.4 Tecnología a utilizar.	7
1.5 Lenguajes de Programación utilizados en el proceso de desarrollo.	9
1.6 Lenguajes de marcas.	11
1.7 Librería ExtJS.	12
1.8 Servidor Web Apache.	13
1.9 Sistema Gestor de Base de Datos.	14
1.10 Lenguaje Unificado de Modelado.	16
1.11 Metodología de desarrollo.	16
1.12 Herramientas de planificación y gestión de proyectos.	18
1.13 Herramientas de gestión de la configuración.	20
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA.....	23
2.1 Modelo de Dominio.	23
2.2 Especificación de los Requerimientos.	24
2.3 Modelo de Casos de Uso del Sistema.	30
CAPÍTULO 3: ANÁLISIS Y DISEÑO.	44
3.1 Análisis.	44
3.2 Diseño.	49
CAPÍTULO 4: IMPLEMENTACIÓN.	59
4.1 Modelo de Implementación.	59
4.2 Descripción de los métodos del negocio.	61
CONCLUSIONES.	64
RECOMENDACIONES.	65
BIBLIOGRAFÍA	66
REFERENCIAS BIBLIOGRÁFICAS.....	68
GLOSARIO DE TÉRMINOS.	70

INTRODUCCIÓN.

Actualmente en el mundo, la producción de software constituye un importante renglón que aporta grandes beneficios económicos a los países. En Cuba existen empresas que se dedican a la creación de software para diversas entidades como es el caso de la educación, la salud, el comercio, entre otras.

Con el desarrollo de las Tecnologías de la Información y las Comunicaciones (TIC) en Cuba, se espera alcanzar un elevado nivel en la informatización de la salud. La medicina cubana ha logrado muy buena reputación a nivel nacional e internacional, gracias al arduo trabajo en las últimas décadas. Además, con la introducción de las computadoras y equipos técnicos, se han facilitado las condiciones a los médicos y demás profesionales de esta rama, lo que propicia un trabajo con mayor eficiencia y rapidez.

Una de las empresas especializadas en brindar soluciones informáticas para la Salud en Cuba es SOFTEL, quien inició sus actividades en 1986, con el fin de lograr el desarrollo de la ingeniería y comercialización de software. Inicialmente sus servicios informáticos no eran solo dirigidos al sector de la Salud, como es en la actualidad, sino que estaban dirigidos a varios sectores como: la gestión empresarial y el turismo.

En la Universidad de las Ciencias Informáticas (UCI) además del proceso docente educativo existen diferentes proyectos productivos encaminados a disímiles temas con el objetivo de apoyar al desarrollo de la informatización del país. Cada facultad cuenta con un perfil y la labor de los proyectos tributa a los mismos.

En el caso de la facultad 7 se trabaja con el propósito de informatizar la Salud cubana, donde existe el proyecto Sistema de Información para la Salud (SISalud) el cual pertenece el Área Temática de Atención Primaria para la Salud (APS), que está en convenio con la empresa SOFTEL y radica en la Infraestructura Productiva (IP) de la UCI.

En el proyecto se desea tener un Portal Web que brinde la información que gestiona el mismo, ya que con la presencia de una aplicación de este tipo, sería más fácil garantizar la gestión de la información y con esta el cumplimiento de los objetivos trazados.

Con la aplicación del portal propuesto se lograría un mejor funcionamiento del proyecto. Debido a que contará con un mecanismo para el control del tiempo de máquina de los estudiantes, de forma tal que sea factible aprovechar al máximo el horario de producción coordinado con el horario docente. También se desea obtener un listado de las personas vinculadas a este proyecto de forma tal que se pueda conocer las personas que cumplan años en el día, así como algunos datos de interés.

Actualmente el proyecto productivo APS no gestiona de una forma eficiente la información. No se encuentran integradas las herramientas de trabajo que se utilizan en el mismo. Ni se cuenta con mecanismos para el control del tiempo de máquina y el control diario de la asistencia a los puestos de trabajo.

Según lo planteado anteriormente se tiene como **problema a resolver** que el Entorno de Desarrollo actual del proyecto productivo APS no garantiza una adecuada gestión de la información del mismo.

Se tiene como **objeto de estudio**:

Proceso de gestión de la información en los proyectos productivos en la Universidad de las Ciencias Informáticas.

El **campo de acción** se enmarca en Proceso de gestión de la información en el proyecto productivo APS.

El **objetivo general** que se persigue es:

Desarrollar una aplicación que integre los procesos de control de versiones y gestión de la información que se llevan a cabo en el proyecto productivo APS, añadiendo otras funcionalidades para perfeccionar el sistema de trabajo en el mismo.

Para dar solución al objetivo que se plantea es necesario realizar las siguientes tareas:

1. Realizar un estudio del estado del arte de las herramientas de gestión de proyectos, control de versiones y los portales web.
2. Identificar elementos de integración como los Portales Web y herramientas de control de versiones.
3. Modelar, siguiendo el Proceso Unificado de Rational (RUP), los Flujos de Trabajo "Modelamiento del Negocio", "Gestión de Requerimientos", "Análisis", "Diseño" e "Implementación".
4. Implementar el Portal Web de APS.

Al realizar el portal web se podrá Con el desarrollo de la aplicación se obtendrá un Portal Web que posibilite la gestión de la información, planificación del control de versiones, gestión del tiempo de máquina, control de asistencia y reportes de la misma. Lo que permitirá garantizar una mejor gestión de la información en el proyecto productivo APS.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.

Introducción.

En el actual capítulo se exponen las principales características y elementos teóricos que sustentan la investigación. Además se brindan los conceptos y definiciones fundamentales relacionados con las herramientas de gestión de proyectos, control de versiones y portales web.

Seguidamente se expone un análisis de las características y funcionalidades de técnicas y tecnologías posibles a utilizar en el desarrollo del trabajo propuesto. Y se presentará una breve valoración de la escogida.

1.1 Portales Web (Definición).

Un Portal Web es un punto centralizado para acceder a diversas aplicaciones, información y otros recursos de internet o en este caso de una empresa o proyecto. Las personas usaban el término "página de Internet" para referirse al sitio en que se encontraba publicada su empresa, pero el concepto de Portal está resultando más atractivo, aún cuando el término más apropiado para referirse a los sitios que se encuentran publicados en Internet debiera ser el de Sitio Web (por su denominación en inglés "*website*").

Un portal de Internet es un sitio web cuyo objetivo es ofrecer al usuario, de forma fácil e integrada, el acceso a una serie de recursos y de servicios, entre los que suelen encontrarse buscadores, foros, documentos, aplicaciones, compra electrónica, etc. Principalmente están dirigidos a resolver necesidades específicas de un grupo de personas o de acceso a la información y servicios de una institución pública o privada. [1]

Existen diferentes clasificaciones para los portales, como son: portales generales o genéricos que van a estar dirigidos a un amplio público y ofrecen información de tipo general y los portales específicos o especializados, que ofrecen una información específica para un determinado público. En resumen, un portal es un punto de partida para la navegación a través de páginas web y la utilización de herramientas o servicios. [2]

Actualmente en la Facultad 7 existen proyectos que tienen su propio portal web, como el proyecto Grupo de Procesamiento de Imágenes (GPI), el cual lo identifica y mantiene al tanto de todo lo que acontece en la Universidad y la Facultad. Con el portal de APS se identificará el proyecto, se brindará información productiva, además se realizará es una aplicación que integre varias herramientas, se pueden realizar otras funcionalidades como: la gestión del tiempo de maquina y el control de la asistencia.

Este portal se clasifica en específicos o especializados, pues su información es específicamente para los miembros del proyecto APS, aunque tengan acceso cualquier persona de la UCI.

1.2 Sistemas de información para la salud en Cuba.

Como parte de los esfuerzos por elevar los niveles de eficiencia en las aplicaciones informáticas el sistema cubano de salud se ha propuesto la creación de sistemas como: el Registro Informatizado de Salud, el cual está constituido por módulos relacionados entre ellos, este sistema posibilita obtener información en tiempo real para la toma de decisiones en los diferentes niveles de dirección, el Sistema de Gestión Hospitalaria brinda información acerca de conferencias y otras noticias relacionadas con la medicina.

En el área de la salud cubana existen numerosos servicios que gestionan la información, la Biblioteca Médica Nacional (BMN) brinda como: sala de lectura, búsquedas bibliográficas, referencias, fotocopias, medios audiovisuales, traducción oral e información señal, además consta con un servicio de Información Especial que consistente en la preparación y entrega de información secundaria a través de las siguientes modalidades o formas de entrega: Información Especial Diaria, Semanal, Mensual y Diseminación Selectiva de la Información.

1.3 Patrones de arquitectura y diseño.

Los patrones son el amplio repertorio de principios generales basados en la experiencia que guían la creación de un software.

1.3.1 Arquitectura en Capas.

Actualmente la arquitectura basada en capas ha tomado un papel protagónico en el desarrollo de aplicaciones modernas. Los elementos ubicados en diferentes capas se comunican teniendo en cuentas diferentes permisos de comunicación.

Existen tres posibilidades de comunicación:

- Arquitectura *top-down* (de arriba a abajo) de capas: Los elementos pueden enviar solicitudes a las capas inferiores. Esto genera una cascada de solicitudes de las capas superiores hacia las inferiores. Una arquitectura *top-down* es no estricta si los elementos de una capa superior pueden enviar peticiones directamente a cualquier elemento de cualquiera de las capas inferiores.
- Arquitectura *bottom-up* (de abajo hacia arriba) de capas: Los elementos de capas inferiores alertan a las capas superiores que ha ocurrido un evento de interacción,

un ejemplo de esta capa son los drivers de dispositivos. Esta arquitectura puede, igualmente, ser no estricta en el caso de que los eventos se notifiquen a cualquiera de las capas superiores.

- Arquitectura bidireccional de capas: Es común su implementación mediante dos pilas de capas que se comunican entre sí, mayormente usada en los protocolos de redes de computadoras. [3]

El portal web se desarrollará utilizando la arquitectura de 2 capas, las cuales serían:

- Capa de Presentación: esta va tener todo lo relacionado con la interfaz en la que interactúan los usuarios.
- Capa de Negocio: en esta capa se tienen todos los elementos y métodos que proveen los procesos del negocio. Además contiene los datos persistentes del sistema y estarán agrupados en tablas de la Base de Datos.

En la realización del Portal se utilizó la arquitectura de 2 capas o arquitectura cliente/servidor por poseer las ventajas que a continuación se muestran:

- El desarrollo de aplicaciones en un ambiente de dos capas es mucho más rápido que en ambientes monolíticos (un solo servidor centralizado con terminales de E/S), pero no es necesariamente más rápido que con el ambiente de tres capas.
- Las herramientas para el desarrollo con dos capas son robustas y evaluadas. Las técnicas de prototipo se emplean fácilmente.
- Las soluciones de dos capas trabajan bien en ambientes no dinámicos estables, pero no se ejecutan bien en organizaciones rápidamente cambiantes. [4]

1.3.2 Modelo Vista Controlador.

El patrón de diseño de modelo-vista-controlador es un patrón de diseño establecido para programar.

- Modelo: Es la lógica del negocio, que en muchos casos involucra accesos de almacenamientos de datos, como base de datos relacionales. El equipo de desarrollo que maneja el modelo debe ser experto en escribir programas en algunas tecnologías apropiadas para almacenar y manipular los datos.

- Vista: Es el código que presenta los datos e imágenes en una página Web. El código comprende JSPs (Java Server Pages) y beans de Java que almacenan los datos para ser usados por JSPs.
- Controlador: Es el código que determina el flujo general de la aplicación.

Beneficios del patrón de diseño MVC:

La aplicación de la división modelo-vista-controlador para el desarrollo de aplicaciones Web tiene varios beneficios:

- Más claridad de diseño.
- Facilita el mantenimiento.
- Mayor escalabilidad de soporte, es decir, construir aplicaciones más grandes.
- Puede distribuir el esfuerzo de desarrollo, donde el responsable del desarrollo de la lógica de negocio, del flujo de control, y los diseñadores de páginas Web pueden trabajar de forma independiente.
- Puede más fácilmente migrar aplicaciones legales, debido a que la vista esta separada del modelo y del control y puede ser tolerable a diferentes plataformas y categorías de usuarios.

1.3.3 Bajo Acoplamiento.

Una clase con bajo acoplamiento no depende de “muchas otras” clases, es la idea de tener las clases lo menos ligadas entre sí posible. Por lo que en caso de producirse una modificación en alguna de ellas, se tenga la mínima repercusión posible en el resto de las clases, potenciando la reutilización, y disminuyendo la dependencia entre las mismas.

1.3.4 Alta Cohesión.

La información que almacena una clase debe de ser coherente y estar en la mayor medida posible relacionada con la clase. Cada elemento del diseño debe realizar una labor única dentro del sistema. Ejemplos de una baja cohesión son las que hacen demasiadas cosas. Este patrón mejora la claridad y la facilidad con que se entiende el diseño, simplifica el mantenimiento y las mejoras en funcionalidad y a menudo se genera un bajo acoplamiento.

La alta cohesión y el bajo acoplamiento se pueden separar, aunque están íntimamente ligados, pues si se aumenta la cohesión del sistema o software, se tiene un alto

acoplamiento entre las clases, y por el contrario si se reduce demasiado el acoplamiento, se verá mermada la cohesión.

1.4 Tecnología a utilizar.

1.4.1 Sistema de gestión de contenidos (CMS).

Content Management System, abreviado (CMS), permite crear sitios web de alta interactividad, profesionalidad y eficiencia. Además de poder crear una estructura de soporte (framework) para la creación y administración de contenidos por parte de los participantes, principalmente en páginas web.

Con Joomla se pueden crear sitios web de noticias, sitios corporativos, portales comunitarios, e incluso también pueden crearse sistemas que funcionen en redes cerradas (Intranets) para gestionar información interna (comunicaciones, usuarios, etc.) de compañías o empresas de negocios. Con esto último puede advertirse que el ámbito de aplicación de este CMS no es exclusivo de Internet.

El funcionamiento de Joomla se lleva a cabo gracias a sus dos principales elementos:

- La base de datos MySQL: Es donde se guarda toda la información y la mayor parte de la configuración del sistema, de una forma ordenada y en distintas tablas, las cuales cada una de ellas almacena información específica y determinada.
- Los scripts PHP: son los que ejecutan las acciones de consulta y realizan modificaciones en la base de datos convirtiendo los datos en simples páginas web interpretables por los navegadores de Internet (Browsers) y perfectamente inteligibles para los usuarios navegantes y administradores.

Existen también otros tipos de archivos que realizan importantes tareas dentro de Joomla (archivos XML, scripts Javascript JS, CSS, etc.), pero el motor fundamental de todo CMS (y de este en particular) son los dos enunciados anteriores. [5]

Ventajas y Características de Joomla: [6]

Ventajas:

Con Joomla CMS sólo se debe ocupar de la información que se desee publicar, ya que el sistema gestionará todos los demás detalles técnicos y administrativos.

- Organización del sitio web: Está preparado para organizar eficientemente los contenidos de un sitio en secciones y categorías, lo que facilita la navegabilidad para

los usuarios y permite crear una estructura sólida, ordenada y sencilla para los administradores. Desde su panel administrador se puede crear, editar y borrar las secciones y categorías de un sitio de la manera más conveniente.

- **Publicación de Contenidos:** Se pueden crear páginas ilimitadas y editarlas desde un sencillo editor que permite formatear los textos con los estilos e imágenes deseados. Los contenidos son totalmente editables y modificables.
- **Escalabilidad e implementación de nuevas funcionalidades:** Ofrece la posibilidad de instalar, desinstalar y administrar componentes y módulos, que agregarán servicios de valor a los visitantes de los sitios web, por ejemplo: galerías de imágenes, foros, clasificados, etc.
- **Administración de usuarios:** Permite almacenar datos de usuarios registrados y también la posibilidad de enviar E-mails masivos a todos los usuarios. La administración de usuarios es jerárquica, y los distintos grupos de usuarios poseen diferentes niveles de permisos dentro de la gestión y administración del sitio.
- **Diseño y aspecto estético del sitio:** Es posible cambiar todo el aspecto del sitio web tan solo con un par de clics, gracias al sistema de templates que utiliza.
- **Navegación y menú:** Totalmente editables desde su panel administrador.
- **Administrador de Imágenes:** Posee una utilidad para subir imágenes al servidor y usarlas en todo el sitio.
- **Disposición de módulos modificables:** En un sitio creado con Joomla, la posición de módulos puede acomodarse como se prefiera.
- **Alimentación de Noticias:** Trae incorporado un sistema de sindicación de noticias por RSS/XMS de generación automática
- **Publicidad:** es posible hacer publicidad en el sitio usando el Administrador de Banners
- **Estadísticas de visitas:** con información de navegador y detalles de los documentos (páginas) más vistos.

Características de publicación de páginas web en Joomla:

- **Automatización en la publicación:** Sus páginas y documentos pueden programarse con fecha de publicación y fecha de caducidad. Es decir un documento puede

programarse para que se publique automáticamente al llegar una determinada fecha, y luego des-publicarse también de forma automática en otra fecha.

- Archivo e historial: Las páginas viejas o publicaciones que hayan perdido vigencia pueden enviarse a un "archivo" de almacenamiento, sin necesidad de tener que borrarlas. Esto permite también dar la posibilidad a los navegantes de consultar artículos viejos o documentos anteriores en un historial.
- Formatos de lectura: Cada documento es generado automáticamente por Joomla en formato PDF, en versión imprimible, y en XML.
- Envío por E-mail: Los usuarios del sitio podrán enviar automáticamente a un amigo por email cada documento publicado.
- Valoración de contenidos: Los visitantes del sitio podrán votar la calidad de lo publicado.

¿Por qué Joomla?

Para la realización del portal web se utiliza el CMS Joomla en su versión 1.0.12, siguiendo las características y ventajas mencionadas anteriormente. Esta versión surge a un mes de diferencia de la 1.5 dando solución a 140 *bugs* menores y a problemas de seguridad de bajo nivel. Además presenta los módulos y funcionalidades requeridas, por lo que se tomó la decisión dado el poco tiempo con que se contaba de no migrar a versiones superiores.

1.4.2 Macromedia Dreamweaver.

Dreamweaver es un editor HTML profesional para diseñar, codificar y desarrollar sitios, páginas y aplicaciones Web. Tanto si desea controlar manualmente el código HTML como si prefiere trabajar en un entorno de edición visual.

Las funciones de edición visual de *Dreamweaver* permiten crear páginas de forma rápida, sin escribir una sola línea de código. También incluye numerosas herramientas y funciones relacionadas con la codificación para crear el código manualmente. Además, ayuda a crear aplicaciones Web dinámicas basadas en bases de datos empleando lenguajes de servidor como ASP, ASP.NET, *ColdFusion Markup Language* (CFML), JSP y PHP.

1.5 Lenguajes de Programación utilizados en el proceso de desarrollo.

En los lenguajes de programación web se pueden encontrar los del lado del servidor y los del lado del cliente. Se caracterizan por desarrollar la lógica de negocio dentro del Servidor, además de ser los encargados del acceso a Bases de Datos y tratamiento de la

información. En el lado del cliente se pueden ver JavaScript, que es el encargado de aportar dinamismo a la aplicación en los navegadores.

1.5.1 PHP.

Es el acrónimo recursivo de Hypertext Preprocessor. Es un lenguaje de programación de servicios web del lado del servidor gratuito e independiente de plataforma, rápido, con una gran librería de funciones y mucha documentación. Es también un lenguaje interpretado y embebido en el HTML. [7]

PHP es un lenguaje de programación potente que, junto con HTML, permite crear sitios Web dinámicos. Se instala en el servidor y funciona con versiones de Apache, Microsoft IIS, Netscape Enterprise Server y otros. Permite la conexión a numerosas bases de datos, incluyendo MySQL, Oracle, ODBC, etc. y puede ser ejecutado en la mayoría de los sistemas operativos (Windows, Mac OS, Linux, Unix. [8]

De manera general el PHP:

- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Es rápido. Su integración con la base de datos MySQL y el servidor Apache, le permite presentarse como una de las alternativas más atractivas del mercado.
- Cuenta con gran magnitud de documentación, pues tiene una de las comunidades más grandes en Internet.
- Posee una potente variedad de extensiones para el acceso a la mayoría de los sistemas de gestión de bases de datos, por lo que una migración a otro sistema de gestión es menos costosa que en otras plataformas.

¿Por qué PHP?

Para la elaboración del portal se utilizó PHP en su versión 5.0, por poseer buena compatibilidad con el CMS Joomla, además por presentar las características antes mencionadas y otra fundamental en esta versión y es que el soporte para la Programación Orientada a Objetos fue mejorado, pues en versiones anteriores era rudimentario.

1.5.2 JavaScript.

Es un lenguaje de programación del lado del cliente, porque es el navegador el que soporta la carga de procesamiento. Gracias a su compatibilidad con la mayoría de los navegadores modernos, es un lenguaje de programación del lado del cliente más utilizado. JavaScript es

un lenguaje de programación que se utiliza para crear programas pequeños encargados de realizar acciones dentro del ámbito de una página web. Es un lenguaje de programación sencillo y pensado para hacer las cosas con rapidez.

En JavaScript se pueden realizar acciones típicas, como son: los efectos especiales sobre páginas web para crear contenidos dinámicos y elementos de la página que tengan movimiento, cambien de color o cualquier otro dinamismo. Permitir que se ejecuten instrucciones como respuesta a las acciones del usuario, con lo que se crean páginas interactivas con programas como calculadoras, agendas, o tablas de cálculo. [9]

1.6 Lenguajes de marcas.

Un lenguaje de marcas o lenguaje de marcado es una forma de llevar a código un documento que, unido al texto, agrega etiquetas o marcas que contienen información adicional acerca de la estructura que presenta el texto o la presentación.

1.6.1 XML.

XML es un acrónimo de eXtensible Markup Language. Es un metalenguaje extensible de etiquetas, es decir, lenguajes para definir lenguajes de estructuración de texto basados en marcas explícitas o *tags*, que son entendibles por una persona y que pueden ser interpretadas por un computador. Su aplicación no fue creada solo para Internet, sino que se propone como un estándar para el intercambio de información estructurada entre diferentes plataformas. Se puede usar en bases de datos, editores de texto, hojas de cálculo y casi cualquier cosa imaginable. [10]

Características de XML:

El analizador es un componente estándar, no es necesario crear un analizador específico para cada lenguaje esto posibilita el empleo de uno de los tantos disponibles. De esta manera se evitan *bugs* y se acelera el desarrollo de la aplicación. Si un tercero decide usar un documento creado en XML, es sencillo entender su estructura y procesarlo. Mejora la compatibilidad entre aplicaciones.

Aunque hoy día XML aún no está tan extendido como HTML, su uso futuro en la Web mejorará la eficiencia de las búsquedas, al proporcionar cada documento XML metadatos sobre sí mismo.

Permite proporcionar diferentes vistas sobre los datos (HTML, PDF, etc.), dependiendo de quién sea el cliente.

Facilita la integración desde fuentes de datos heterogéneas, por ejemplo, páginas Web y distintas bases de datos.

Los documentos tienen una estructura que los hace legibles e inteligibles no sólo para los ordenadores, si no también para los humanos. [11]

Ventajas de XML:

- Las aplicaciones se pueden generar rápidamente y su mantenimiento es más sencillo.
- Separa los datos de la presentación y del proceso, lo que permite mostrar y procesar los datos al gusto deseado con sólo aplicar distintas hojas de estilo y aplicaciones.
- La información es más accesible y reutilizable, por la flexibilidad de las etiquetas de XML que permiten su utilización sin tener que amoldarse a reglas específicas de un fabricante.

1.6.2 HTML.

Este es un lenguaje para la composición de documentos y especificación de ligas de hipertexto que define una sintaxis adaptada a la Web. Está basado en el estándar ISO 8879 que corresponde al SGML (Standard Generalized Markup Language : Lenguaje de Marcas Estándar Generalizado).

El elemento central de éstos lenguajes es la marca, ésta indica la manera en que una palabra, párrafo o grupo de líneas será tratado. Normalmente los documentos se escriben en ASCII con herramientas simples y son interpretados por un visualizador, en el caso de la Web el visualizador es el Navegador Web. [12]

1.7 Librería ExtJS.

En la actualidad existen diversas librerías que facilitan las herramientas para la elaboración de aplicaciones web obteniendo gráficos de gran calidad, un ejemplo de esta librería es ExtJS; esta contiene una colección de componentes para el diseño de interfaces, ventanas, pestañas, menús, tablas, etc.

A principios del 2007 se creó una compañía para comercializar y dar soporte al ExtJS, dicha compañía proporciona los servicios de consultoría necesarios para ayudar a los clientes en el aprovechamiento máximo de las ventajas de ExtJS, por tal motivo es considerado un Framework independiente.

Además es importante señalar que la ExtJS es un framework JavaScript del lado del cliente para el desarrollo de aplicaciones web. Tiene un sistema dual de licencia: Open Source y Comercial, esta última es obligatoria si se desea obtener soporte. Este framework puede correr en cualquier plataforma que pueda procesar POST y devolver datos estructurados (PHP, Java, .NET y algunas otras). [13]

Brinda soporte para:

- Construir interfaces gráficas complejas y dinámicas.
- Comunicar datos de forma asíncrona con el servidor.
- Manejar datos de distinta índole de una manera simple.

1.8 Servidor Web Apache.

Es el servidor Web más utilizado en el mundo. Su coste gratuito, fiabilidad y extensibilidad le convierten en una herramienta potente y fiable.

Dentro de sus puntos fuertes se encuentran:

- Tiene interfaz con todos los sistemas de autenticación.
- Facilita la integración como "*plug-ins*" (aditamento para agregar a un equipo) de los lenguajes de programación de páginas Web dinámicas más comunes.
- Provee interfaz a todas las bases de datos.
- Posee Virtual Host.

Apache fue diseñado para proveer un alto grado de calidad y fortaleza para las implementaciones que utilizan el protocolo HTTP. Está ligado a la plataforma (Linux, Windows, UNIX) sobre la cual los individuos o instituciones pueden construir sistemas confiables con fines experimentales o para resolver un problema específico de la organización.

Apache es un software libre, porque sus desarrolladores defienden la teoría de que las transmisiones usando la red deben estar en las manos de todos, y que las compañías de software deben ganar dinero solamente ofertando servicios de valor añadido tales como módulos especializados, soportes y otros, y no adueñándose de un protocolo. Así, el proyecto de crear una implementación robusta con referencia absolutamente libre para quien lo quiera usar es un buen paso para evitar la propiedad sobre los protocolos.

Tiene capacidad para servir páginas tanto de contenido estático, para lo que serviría sencillamente un viejo ordenador 486, como de contenido dinámico a través de otras herramientas soportadas que facilitan la actualización de los contenidos mediante bases de datos, ficheros u otras fuentes de información. [14]

1.9 Sistema Gestor de Base de Datos.

Los Sistema Gestor de Base de Datos (SGBD) son un tipo de software que es una interfaz entre la aplicación que la utiliza y la base de datos. Tiene como objetivo principal facilitar a los usuarios las herramientas que permiten manipular los datos. [15]

1.9.1 MySQL.

MySQL Database Server es la base de datos de código fuente abierto más usada del mundo. Su ingeniosa arquitectura lo hace extremadamente rápido y fácil de personalizar. La extensiva reutilización del código dentro del software y una aproximación mínima para producir características funcionalmente ricas, ha dado lugar a un sistema de administración de la base de datos incomparable en velocidad, compactación, estabilidad y facilidad de despliegue. La exclusiva separación del core server del manejador de tablas, permite funcionar a MySQL bajo control estricto de transacciones o con acceso a disco no transaccional rápido. [16]

Características:

Interioridades y portabilidad:

- Probado con un amplio rango de compiladores diferentes.
- Funciona en diferentes plataformas.
- Relativamente sencillo de añadir otro sistema de almacenamiento. Esto es útil si desea añadir una interfaz SQL para una base de datos propia.
- Joins muy rápidos usando un multi-join de un paso optimizado.
- El servidor está disponible como un programa separado para usar en un entorno de red cliente/servidor. También está disponible como biblioteca y puede ser incrustado en aplicaciones autónomas. Dichas aplicaciones pueden usarse por sí mismas o en entornos donde no hay red disponible.

Seguridad:

Es un sistema flexible y seguro con privilegios y contraseñas, permite verificación basada en el host. Como todo el tráfico de contraseña al conectarse con un servidor está encriptado se dice que las mismas son seguras.

Escalabilidad y límites:

- Soporte a grandes bases de datos. Se usa MySQL Server con bases de datos que contienen 50 millones de registros.
- Se permiten hasta 64 índices por tabla. Cada índice puede consistir desde 1 hasta 16 columnas o partes de columnas. Un índice puede usar prefijos de una columna para los tipos de columna CHAR, VARCHAR, BLOB, o TEXT.

Conectividad:

- Los clientes pueden conectar con el servidor MySQL usando sockets TCP/IP en cualquier plataforma. En sistemas Windows de la familia NT (NT, 2000, XP, o 2003), los clientes pueden usar conductos para la conexión. En sistemas Unix, los clientes pueden conectar usando ficheros socket Unix.
- La interfaz para el conector ODBC (MyODBC) proporciona a MySQL soporte para programas clientes que usen conexiones ODBC (Open Database Connectivity). Los clientes pueden ejecutarse en Windows o Unix. El código fuente de MyODBC está disponible. Todas las funciones para ODBC 2.5 están soportadas, así como muchas otras.
- La interfaz para el conector J MySQL proporciona soporte para clientes Java que usen conexiones JDBC. Estos clientes pueden ejecutarse en Windows o Unix. El código fuente para el conector J está disponible.

Localización:

- El servidor puede proporcionar mensajes de error a los clientes en muchos idiomas.
- Todos los datos se guardan en el conjunto de caracteres elegido. Todas las comparaciones para columnas normales de cadenas de caracteres son *case-insensitive*.

Clientes y herramientas:

- MySQL server tiene soporte para comandos SQL para chequear, optimizar, y reparar tablas. Estos comandos están disponibles a través de la línea de comandos y el cliente mysqlcheck.
- Todos los programas MySQL pueden invocarse con las opciones --help o -? para obtener asistencia en línea.

¿Por qué MySQL?

Se decidió usar para la elaboración del portal como Sistema Gestor de Base de Datos: MySQL en su versión 5.1, por ser uno de los SGBD que utiliza Joomla, además por presentar las características antes mencionadas.

1.10 Lenguaje Unificado de Modelado.

UML (Unified Modeling Language) o Lenguaje de Modelación Unificado es un lenguaje gráfico para especificar, construir, visualizar y documentar las partes o artefactos (información que se utiliza o produce mediante un proceso de software).

Es expresivo, claro y uniforme. Además se utiliza para modelar sistemas orientados a objetos e incluye un conjunto de diagramas y notaciones estándar y su significado. UML cuenta con varios tipos de diagramas para definir sistemas de software y hardware, detallar artefactos en los sistemas y documentar la programación. Puede ser utilizado en variadas metodologías de software aunque se utilizará específicamente en el Proceso Unificado de Desarrollo (RUP) en el actual proyecto. [17]

1.11 Metodología de desarrollo.

En un proyecto de desarrollo de software se define “quién” debe hacer “qué”, “cuándo” y “cómo” debe hacerlo. La metodología es un proceso, y se tiene considerado que no existe una metodología de software universal. Lo que hace que las características del proyecto sean las que lo configuren.

El Proceso Unificado está basado en componentes, utiliza UML para preparar todos los esquemas de un sistema de software. Por lo que, es una parte esencial de Proceso Unificado de Desarrollo (RUP), sus desarrollos fueron paralelos. No obstante los verdaderos aspectos definitorios del proceso unificado se resumen en tres características fundamentales: dirigido por casos de uso, centrado en la arquitectura, e iterativo e incremental.

RUP es un proceso de ingeniería de software que mejora la productividad del equipo de trabajo y entrega las mejores prácticas del software a todos los miembros del mismo. Proporciona una guía específica en áreas tales como la de Modelado de Negocios, Arquitecturas Web, Pruebas y Calidad.

Contiene patrones que permiten a los gestores de proyectos añadir o quitar rápidamente piezas que resuelvan problemas comunes y enfocarlas en el proyecto específico. Permite manejar contenidos en diferentes dominios como el modelado de bases de datos o control de requisitos.

Incluye artefactos (que son los productos tangibles del proceso como por ejemplo, el modelo de casos de uso, el código fuente, etc.) y roles (papel que desempeña una persona en un determinado momento, una persona puede desempeñar distintos roles a lo largo del proceso).

Los casos de uso (procesos), o sea, lo que los usuarios futuros necesitan o desean, guían el proceso de desarrollo. Los modelos que se obtienen como resultado de los diferentes flujos de trabajo representan la realización de un Caso de Uso (CU).

La arquitectura muestra la visión común del sistema, describe los elementos del modelo que son más importantes para su construcción. RUP se desarrolla mediante iteraciones, comenzando por los CU relevantes desde el punto de vista de la arquitectura.

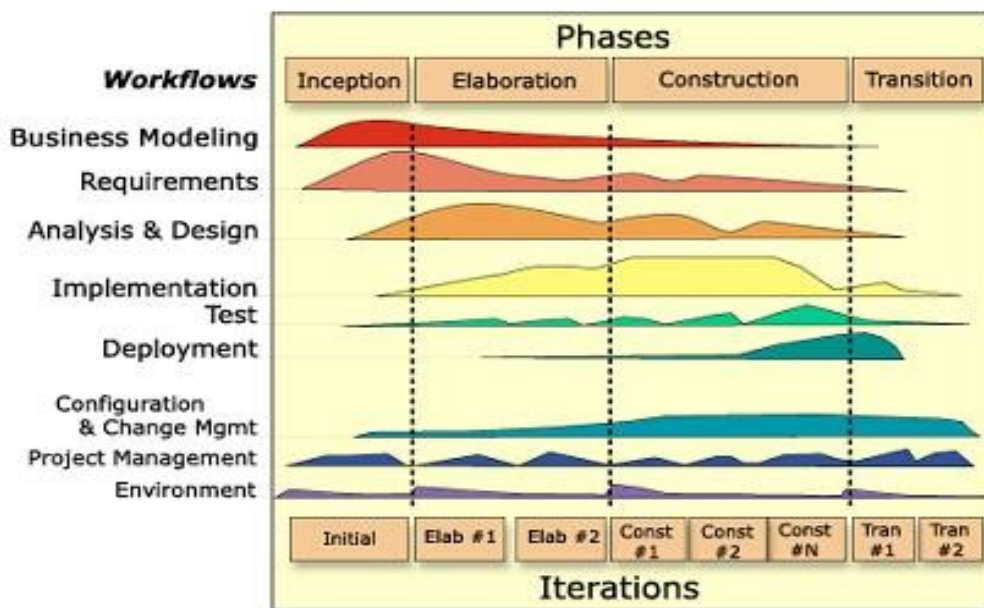


Figura 1. 1 Representación de la Metodología RUP

RUP propone que cada fase del proyecto se desarrolle en iteraciones. Una iteración involucra actividades de todos los flujos de trabajo aunque se enfoca en unos más que en otros. [18]

1.11.1 Herramientas Case.

Rational Rose Enterprise es el producto más completo de la familia Rational Rose. Todos los productos Rational Rose incluyen soporte UML. Su principal ventaja es que permite generar código en distintos lenguajes de programación (Ada, ANSI C++, C++, CORBA, Java/J2EE, Visual C++ y Visual Basic) a partir del diseño UML. Como todos los demás productos Rational Rose, proporciona un lenguaje común de modelado para el equipo que facilita la creación de software de calidad más rápidamente.

Rational Rose Enterprise Edition es una herramienta que esta comprendida dentro del grupo de herramientas más técnicas, pues es la encargada de llevar a cabo la automatización de los sistemas para la posterior generación de código, y para labores de ingeniería inversa.[19]

1.12 Herramientas de planificación y gestión de proyectos.

Para la elaboración de todo proyecto es preciso tener una planificación desde que se inicia hasta que se termina las tareas que serán realizadas. No es menos cierto que existen fases que no necesitan de la planificación, pero si deben ser tomadas en cuenta. Lo que da soporte a estas aplicaciones se denominan: Herramientas de planificación y gestión de proyectos. Estas son las encargadas de la gestión del tiempo, administrar recursos y administrar el coste definido.

1.12.1 DotProject.

DotProject es una herramienta de software libre para la gestión de proyectos. La que dispone de foros de discusión, calendario, tareas, recursos, contactos, tickets y repositorio de ficheros. Además permite la creación de planes de proyectos mediante la descomposición en tareas y presenta una vista de eventos y tareas en un calendario. Otro elemento que ofrece el *DotProject* es la visualización de las estadísticas e informes del proyecto y un sistema de administración y configuración del portal.

Dicha herramienta posee una integrabilidad Web debido a que toda su interfaz es web.

El *DotProject* tiene como ventajas:

- Su instalación es sencilla y bien documentada.

- Pueden calcular fechas de inicio y fin automáticamente.
- Pueden desplazar las tareas dependientes de una tarea a la vez.
- Se visualizan todas las tareas a realizar y eventos registrados del día en un calendario compartido, para mejorar el trabajo colaborativo.
- Se aprecian el estado (tareas completas y la que sufren desviaciones) del proyecto y las estadísticas (porcentaje de avance de las tareas y tiempo incurrido por los usuarios) sobre el mismo.
- Se visualizan las tareas mediante el diagrama de Gantt. [20]

1.12.2 Gantt Project.

Gantt Project al igual que el *DotProject* es una herramienta de software libre que se utiliza para la planificación de proyecto. Con la diferencia que es un editor gráfico del diagrama de Gantt.

Esta herramienta permite importar y exportar archivos de *MS Project*. Posibilita la carga de recursos de otro proyecto al actual. Además exporta el proyecto a páginas HTML y publica un informe en formato PDF. Se puede trabajar con proyectos almacenados en servidores web y personalizar las funciones de los recursos en el diálogo de configuración. Otra de las ofertas que brinda es enviar correos electrónicos a los recursos y resalta el camino crítico de su proyecto. En caso que se modifiquen las tareas, el diagrama de Gantt se modifica automáticamente y viceversa.

Gantt Project presenta las siguientes ventajas:

- Interfaz estándar de Windows con un número razonable de botones y opciones.
- Fácil de usar.
- Disponible en todas las plataformas
- Exporta los diagramas a varios formatos que pueden ser leídos por otros programas. [21]

1.12.3 XPlanner.

XPlanner es una herramienta de software libre para la planificación y seguimiento de proyectos, específicamente para equipos que trabajan con la metodología XP.

Oferta tarjetas virtuales para la definición de historias de usuario y tareas. Soporte para grabar y registrar tareas, historias de usuario e iteraciones. Además de informes y gráficas de velocidades de iteración, distribución de tipos de tareas, horas estimadas aún no consumidas (scrum burn-down). Otros elementos que proporciona es una vista para medir la precisión de las estimaciones y genera informes multiformato (*MS Excel, MSProject, PDF, XML*). Brinda además una autenticación integrada y extensible; e interfaces SOAP para integrar la funcionalidad de *XPlanner* o extenderla.

XPlanner presenta las siguientes ventajas:

- Es sencillo de usar y ofrece una herramienta potente para un seguimiento de las estimaciones y planificaciones seguidas según una metodología XP.
- Las tarjetas virtuales son una herramienta valiosa para la mejora continua y evitar pérdidas de información.
- El seguimiento pormenorizado de las estimaciones permite la mejora continua de las mismas.
- La interfaz web facilita el trabajo en grupo distribuido geográficamente. [22]

1.12.4 Valoración de la herramienta de planificación y gestión de proyectos.

La herramienta escogida es *DotProject*, ya que es una herramienta de trabajo colaborativo que sirve para relacionar empresas, proyectos, personas y tareas dentro de un ambiente configurable y abierto. Es una aplicación que entre otras opciones permite:

- Planificación: Organizar el proyecto en función de hitos, tareas y sub-tareas, con asignación y control de tiempos y recursos materiales y humanos.
- Gestión Documental: Servirá para almacenar y mantener los documentos obtenidos o generados durante el desarrollo del proyecto y acceder a ellos cómodamente. Cada hito, tarea o sub-tareas puede implicar la obtención o generación de documentación (actas de reuniones, documentos de diseño, etc.).

1.13 Herramientas de gestión de la configuración.

La gestión de la configuración de software es la que controla las evoluciones que van tomando el software y los documentos que lo componen. Se encarga de identificar y definir los elementos del sistema. Además registra y reporta el estado de los elementos y las solicitudes de cambio y verifica que los elementos estén completos y sean los correctos.

1.13.1 Concurrent Versions System (CVS).

CVS es un sistema open-source de control de versiones que permite guardar y mantener accesibles las diferentes versiones que se van generando de un mismo fichero de código.

Al contrario de los sistemas de control de versiones tradicionales, CVS permite que varios desarrolladores se conecten y trabajen simultáneamente sobre un mismo fichero, proporcionando la resolución automática de los conflictos que se puedan producir al entregar las modificaciones, debido a esto es muy útil en los proyectos open-source desarrollados a través de Internet pues permite a cada desarrollador trabajar de forma independiente y sin preocuparse de las modificaciones que otros desarrolladores puedan estar haciendo.

Utiliza la arquitectura cliente-servidor, donde el servidor guarda las versiones actuales y el cliente accede al servidor y saca las copias completas del proyecto. Otras funcionalidades que brinda dicha herramienta son: comparar disímiles versiones de archivos, solicitar la historia completa de cambios, ver el estado del proyecto en una fecha determinada o en un número de revisión determinado, etc.

Se dispone de un plugin para Eclipse que permite el uso de la herramienta desde dentro de dicho entorno, pero no es posible la navegación web, aunque si permite recuperar versiones anteriores del código de un proyecto. [23]

1.13.2 Subversión (SVN).

Es una herramienta que controla las versiones de los elementos (desde el código fuente hasta los documentos pasando por ejecutables, imágenes, etc.) que forman parte de un producto. La herramienta permite:

- Mantener la historia de archivos y directorios a través de copias y renombrados.
- Creación de ramas y etiquetas de forma eficiente.
- Gestión eficiente de archivos binarios.

Esta herramienta brinda funcionalidades como: confirmación de cambios de versiones automáticamente y mover o copiar directorios completos sin cambiar su historial de versiones. Además las diferencias se envían tanto del cliente al servidor como del servidor al cliente. La salida está diseñada para que sea fácilmente comprensible por las personas o que se pueda leer automáticamente por otros programas.

Se puede acceder al repositorio subversión vía web gracias a un módulo que se instala en Apache. Además existen otros scripts/clientes para acceder al repositorio SVN.

Tiene como ventajas que: el tiempo en ejecutar una operación es proporcional al tamaño de cambio y no al tamaño de los datos. La administración del servidor es sencilla. Permite crear ramas de forma sencilla y eficiente. Además admite espacios en los nombres de fichero. Está reemplazando a CVS en el desarrollo de proyectos *Open Source*. [24]

1.13.3 Valoración de la herramienta de gestión de la configuración.

En un proyecto productivo a lo largo de su desarrollo se van realizando una serie de revisiones en los códigos que se utilizan, para permitir el desarrollo presente y mantener la historia del código fuente y parte de la documentación producida que deben ser modificados por múltiples personas, por lo tanto resulta casi imprescindible disponer de un Sistema de Control de Versiones, en este caso el Subversión, que permita mantener la historia de los ficheros generados y que más de una persona trabaje concurrentemente sobre el mismo código. Por tales motivos se considera que es la mejor herramienta para controlar las versiones en el trabajo a realizar.

Conclusiones.

En este capítulo se realizó una profunda exposición de las herramientas, lenguaje de modelado y metodología de desarrollo que serán usadas en el desarrollo de la aplicación, teniendo siempre presente las ventajas y desventajas. Finalmente, se desarrollará la aplicación usando la metodología RUP; UML como lenguaje para describir todo el proceso de desarrollo, como herramienta de modelado el *Rational Rose Enterprise Edition 2003*, y para la programación se utilizará el lenguaje de PHP en su versión 5.0. Además se usará como CMS Joomla el cual funciona con mejor efectividad utilizando como Gestor de Base de Datos MySQL y el lenguaje de programación mencionado anteriormente.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA.

Introducción.

En el presente capítulo se hace una descripción de la propuesta de solución en el Portal Web de APS. Como no tiene los procesos del negocio con la estructuración correcta, es decir no están definidos, para poder tener un correcto entendimiento del contexto que se desarrolla en el sistema, se decide realizar un Modelo de Dominio, donde se muestra los conceptos y sus relaciones. Además se tienen los requisitos funcionales y no funcionales; seguido a estos los Diagramas de Casos de Uso del Sistema divididos por paquetes y con su descripción textual.

2.1 Modelo de Dominio.

Para la realización de la aplicación se decidió elaborar un Modelo de Dominio, el cual hace una representación de un conjunto de conceptos que se relacionan entre sí, pues después hacer un análisis profundo se ultimó que no es posible hacer el Diagrama de Casos de Uso del Negocio, debido a que no se pueden identificar claramente los procesos del negocio. Se aprecian solamente los conceptos que se relacionan en la aplicación.

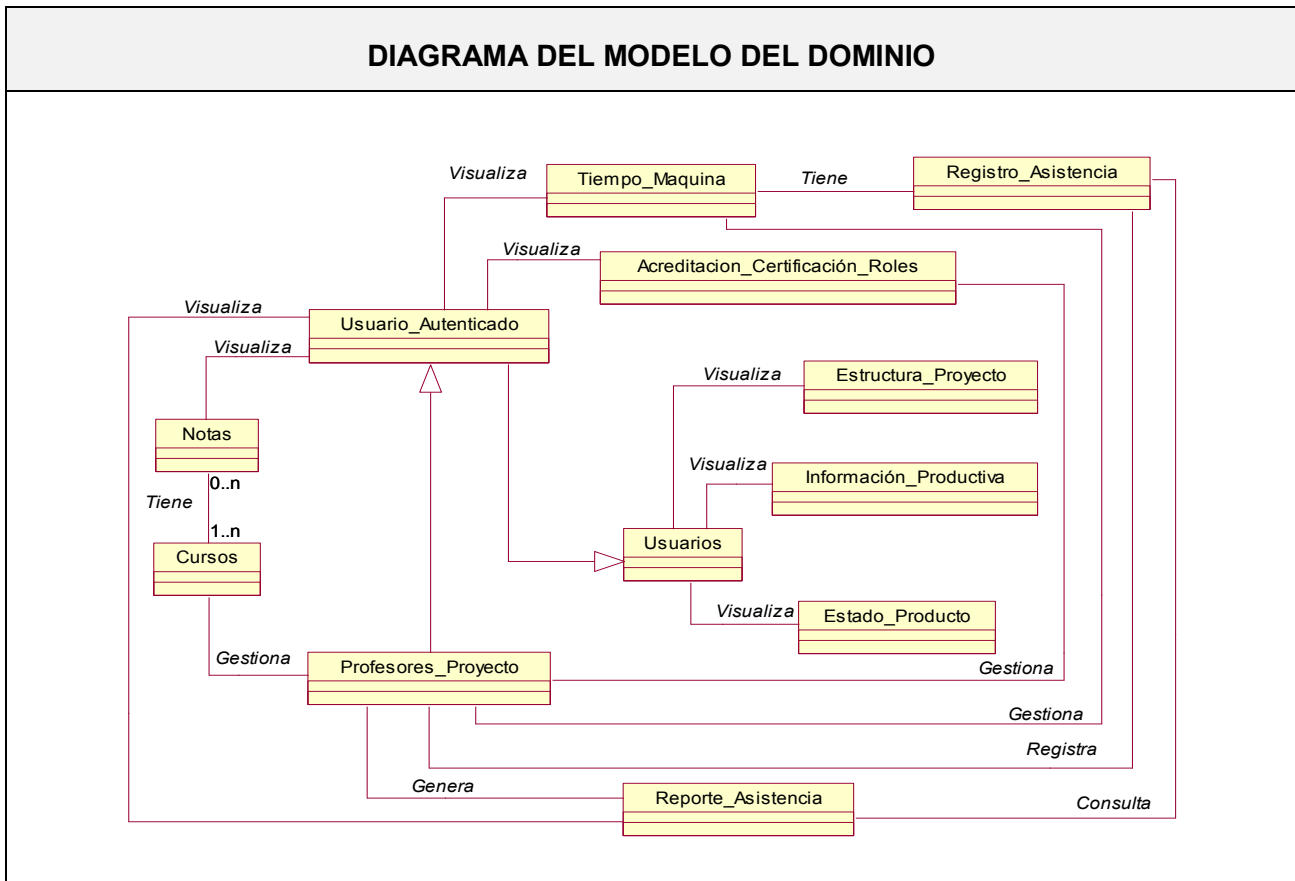


Figura 2. 1 Modelo de Dominio.

2.1.1 Conceptos de las clases del modelo del dominio.

Usuarios: Persona que accede al portal sin haberse autenticado aún para acceder a información general del proyecto.

Usuario Autenticado: Persona que interactúa directamente con la aplicación después de autenticarse.

Profesores del Proyecto: Son los profesores del proyectos, los que se encuentran registrados como usuarios locales del sistema con privilegios de administración.

Notas: Son las calificaciones que va a tener un estudiante del proyecto en un curso determinado.

Cursos: Se refiere a los cursos optativos que los profesores les van a poner a los estudiantes vinculados a este proyecto productivo.

Tiempo Máquina: Es la distribución de tiempos de máquinas a todos los estudiantes y profesores del proyecto, donde se le asigna una máquina a cada uno de los antes mencionados.

Registro de Asistencia: En éste se controla la asistencia de los estudiantes, por puesto de trabajo.

Reporte Asistencia: Según los registros que ya se tienen, se puede hacer los reportes de asistencia por fecha, ya sea para verificar las computadoras vacías, estudiantes ausentes o presentes.

Acreditación Certificación de Roles: Aquí se puede ver los resultados de los estudiantes que han acreditado o certificado algún rol, y se pondrán los que van realizando esta actividad.

Estado Producto: Se refiere al estado de liberación en que se encuentra el producto.

Estructura Proyecto: Aquí se expone toda la estructura del proyecto incluyendo los módulos que lo integran.

Información Productiva: Está integrado por todas las informaciones, ya sean del proyecto o la facultad y que tengan que ver con todo lo referente a la producción.

2.2 Especificación de los Requerimientos.

Todas las ideas que los clientes, usuarios y miembros del equipo de proyecto tengan acerca de lo que debe hacer el sistema, deben ser analizadas como candidatas a requisitos.

2.2.1 Requerimientos Funcionales.

Los requerimientos funcionales son capacidades o condiciones que el sistema debe cumplir sin alterar la funcionalidad del producto.

RF 1 Gestionar tiempo de máquina.

1.1 Insertar un nuevo tiempo de máquina.

1.2 Eliminar tiempo de máquina.

1.3 Editar tiempo de máquina.

RF 2 Controlar asistencia de estudiantes por el tiempo de máquina.

RF 3 Gestionar Curso de los estudiantes.

3.1 Insertar curso.

3.2 Eliminar curso.

3.3 Editar curso.

RF 4 Gestionar Roles Acreditados o Certificados.

4.1 Insertar roles.

4.2 Eliminar roles.

4.3 Editar roles.

RF 5 Generar Reportes de asistencia dado una fecha.

5.1 Generar Reportes de estudiantes presentes.

5.2 Generar Reportes de estudiantes ausentes.

5.3 Generar Reportes de computadoras vacías.

RF 6 Listar tiempo de máquina.

RF 7 Listar datos de estudiantes.

RF 8 Listar roles acreditados o certificados por estudiantes.

RF 9 Listar cursos y notas de los estudiantes.

RF 10 Mostrar diariamente los integrantes que cumplen años.

RF 11 Mostrar versiones y sistemas desplegados.

RF 12 Mostrar la estructura del proyecto.

RF 13 Mostrar información productiva.

2.3.2 Requerimientos no Funcionales.

Usabilidad.

Estos requerimientos describen los niveles apropiados de usabilidad, dados los usuarios finales del producto y para ello debe revisarse las especificaciones de los perfiles de usuarios y las clasificaciones de sus niveles de experiencia.

RNF 1

Cada usuario que entre a la aplicación tendrá acceso a la información que le corresponda, y en caso de que se autentique, tendrá el permiso de editor o visualizador, en dependencia de que sea profesor o estudiante.

Seguridad.

Son los que provocan mayor cantidad de riesgo si no son tratados correctamente.

Confidencialidad.

Toda la información manejada en el sistema está protegida de acceso no autorizado.

RNF 2

Debe disponer de un mecanismo de seguridad basado en el modelo de Autenticación, Autorización y Auditoría (AAA).

RNF 3

El usuario se autentica con los datos del dominio UCI.

RNF 4

Si al autenticarse un usuario no lo hace por el dominio, entonces la aplicación debe mostrar un error de acceso.

Integridad.

El sistema maneja la información que será protegida contra la corrupción y estados inconsistentes, de la misma forma será considerada igual a la fuente o autoridad de los datos. Además pueden incluir mecanismos de chequeo de integridad.

RNF 5

Prevenir posibles fallos y recuperarse ante la presencia de uno de ellos.

RNF 6

Para facilitar la recuperación del sistema en caso de que ocurra algún fallo deberá existir una estrategia de replicación que permita balancear la carga de acceso entre múltiples servidores aumentando los tiempos de respuesta, de manera transparente para el usuario final.

RNF 7

Se permitirá la creación de copias de respaldo que puedan restaurar el sistema en caso de fallo crítico o pérdida total de la información.

Disponibilidad.

En este requisito a los usuarios autorizados se les garantizará el acceso a la información, y los dispositivos o mecanismos utilizados para lograr la seguridad no ocultarán o retrasarán a los usuarios para obtener los datos deseados en un momento dado.

RNF 8

La seguridad del sistema no molestará en ningún momento la rapidez con que se dará la respuesta al usuario, y se debe minimizar y reducir el tiempo de respuesta, así como optimizar el código.

Eficiencia.

Esto especifica los requisitos relacionados con la carga que se espera que tenga que soportar el sistema. Por ejemplo, el tiempo de respuesta por transacción, el número esperado de usuarios simultáneamente conectados, número de transacciones por segundo que deberá soportar el sistema, etc. Todos estos requisitos deben ser medibles. Debe indicarse en porcentaje (%) relacionado con su tiempo.

RNF 9

El sistema debe soportar un tiempo de respuesta menor de 30 segundos.

RNF 10

El sistema debe soportar conexión simultánea de hasta 7 000 usuarios.

RNF 11

El sistema debe soportar respuestas al usuario que no deben exceder los 50 Kbyte.

Soporte.

Este requisito abarca todas las acciones que se toman una vez terminado el desarrollo del software, con motivos de lograr su mejoramiento progresivo y mejoras en el tiempo. Pueden incluir: Pruebas, Extensibilidad, Adaptabilidad, Mantenimiento, Compatibilidad, Configuración, Servicios, Instalación e Internacionalización.

RNF 12

Las personas que trabajan con el módulo deben contar con el nivel técnico requerido.

Restricciones de diseño.

Son aquellos que especifican o restringen la codificación o construcción de un sistema, son restricciones que han sido ordenadas y deben ser cumplidas estrictamente. Ejemplos de ellas son: estándares requeridos, requisitos de proceso de software, lenguajes de programación a ser usados para la implementación, uso obligatorio de ciertas herramientas de desarrollo, restricciones en la arquitectura y el diseño, bibliotecas de clases, etc.

RNF 13

La lógica de presentación será una capa independiente de la lógica de negocio, centrando su función en la interfaz de usuario y validaciones simples de los datos de entrada.

RNF 14

Validar el proceso de la captación de datos para evitar entradas inadecuadas.

RNF 15

Las herramientas de desarrollo a utilizar deben garantizar la calidad de todo el ciclo de desarrollo de la aplicación.

Requisitos para la documentación de usuarios en línea y ayuda del sistema.

Describe los requisitos para la documentación necesaria para la utilización, manipulación y ayuda del sistema así como para el entendimiento del mismo.

RNF 16

Debe disponer de una documentación del sistema, argumentada con alguna metodología de desarrollo.

Interfaz Externa.

Son aquellos que describen la apariencia del producto. Es importante destacar que no se trata del diseño de la interfaz en detalle, sino que especifican cómo se pretende que sea la interfaz externa del producto.

RNF 17

Todas las funciones se realizan desde cualquier página principal, pudiendo regresar a ella.

RNF 18

A todas las funcionalidades del sistema se accede desde cualquier página.

Interfaces Interna.

Son aquellas que describen la apariencia del producto internamente, como deben ser sus componentes, la relación entre ellos, así como el funcionamiento interno del software.

RNF 19

Los componentes desarrollados en el sistema deben seguir el principio de alta cohesión y bajo acoplamiento.

Hardware.

Son los que especifican las características lógicas para cada interfaz entre el producto y los componentes de hardware del sistema. Incluyendo la estructura lógica, direcciones físicas, el comportamiento esperado, etc.

RNF 20

Impresoras locales o de red para imprimir los reportes realizados.

Software.

Describen los programas para que el sistema funcione correctamente. Indican si es necesario integrar el producto con otro software o si hay que asociar otros programas al mismo.

RNF 21

Los usuarios tendrán acceso al portal web a través de cualquier navegador, recomendado Mozilla 1.5, Internet Explorer 5.0 o superior.

Portabilidad.

Son aquellos que especifican los atributos que debe presentar el software para facilitar su traslado a otras plataformas o entornos de desarrollo.

RNF 22

Permitir que el sistema se ejecute sobre el Sistema Operativo Windows 98 o superior, o Linux.

2.3 Modelo de Casos de Uso del Sistema.

Con el Modelo de Casos de Uso del Sistema los desarrolladores del software y clientes van a entender con mejor facilidad los requisitos a los que debe dar cumplimiento el sistema. Además teniendo un sistema bien modelado es más sencillo realizar el análisis, diseño e implementación. En este Modelo se puede apreciar los casos de uso, actores y la relación entre ellos.

2.3.1 Definición de Actores del Sistema.

El Modelo de CUS describe lo que hace para cada tipo de usuario. Los actores pueden ser lo mismo una persona que otro sistema que interactúe con el sistema en desarrollo, seguidamente se mencionan los que están presentes en aplicación que se está creando.

Actor	Descripción
Usuarios (Usuario Autenticado)	Es una generalización de Usuario Autenticado donde incluye además a todo el personal de la UCI y se encarga de visualizar la información general del proyecto.
Usuario Autenticado (Profesor Proyecto)	Es una generalización de Profesores Proyecto, que además incluye a los estudiantes y profesores de la UCI, se encarga de listar y recibir la información que brinda el sistema una vez que se haya autenticado.

Profesor Proyecto	Es el encargado de gestionar y registrar toda la información que brinda el sistema, ya que puede insertar, modificar y listar la información en la base de datos, por ser usuarios locales del sistema con privilegios de administración.
--------------------------	---

Tabla 2. 1 Descripción de los actores.

2.3.1.1 Relación de los Actores en el Sistema.

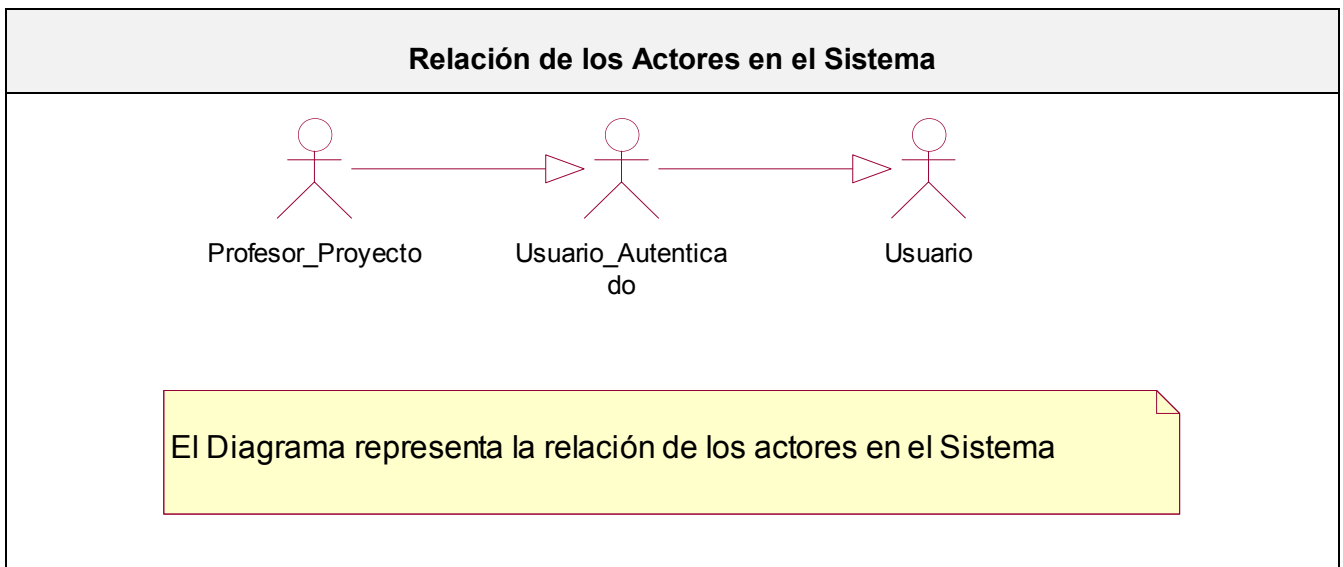


Figura 2. 2 Relación de los actores en el Sistema.

2.3.2 Diagrama de Casos de Uso del Sistema.

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

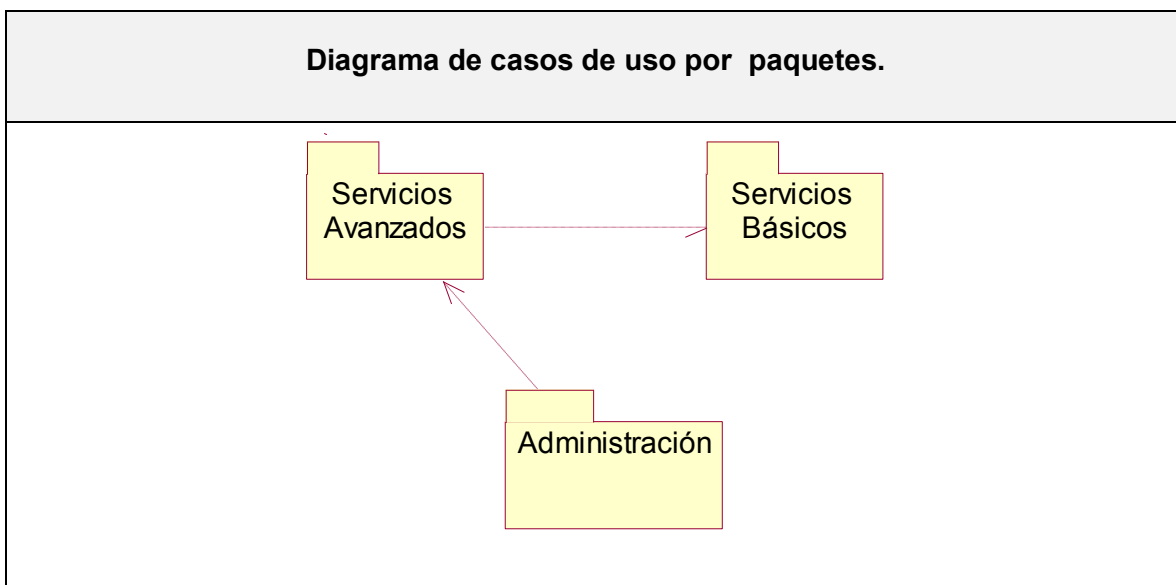


Figura 2. 3 Diagrama de Casos de Uso por paquetes.

Diagrama de casos de uso del paquete Administración.

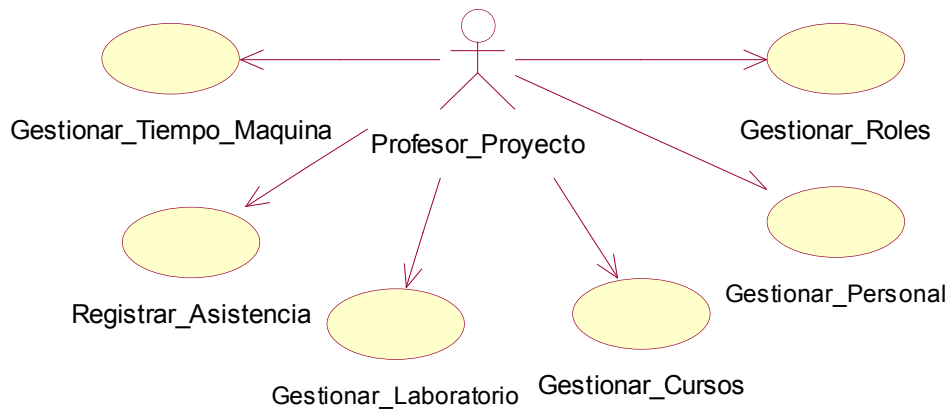


Figura 2. 4 Diagrama de Casos de Uso del paquete Administración.

Diagrama de casos de uso del paquete Servicios Avanzados.

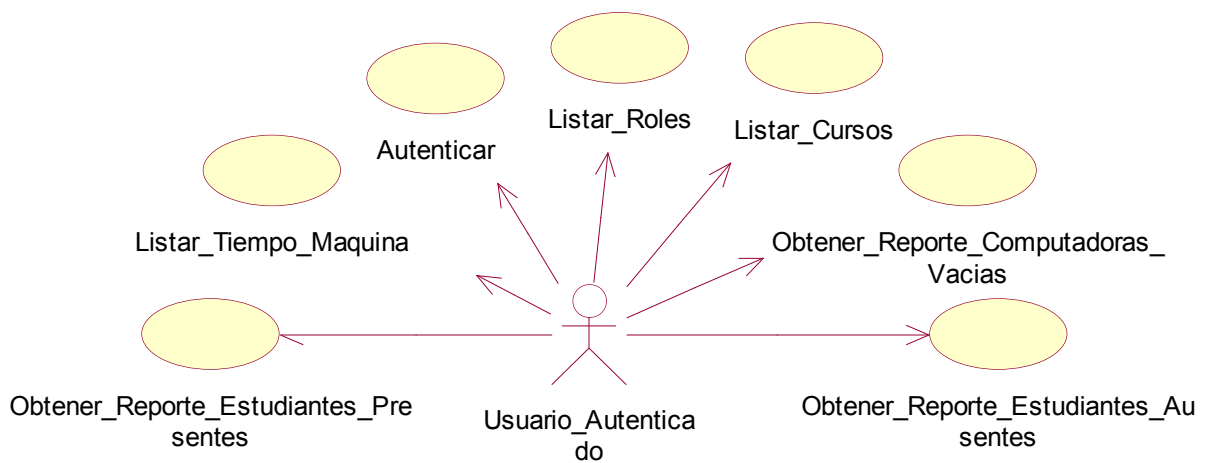


Figura 2. 5 Diagrama de Casos de Uso del paquete Servicios Avanzados.

Diagrama de casos de uso del paquete Servicios Básicos.

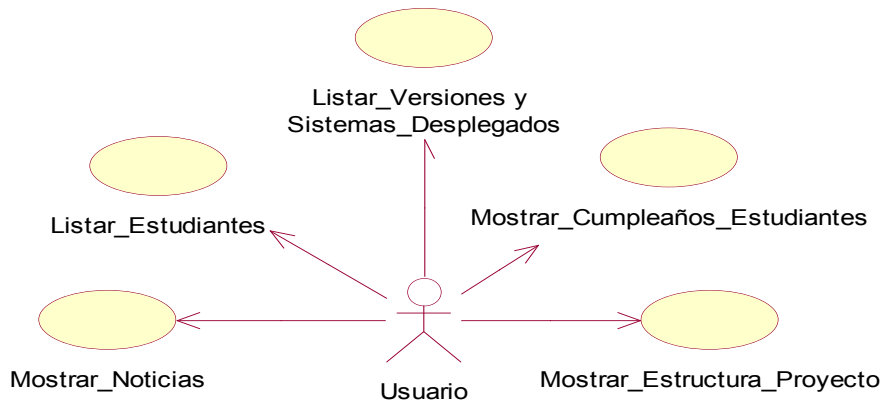


Figura 2. 6 Diagrama de Casos de Uso del paquete Servicios Básicos.

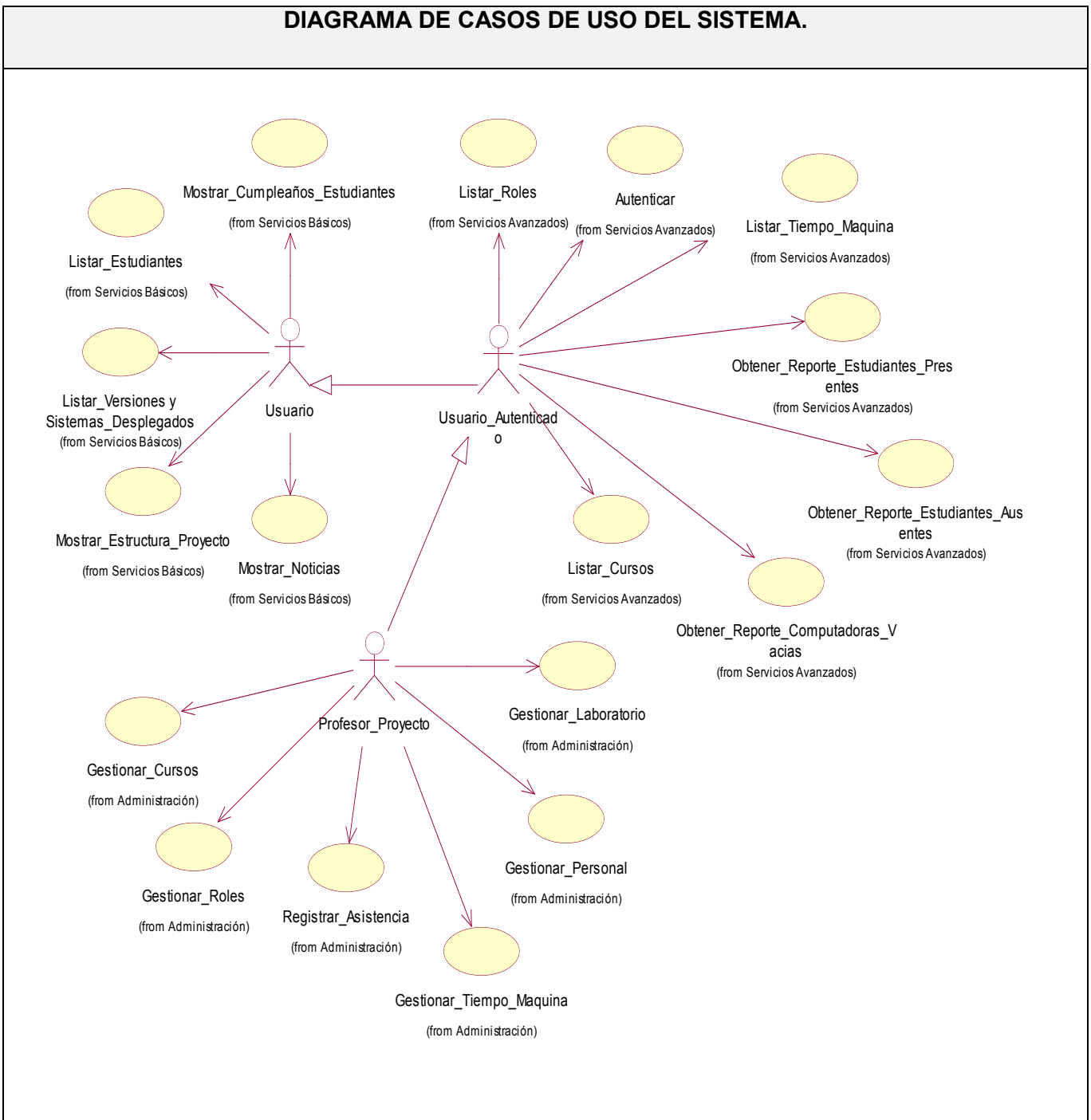


Figura 2. 7 Diagrama de Casos de Uso del Sistema.

2.3.3 Descripción textual de los Casos de Uso del Sistema.

Caso de Uso:	Autenticar
Actores:	Usuario Autenticado
Resumen:	El caso de uso inicia cuando el actor para acceder a la aplicación se autentica introduciendo los datos del dominio UCI.
Prioridad:	Crítico.
Referencias	RF 1
Pos condiciones	El sistema muestra la interfaz después que se ha autenticado.

Tabla 2. 2 Descripción textual del CUS "Autenticar".

Caso de Uso:	Gestionar Tiempo Máquina
Actores:	Profesor Proyecto
Resumen:	El caso de uso inicia cuando el profesor luego de ser autenticado selecciona la opción de Gestionar, donde puede insertar, editar, listar o eliminar los tiempos de máquinas de su laboratorio. Para esto necesita tener actualizada toda la información referente a los estudiantes del proyecto y los laboratorios.
Prioridad:	Crítico.
Referencias	RF 1
Pos condiciones	El sistema deja actualizada la base de datos con la información de los tiempos de máquinas.

Tabla 2. 3 Descripción textual del CUS Gestionar Tiempo Máquina.

CAPÍTULO 2

Características del Sistema

Caso de Uso:	Gestionar Personal
Actores:	Profesor Proyecto
Resumen:	El caso de uso inicia cuando el profesor después de haberse autenticado selecciona la opción de Gestionar Personal, donde puede insertar, listar o eliminar una persona del proyecto. Para esto realiza la búsqueda de de dicha persona en el dominio mediante los criterios: solapín, usuario y CI.
Prioridad:	Crítico.
Referencias	RF 1
Pos condiciones	El sistema deja actualizada la base de datos con la información de la nueva persona.

Tabla 2. 4 Descripción textual del CUS Gestionar Personal.

Caso de Uso:	Gestionar Laboratorio
Actores:	Profesor Proyecto
Resumen:	El caso de uso inicia cuando el profesor luego de ser autenticado selecciona la opción de Gestionar Laboratorio en donde puede insertar, editar, listar o eliminar un laboratorio.
Prioridad:	Crítico.
Referencias	RF 1
Pos condiciones	El sistema deja actualizada la base de datos con la información de los laboratorios.

Tabla 2. 5 Descripción textual del CUS Gestionar Laboratorio.

CAPÍTULO 2

Características del Sistema

Caso de Uso:	Registrar Asistencia
Actores:	Profesor Proyecto
Resumen:	El caso de uso inicia después que el profesor ha registrado los tiempos de máquinas de su equipo de trabajo. Se realiza una búsqueda de tiempo de máquina y luego se controla la asistencia de los estudiantes la cual se clasifica en estudiantes presentes, estudiantes ausentes y computadoras vacías. Teniendo registrada la asistencia se pueden elaborar los reportes de asistencia.
Prioridad:	Crítico.
Referencias	RF 2
Pos condiciones	La base de datos queda actualizada con los registros realizados y está disponible el sistema para hacer los reportes que se deseen.

Tabla 2. 6 Descripción textual del CUS Registrar Asistencia.

Caso de Uso:	Gestionar Curso
Actores:	Profesor Proyecto
Resumen:	El caso de uso inicia cuando el profesor selecciona la opción de Gestionar Curso de los estudiantes, en los se inserta, edita, listar o elimina. Para esto debe estar actualizada la información referente a los estudiantes y los cursos deben ser insertados.
Prioridad:	Crítico.
Referencias	RF 3
Pos condiciones	En el sistema se almacena el curso con las notas de los estudiantes en la base de datos y queda disponible para poder acceder a ellas.

Tabla 2. 7 Descripción textual del CUS Gestionar Curso.

CAPÍTULO 2

Características del Sistema

Caso de Uso:	Gestionar Roles Acreditados y Certificados
Actores:	Profesor Proyecto
Resumen:	El caso de uso inicia cuando el profesor luego de ser autenticado selecciona en el menú la opción de Gestionar Roles, donde puede insertar, editar, listar o eliminar. Para esto debe estar actualizada la información referente a los estudiantes y los roles deben ser insertados.
Prioridad:	Secundario.
Referencias	RF 4
Pos condiciones	El sistema actualiza la base de datos con los registros realizados y queda disponible la información para los que la deseen ver.

Tabla 2. 8 Descripción textual del CUS Gestionar Roles.

Caso de Uso:	Obtener Reporte Estudiantes Presentes
Actores:	Usuario Autenticado
Resumen:	El caso de uso inicia cuando un miembro del proyecto después de haberse autenticado selecciona en las opciones del menú “Reporte Estudiantes Presentes”, para ver los estudiantes que estuvieron presentes en el laboratorio dado el laboratorio y la fecha.
Prioridad:	Secundario.
Referencias	RF 5
Pos condiciones	Se obtiene el reporte de asistencia en un día determinado.

Tabla 2. 9 Descripción textual del CUS Obtener Reporte Estudiantes Presentes.

CAPÍTULO 2

Características del Sistema

Caso de Uso:	Obtener Reporte Estudiantes Ausentes
Actores:	Usuario Autenticado
Resumen:	El caso de uso inicia cuando un miembro del proyecto después de haberse autenticado selecciona en las opciones del menú “Reportes Estudiantes Ausentes”, para ver los estudiantes que estuvieron ausentes al laboratorio en una fecha y un laboratorio determinado.
Prioridad:	Secundario.
Referencias	RF 5
Pos condiciones	Se obtiene el reporte de asistencia de un día determinado.

Tabla 2. 10 Descripción textual del CUS Obtener Reporte Estudiantes Ausentes.

Caso de Uso:	Obtener Reporte Computadoras Vacías
Actores:	Usuario Autenticado
Resumen:	El caso de uso inicia cuando un miembro del proyecto después de haberse autenticado selecciona en las opciones del menú “Reportes Computadoras Vacías”, para ver las computadoras que estuvieron vacías en una fecha y un laborstorio determinado.
Prioridad:	Secundario.
Referencias	RF 5
Pos condiciones	Se obtiene el reporte de las computadoras que estuvieron vacías en un día determinado.

Tabla 2. 11 Descripción textual del CUS Obtener Reporte Computadoras Vacías.

CAPÍTULO 2

Características del Sistema

Caso de Uso:	Listar Tiempo Máquina
Actores:	Usuario Autenticado
Resumen:	El caso de uso inicia cuando el actor quiere ver el tiempo de máquina y para esto selecciona en el menú la opción Listar Tiempo de Máquina y se muestra una interfaz donde aparecen los tiempos de máquinas.
Prioridad:	Secundario.
Referencias	RF 6
Pos condiciones	El sistema deja visualizada la información de los tiempos de máquina.

Tabla 2. 12 Descripción textual del CUS Listar Tiempo Máquina.

Caso de Uso:	Listar Estudiantes
Actores:	Usuario Autenticado
Resumen:	El caso de uso inicia cuando un usuario luego de autenticarse quiere ver los datos de los estudiantes del proyecto y para esto selecciona en el menú la opción Listar Estudiantes y se muestra en un interfaz donde aparece dicha información.
Prioridad:	Secundario.
Referencias	RF 7
Pos condiciones	El sistema deja visualizada la información de cada estudiante.

Tabla 2. 13 Descripción textual del CUS Listar Estudiantes.

Caso de Uso:	Listar Roles Acreditados y Certificados
Actores:	Usuario Autenticado
Resumen:	El caso de uso inicia cuando cualquier persona después de autenticada

CAPÍTULO 2

Características del Sistema

	quiere ver los roles acreditados o certificados de un estudiante determinado, hay que seleccionar en el menú la opción Listar Roles y se muestra una interfaz donde se listan los datos correspondientes.
Prioridad:	Secundario.
Referencias	RF 8
Pos condiciones	El sistema muestra los estudiantes que tienen algún Rol Acreditado y Certificado

Tabla 2. 14 Descripción textual del CUS Listar Roles Acreditados y Certificados.

Caso de Uso:	Listar Curso
Actores:	Usuario Autenticado
Resumen:	El caso de uso inicia cuando una persona autenticada quiere ver las notas que tiene un estudiante en un curso determinado, hay que seleccionar en el menú la opción Listar Curso y se muestra una interfaz donde se listan los datos correspondientes
Prioridad:	Secundario.
Referencias	RF 9
Pos condiciones	El sistema muestra el curso con las notas del estudiante seleccionado.

Tabla 2. 15 Descripción textual del CUS Listar Curso.

Caso de Uso:	Mostrar Cumpleaños Estudiantes
Actores:	Usuarios
Resumen:	El caso de uso inicia cuando se le muestran al usuario los miembros del proyecto que están cumpliendo año ese día.
Prioridad:	Opcional.

CAPÍTULO 2

Características del Sistema

Referencias	RF 10
Pos condiciones	El sistema muestra las personas homenajeadas.

Tabla 2. 16 Descripción textual del CUS Mostrar Cumpleaños Estudiantes.

Caso de Uso:	Mostrar Información Productiva
Actores:	Usuarios
Resumen:	El caso de uso inicia cuando un usuario quiere ver la Información Productiva. Para esto debe pinchar en el vínculo Noticias.
Prioridad:	Auxiliar
Referencias	RF 13
Pos condiciones	El sistema deja visualizada la información productiva que presenta.

Tabla 2. 17 Descripción textual del CUS Mostrar Información Productiva.

Caso de Uso:	Mostrar Estructura
Actores:	Usuarios
Resumen:	El caso de uso inicia cuando un usuario quiere ver la estructura que tiene el proyecto. Para esto debe pinchar en la opción "Qué hacemos" donde se mostrará una interfaz con los módulos y con vinculos a sus características.
Prioridad:	Auxiliar
Referencias	RF 12
Pos condiciones	El sistema deja visualizada la estructura que presenta el proyecto.

Tabla 2. 18 Descripción textual del CUS Mostrar Estructura.

CAPÍTULO 2

Características del Sistema

Caso de Uso:	Mostrar Versiones y Sistemas Desplegados
Actores:	Usuarios
Resumen:	El caso de uso inicia cuando un usuario quiere ver las versiones y los sistemas desplegados y para esto selecciona en el menú el vínculo “Qué hacemos” y se muestra en un interfaz donde aparece la información solicitada.
Prioridad:	Auxiliar
Referencias	RF 11
Pos condiciones	El sistema deja visualizada la información de las versiones y los sistemas desplegados.

Tabla 2. 19 Descripción textual del CUS Mostrar Versiones del Proyecto.

Conclusiones.

Con la realización de este capítulo se tiene una vista del sistema a desarrollar, requisitos funcionales y no funcionales. También se tiene la identificación y descripción de los actores que se relacionan con las funcionalidades del sistema y con cada caso de uso propuesto a desarrollar en el Portal Web de APS. Dándoles la prioridad requerida y relacionándolos con las funcionalidades que el sistema debe cumplir. Con la conclusión de este capítulo se le da paso al desarrollo de las Disciplinas pendientes como son: Análisis y Diseño, Implementación y Pruebas del Portal de APS.

CAPÍTULO 3: ANÁLISIS Y DISEÑO.

Introducción.

Para la realización de este capítulo se desarrolla el análisis del sistema, mediante la elaboración de diagrama de clases de análisis de la aplicación, así como los diagramas de interacción, en este caso el de colaboración, el cual tiene flujos de mensajes entre sus clases de análisis.

Se encuentra además el diseño del sistema que parte del análisis anterior, definiéndose los diagramas de clases de diseño y los diagramas de interacción siguiendo un flujo de proceso. Posterior a esto es que se realiza la descripción de las clases y de los métodos del negocio.

3.1 Análisis.

En el Flujo de Trabajo de Análisis y Diseño se le presta mayor interés a los requisitos funcionales y se tiene la vista del sistema que trata de ver “qué” es lo que se hace.

3.1.1 Modelo de Análisis.

El modelo de análisis es el que contiene las clases de análisis y sus objetos ordenados en paquetes. Además es el resultado de las actividades de analizar los Casos de Uso.

3.1.2 Diagramas de Clases de Análisis.

El diagrama de clases del análisis es un artefacto que se realiza para cada caso de uso del sistema y muestra las clases que participan y sus relaciones. En este diagrama se aprecian tres tipos de clase: Interfaz, Controladora y Entidad.

- La clase Interfaz: es donde se modela la interacción del sistema con los actores, dicho sistema puede ser externo.
- La clase Controladora: es la encargada de la coordinación, secuencia, transacciones y control de otros objetos. En varias ocasiones los aspectos dinámicos se modelan con esta clase, pues ella maneja y coordina las acciones y los flujos de control principales, y delega trabajo a los objetos de interfaz y de entidad.
- La clase Entidad: es la encargada de modelar la información que tiene una larga vida y en ocasiones persistente, además, muestra una estructura de datos lógica y contribuye a comprender de que información depende el sistema.

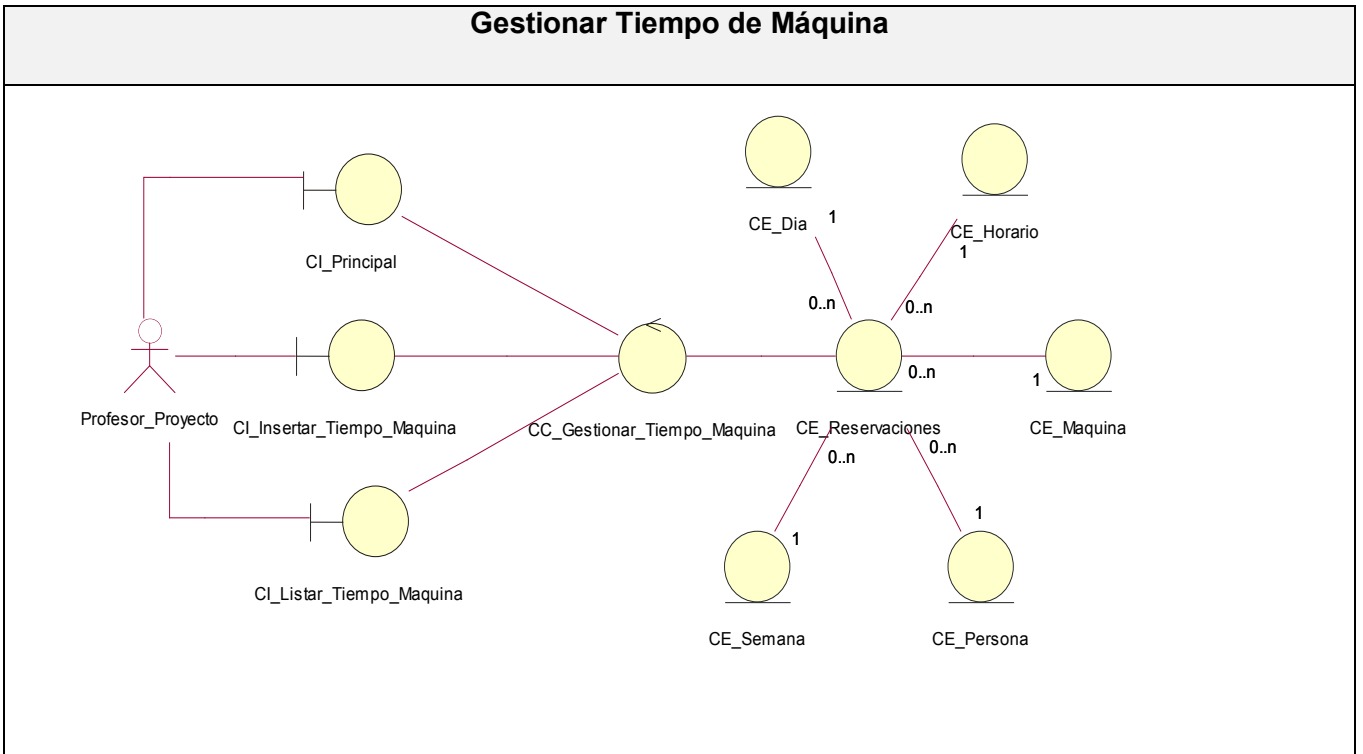


Figura 3. 1 Diagrama de clase de Análisis "Gestionar Tiempo de Máquina".

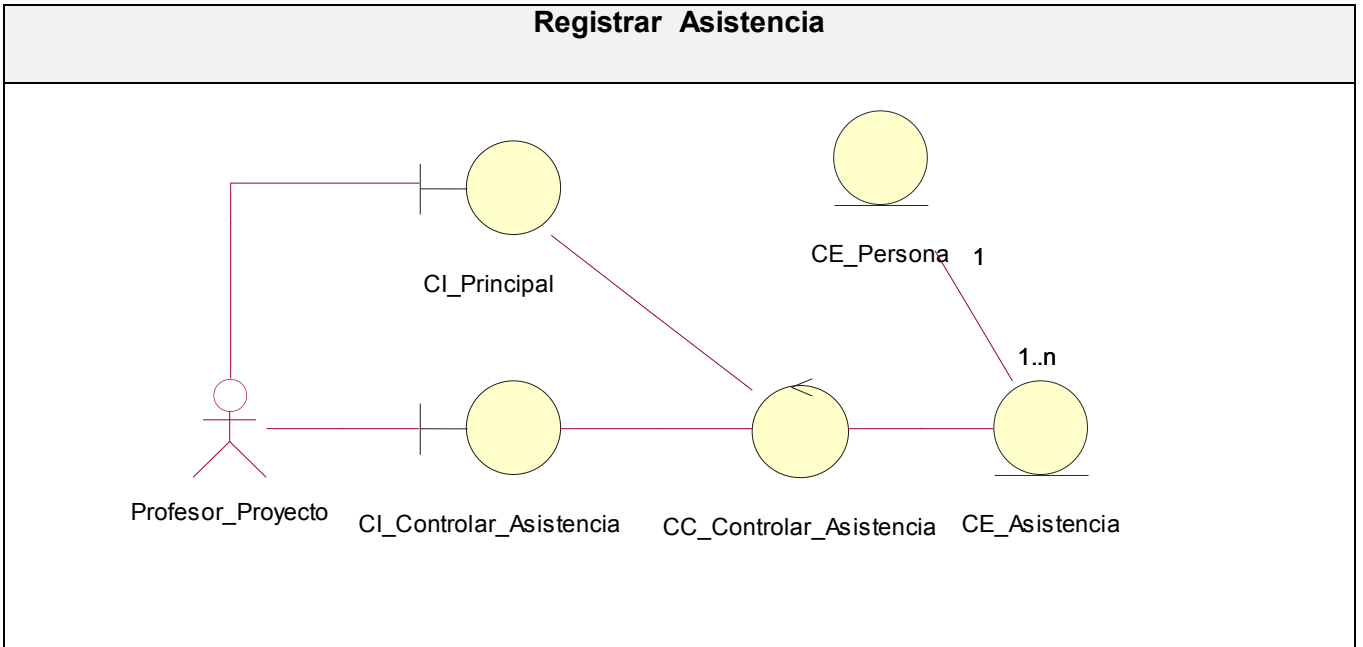


Figura 3. 2 Diagrama de clase de Análisis "Registrar Asistencia".

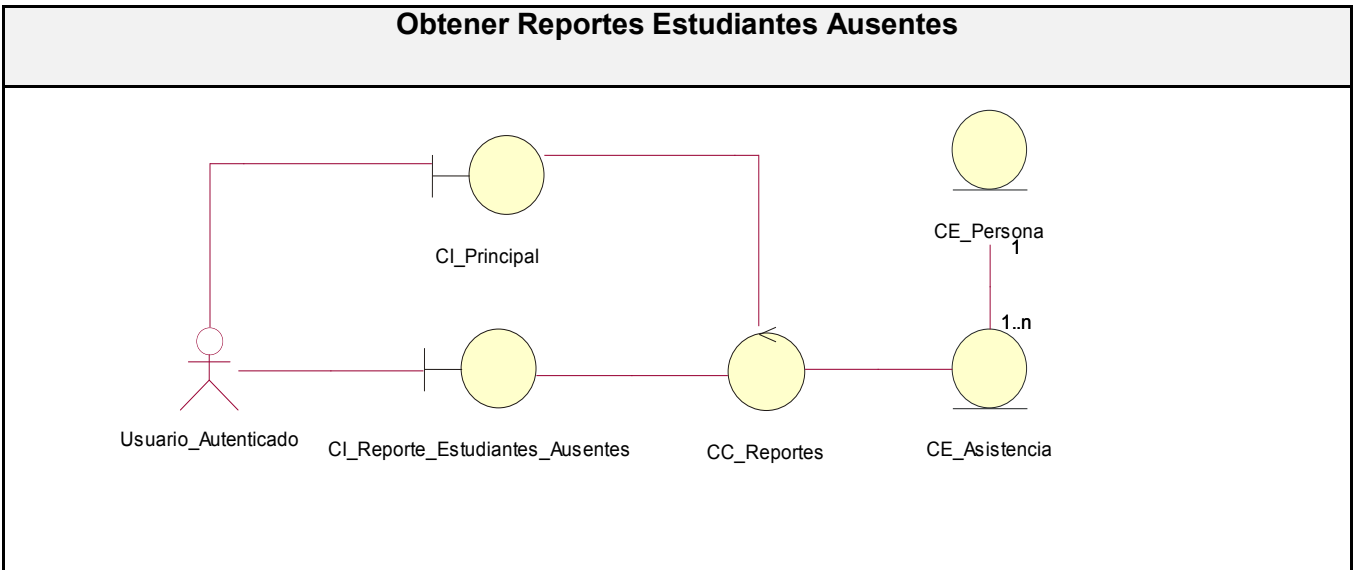


Figura 3. 3 Diagrama de clase de Análisis "Obtener Reportes Estudiantes Ausentes".

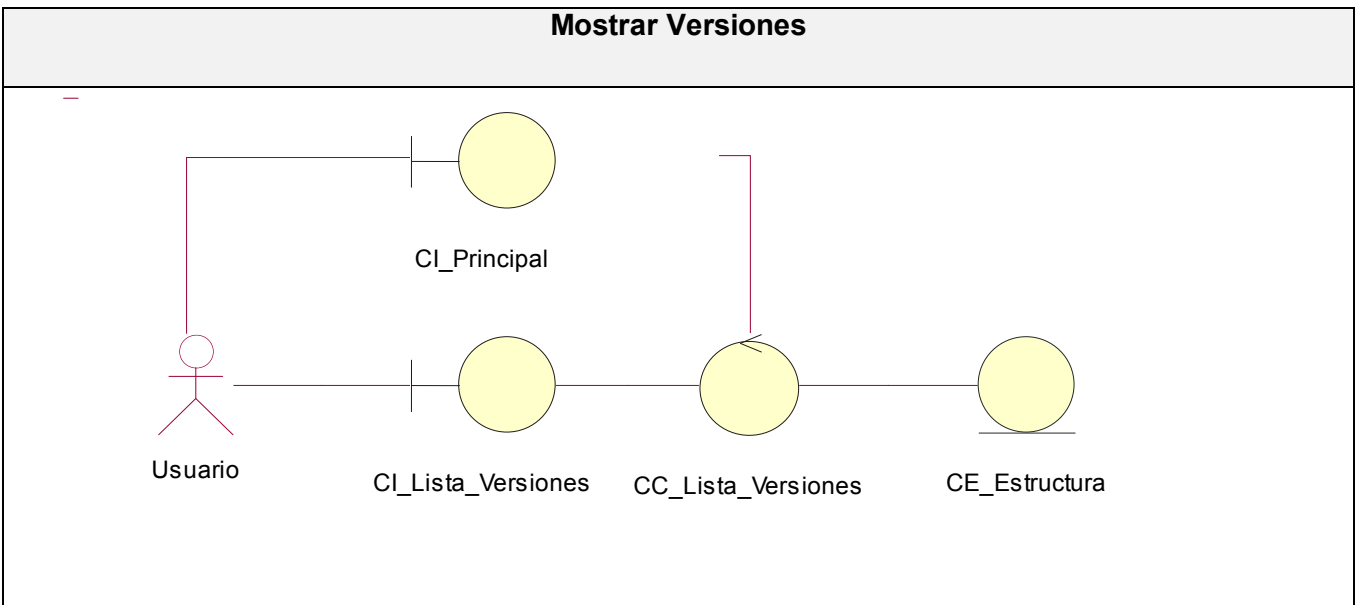


Figura 3. 4 Diagrama de clase de Análisis "Mostrar Versiones".

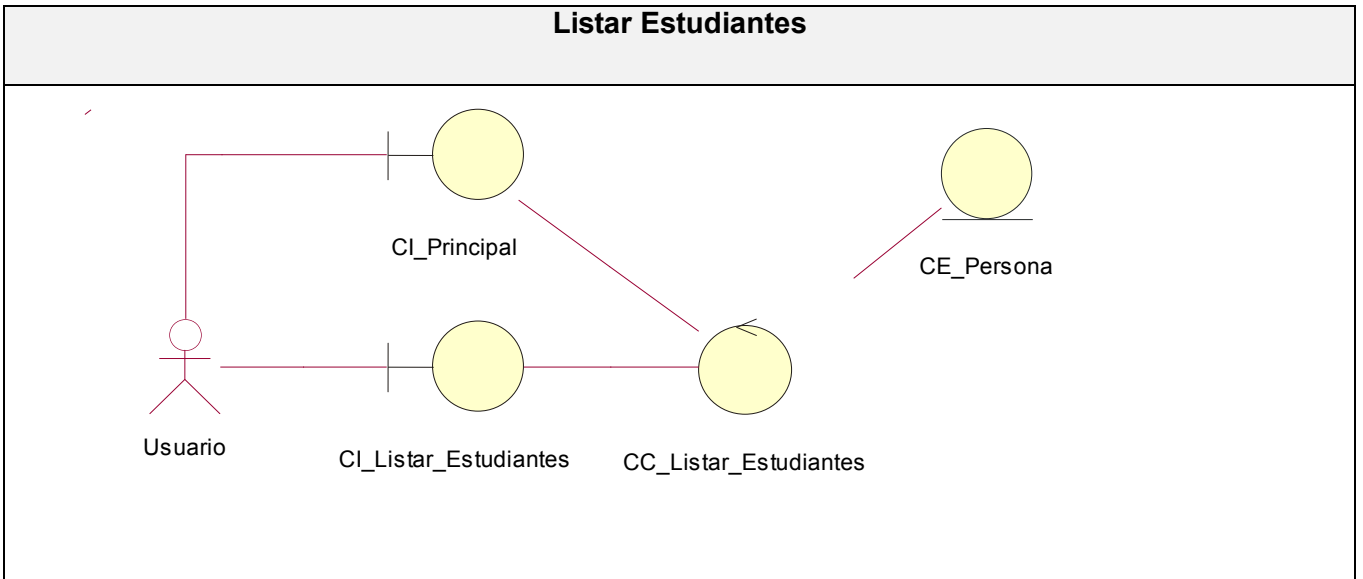


Figura 3. 5 Diagrama de clase de Análisis "Listar Estudiantes".

3.1.3 Diagramas de Interacción.

A continuación se muestran algunos de los diagramas de colaboración, el resto están en el expediente del proyecto.

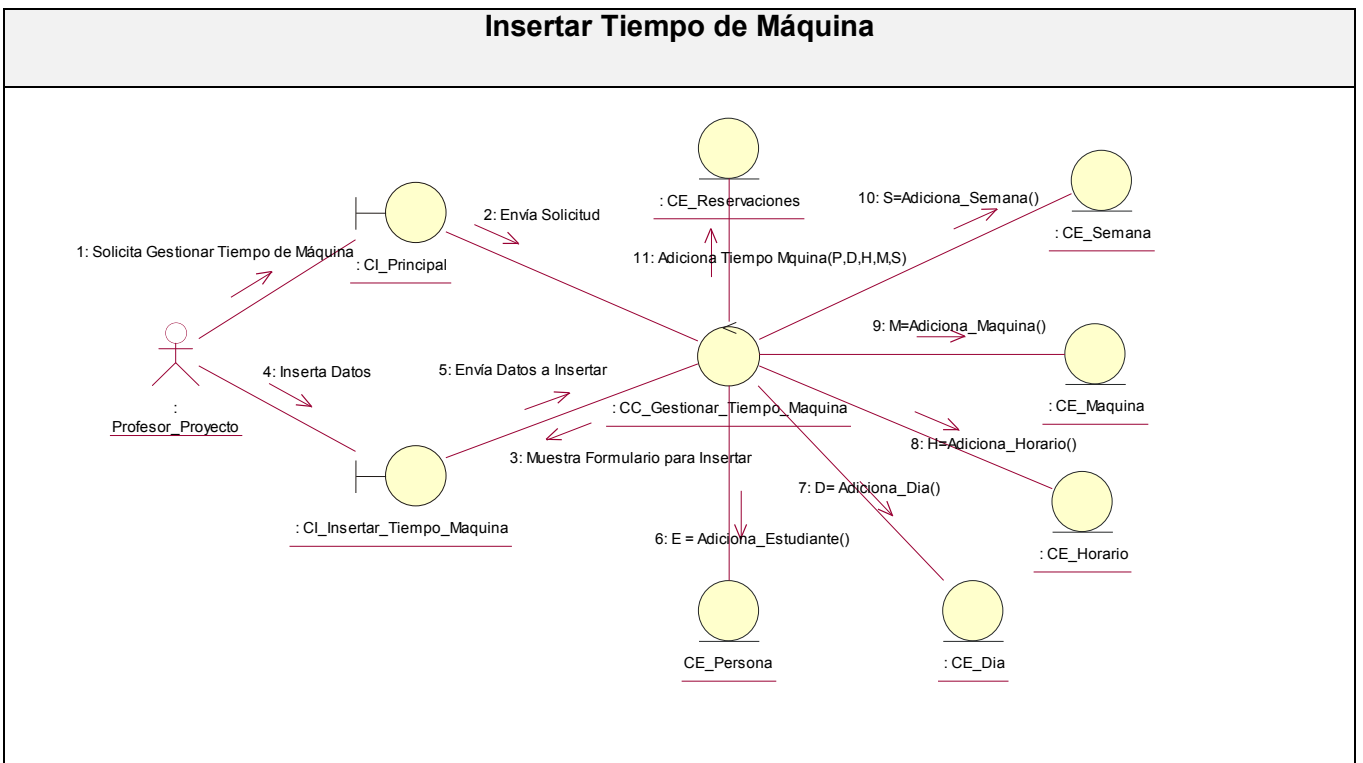


Figura 3. 6 Diagrama de Colaboración "Insertar Tiempo de Máquina".

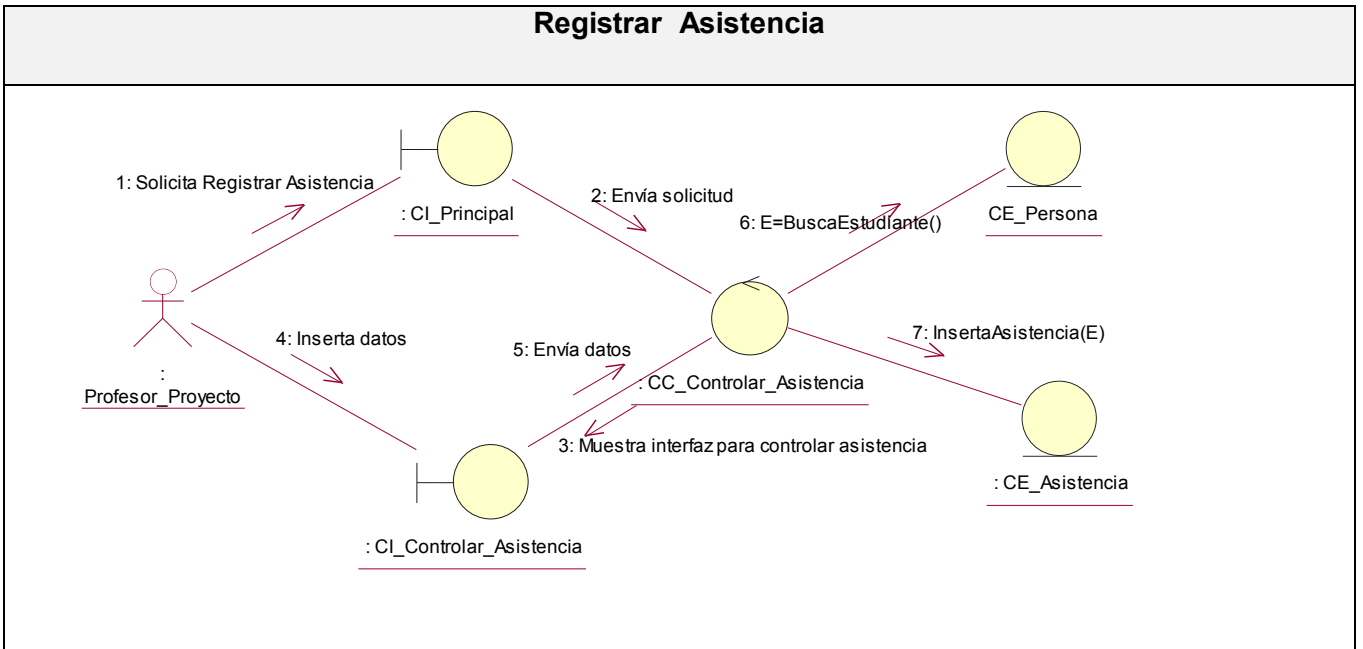


Figura 3. 7 Diagrama de Colaboración "Registrar Asistencia".

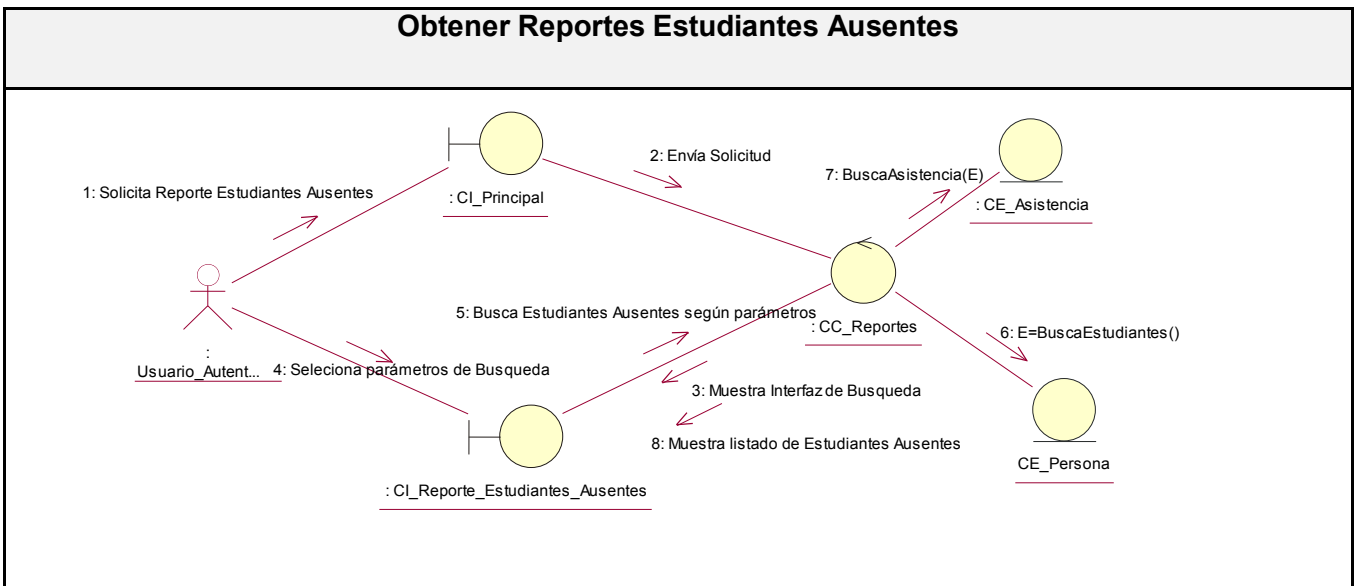


Figura 3. 8 Diagrama de Colaboración "Obtener Reportes Estudiantes Ausentes".

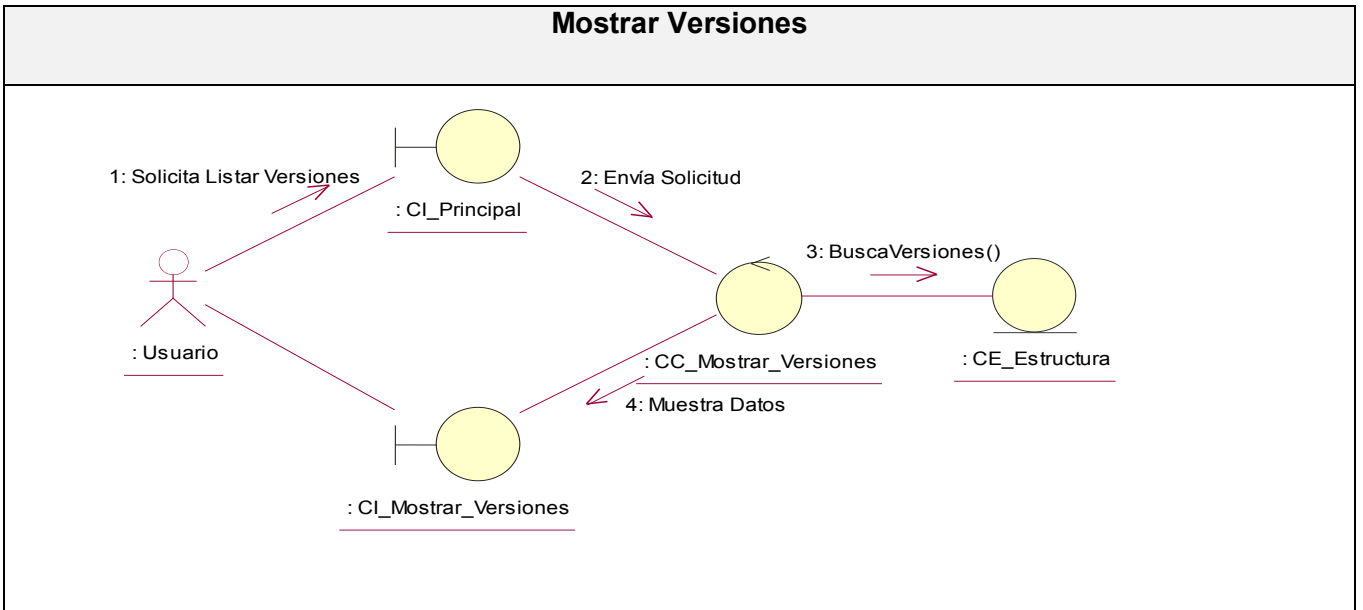


Figura 3. 9 Diagrama de Colaboración "Mostrar Versiones".

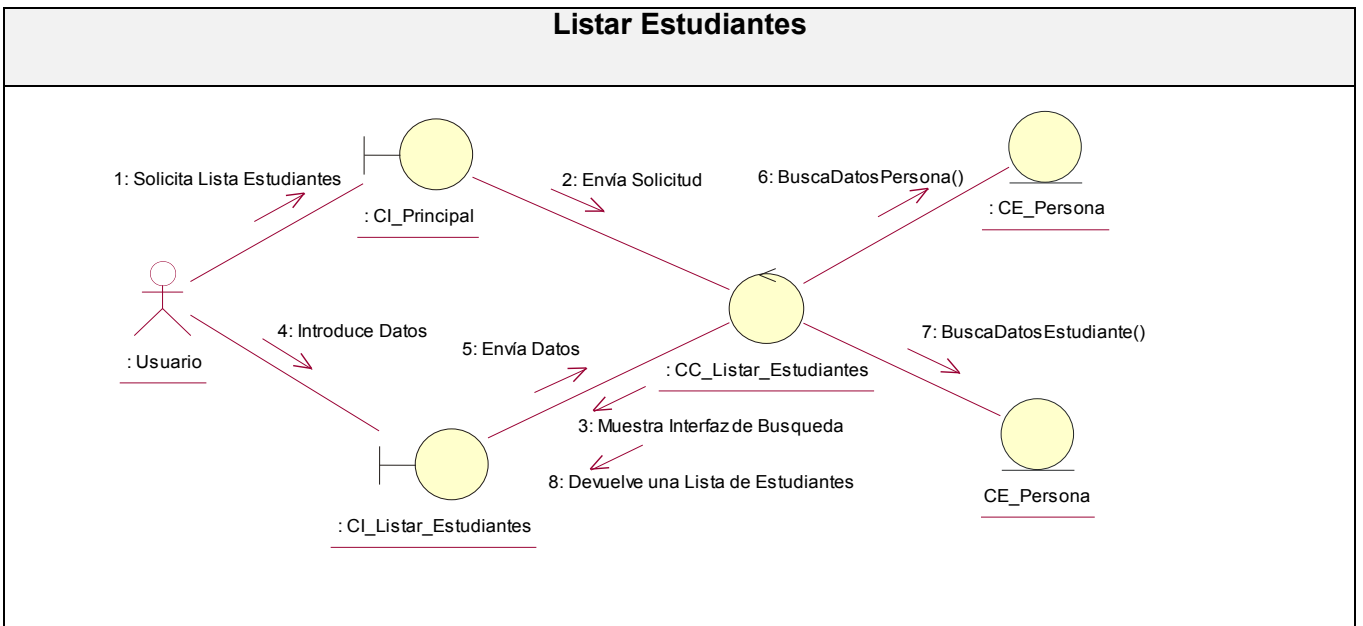


Figura 3. 10 Diagrama de Colaboración "Listar Estudiantes".

3.2 Diseño.

El diseño es el refinamiento del análisis, tiene en cuenta los requisitos no funcionales, y es el "cómo" cumple el sistema sus objetivos. El diseño es lo que ayuda a mantener una arquitectura estable y sólida, y crear un plano del modelo de implementación, es normal guardar y mantener el modelo de diseño a través del ciclo de vida completo del software. Además es el inicio al Flujo de Trabajo Implementación.

3.2.1 Modelo de Diseño.

Es un modelo de objeto que describe la realización física de los casos de uso, centrándose en como los requisitos funcionales y no funcionales, junto con otras restricciones relacionadas con el entorno de implementación, tiene impacto en el sistema a considerar.

3.2.2 Diagrama de Clases de Diseño.

El diagrama de clases de diseño es una representación más concreta que el diagrama de clases del análisis. Representa la parte estática del sistema, las clases y sus relaciones.

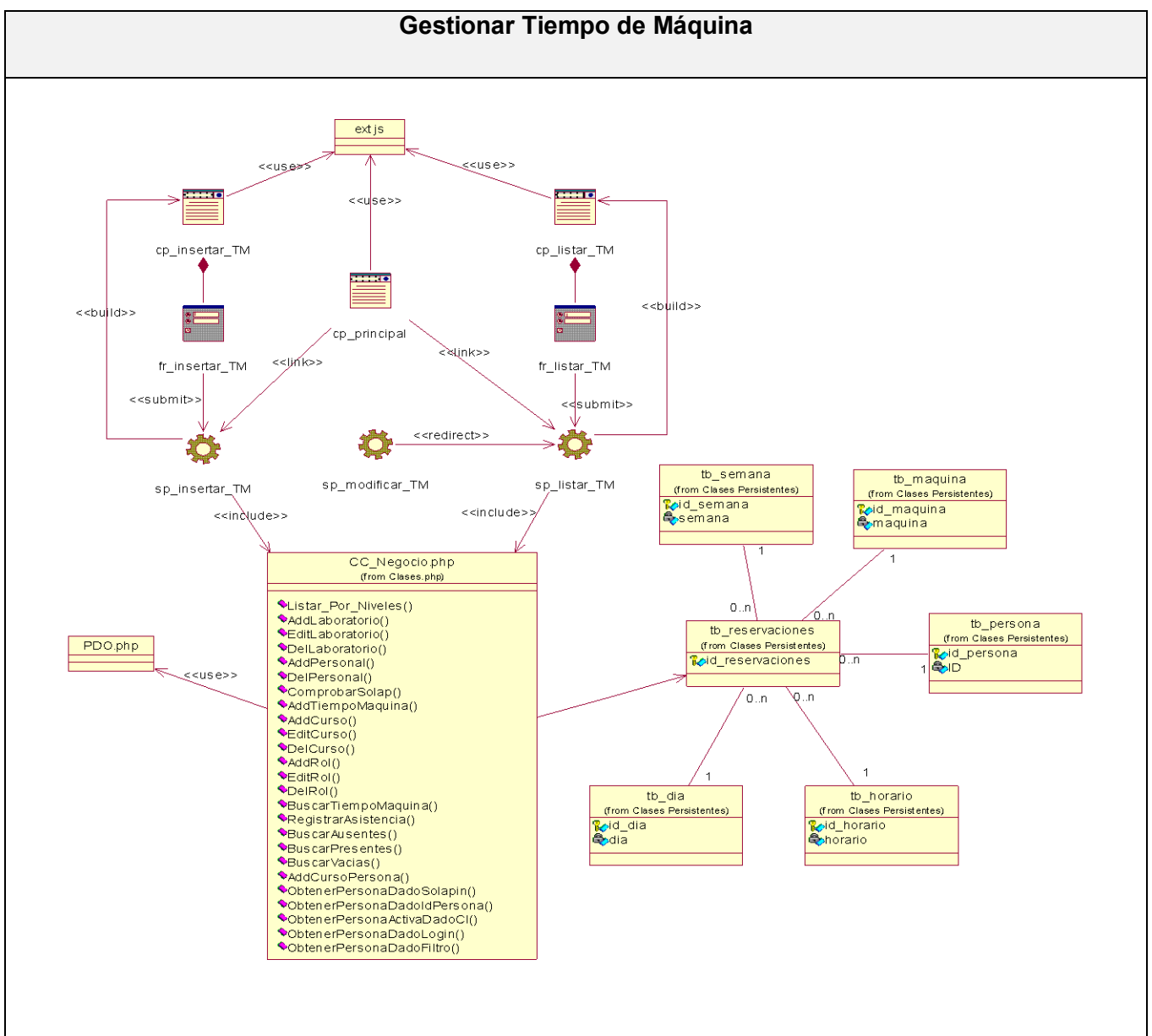


Figura 3. 6 Diagrama de clase de Diseño "Gestionar Tiempo de Máquina".

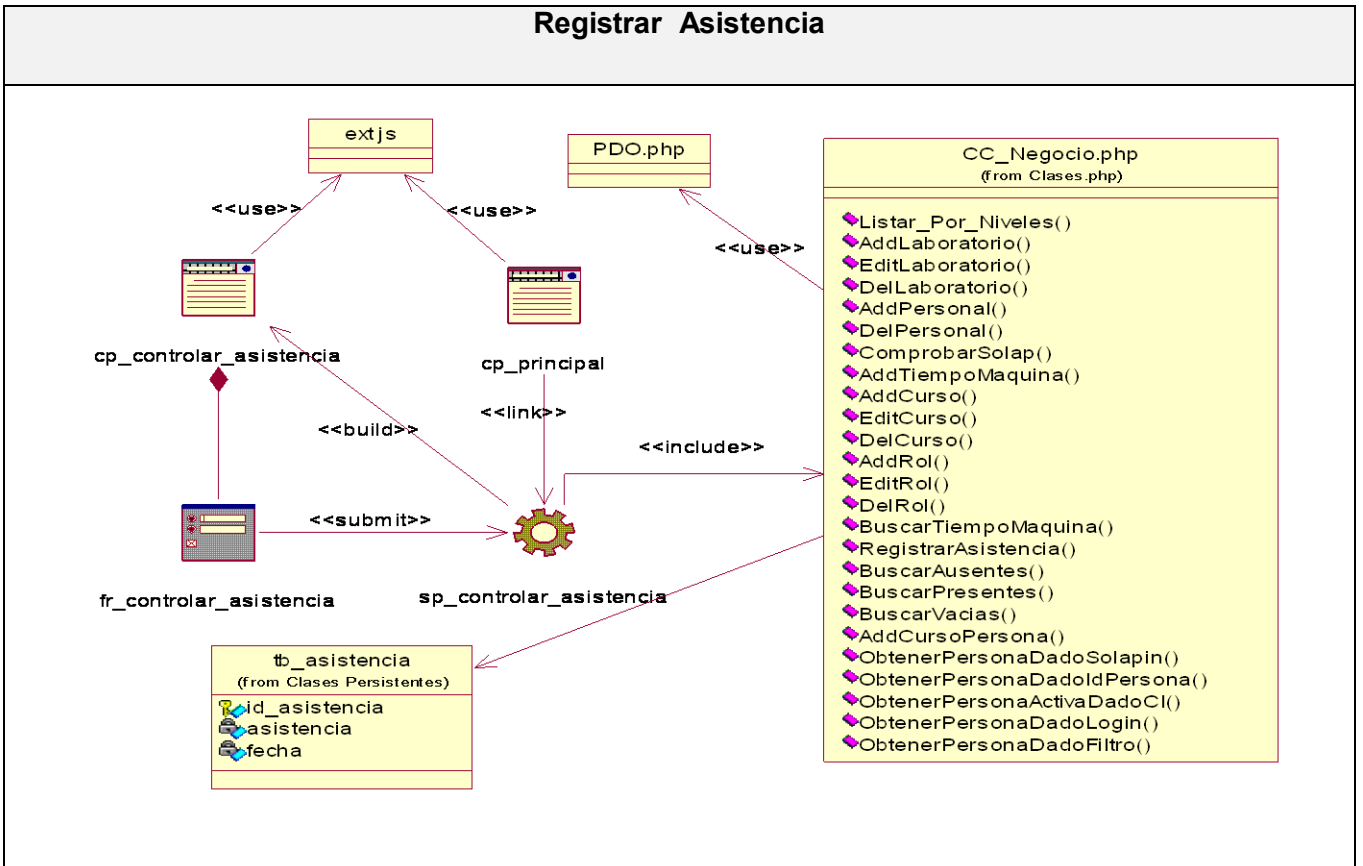


Figura 3. 7 Diagrama de clase de Diseño "Registrar Asistencia".

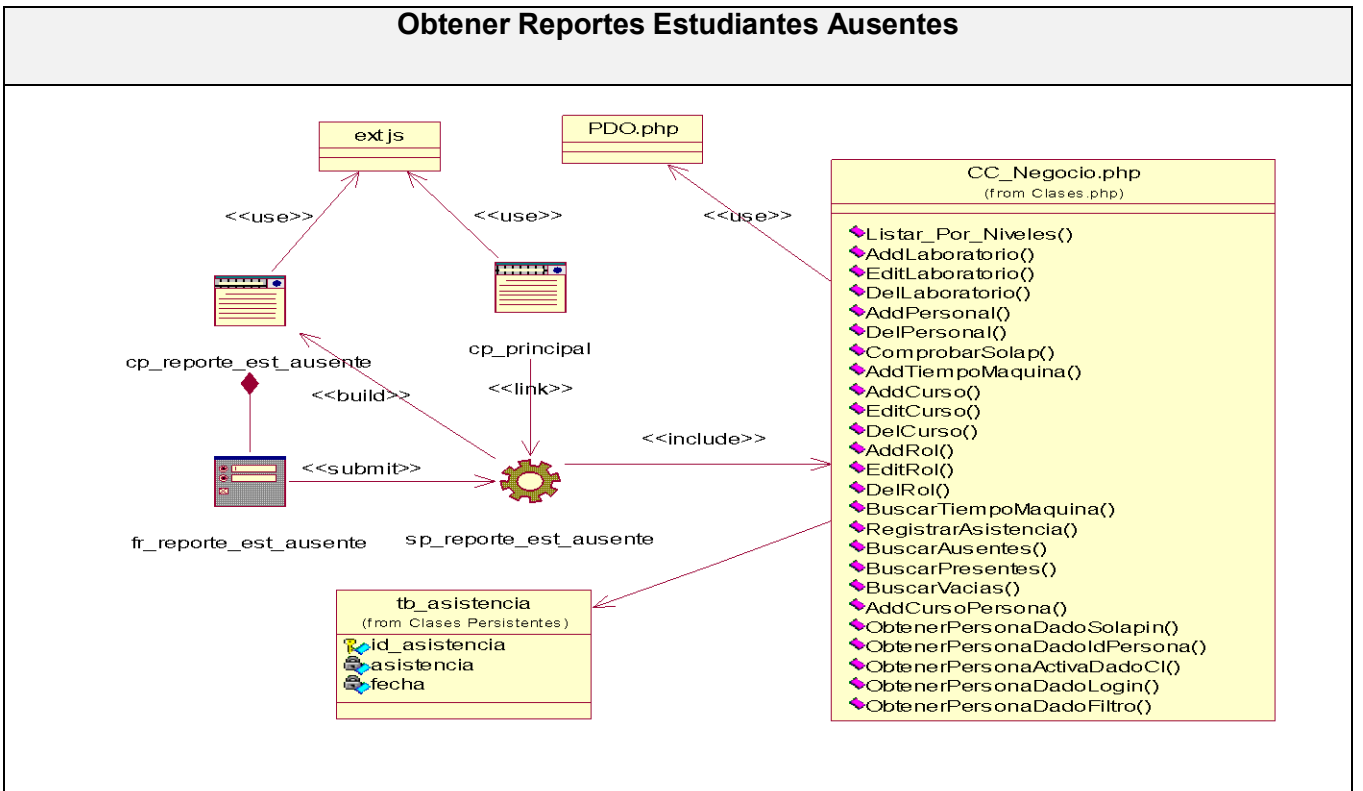


Figura 3. 8 Diagrama de clase de Diseño "Obtener Reportes Estudiantes Ausentes".

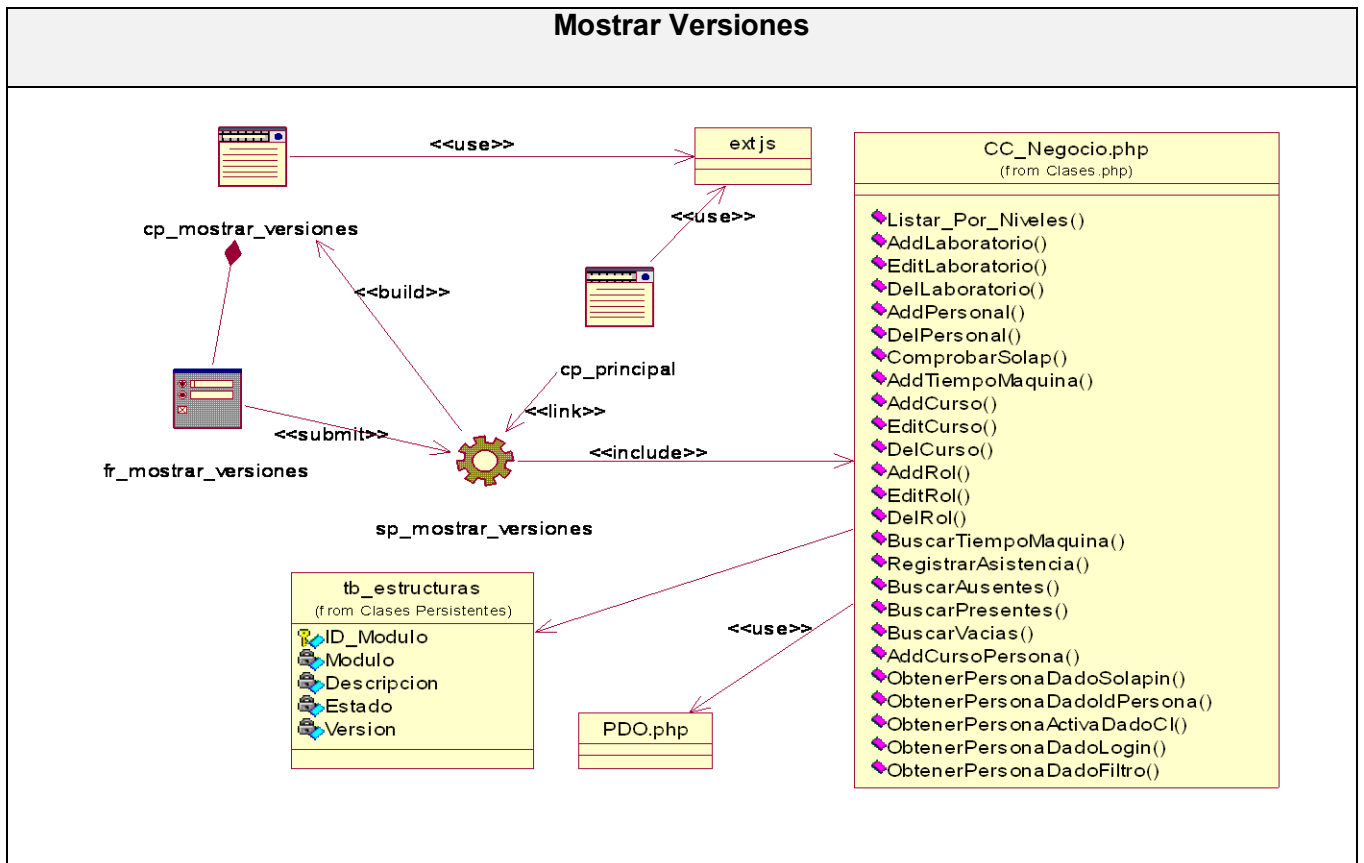


Figura 3. 9 Diagrama de clase de Diseño "Mostrar Versiones".

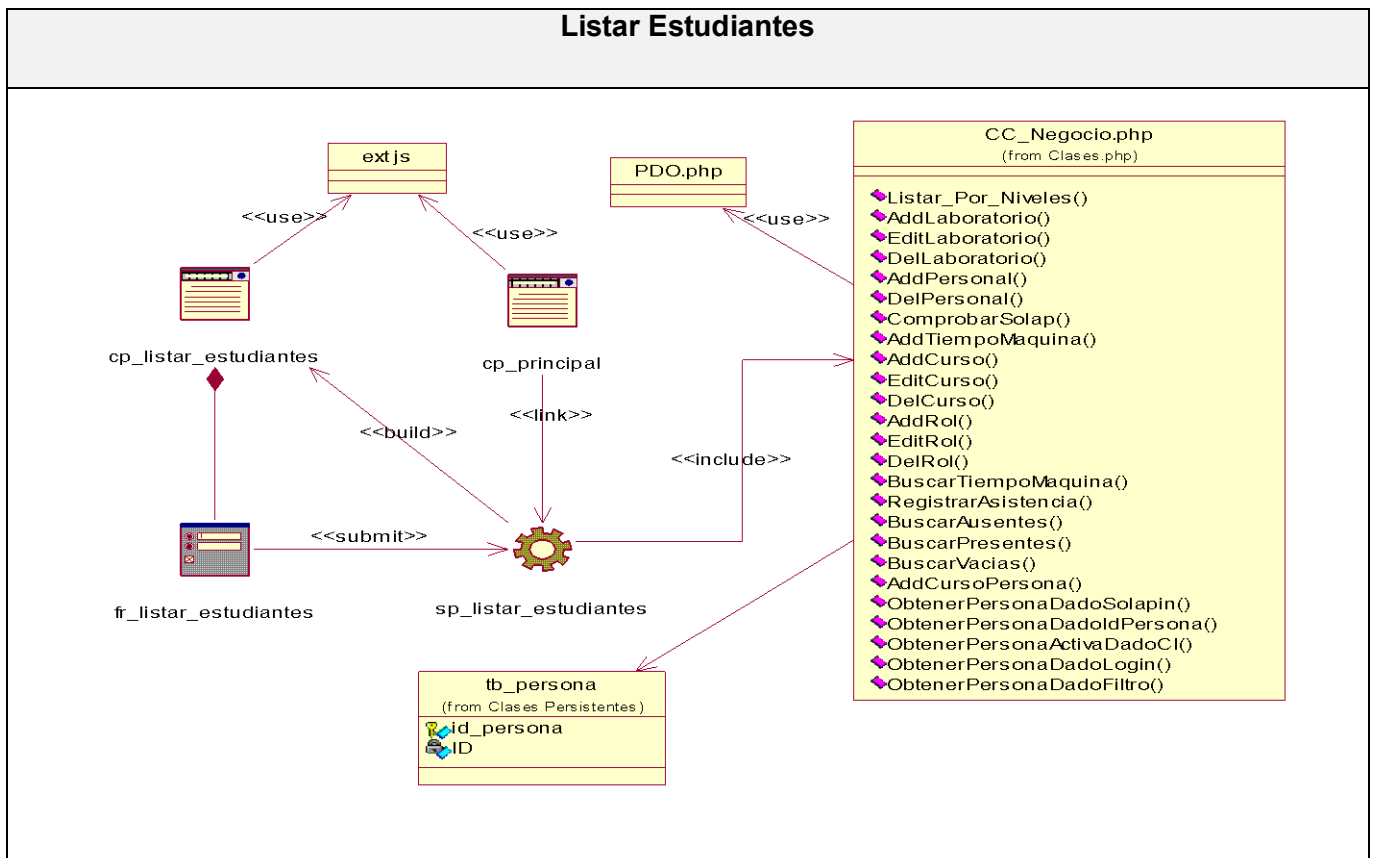


Figura 3. 10 Diagrama de clase de Diseño "Listar Estudiantes".

3.2.3 Descripción de las páginas Clientes y Servidoras.

Para tener un mejor entendimiento y comprensión del funcionamiento que tendrá el sistema en desarrollo se describen a continuación algunas de las clases asociadas a la Capa de Presentación, la cual cuenta con las páginas clientes, y las páginas servidoras.

3.2.3.1 Descripción de las páginas clientes.

A continuación se muestra la descripción de algunas de las páginas clientes del Portal de APS, para ver las descripciones restantes remitirse al expediente de proyecto.

Nombre	cp_insertar_rol
Tipo de Clase	Client Page
<p>Descripción General: La clase cp_insertar_rol es una página Web que se ejecuta del lado del cliente sobre un navegador Web. Permite a los Editores insertar un nuevo rol. Es utilizada en el siguiente caso de uso:</p> <ul style="list-style-type: none"> ➤ Gestionar_Roles 	

Tabla 3. 1 Descripción de la páginas cliente cp_insertar_rol.

Nombre	cp_listar_curso
Tipo de Clase	Client Page
<p>Descripción General: La clase cp_listar_curso es una página Web que se ejecuta del lado del cliente sobre un navegador Web. Permite a los Visualizadores o Editores obtener un listado de los cursos programados por el proyecto. A través de esta página se puede visualizar la información y el resultado de las búsquedas realizada. Es utilizada en el siguiente caso de uso:</p> <ul style="list-style-type: none"> ➤ Gestionar_Cursos ➤ Listar_Cursos 	

Tabla 3. 2 Descripción de la página cliente cp_listar_curso.

3.2.3.2 Descripción de las páginas servidoras.

A continuación se muestra la descripción de algunas de las páginas servidoras del Portal de APS, para ver las descripciones restantes remitirse al expediente de proyecto.

Nombre	sp_insertar_rol
Tipo de Clase	Server Page
<p>Descripción General: La clase sp_insertar_rol es una clase que se ejecuta del lado del servidor en la Capa de Presentación. Su actividad es construir la página cliente cp_insertar_rol. La clase sp_insertar_rol envía los datos recibidos hacia la Clase de Negocio la cual contiene una serie de funcionalidades entre las que esta la de insertar rol, devolviendo una respuesta a la clase sp_insertar_rol. Es utilizada en los siguientes casos de uso:</p> <ul style="list-style-type: none"> ➤ Gestionar_Roles 	

Tabla 3. 3 Descripción de la página servidora sp_insertar_rol.

Nombre	sp_listar_curso
Tipo de Clase	Server Page
<p>Descripción General: La clase sp_listar_curso es una clase que se ejecuta del lado del servidor en la Capa de Presentación. Su actividad es construir una página cliente cp_listar_curso. La clase sp_listar_curso envía los datos recibidos hacia la Clase de Negocio la cual contiene una serie de funcionalidades entre las que esta la de listar curso, devolviendo una respuesta a la clase sp_listar_curso. Es utilizada en los siguientes casos de uso:</p> <ul style="list-style-type: none"> ➤ Gestionar_Cursos ➤ Listar_Cursos 	

Tabla 3. 4 Descripción de la página servidora sp_listar_curso.

3.2.4 Diagrama de Interacción.

A continuación se muestran algunos de los diagramas de secuencias, el resto están en el expediente del proyecto.

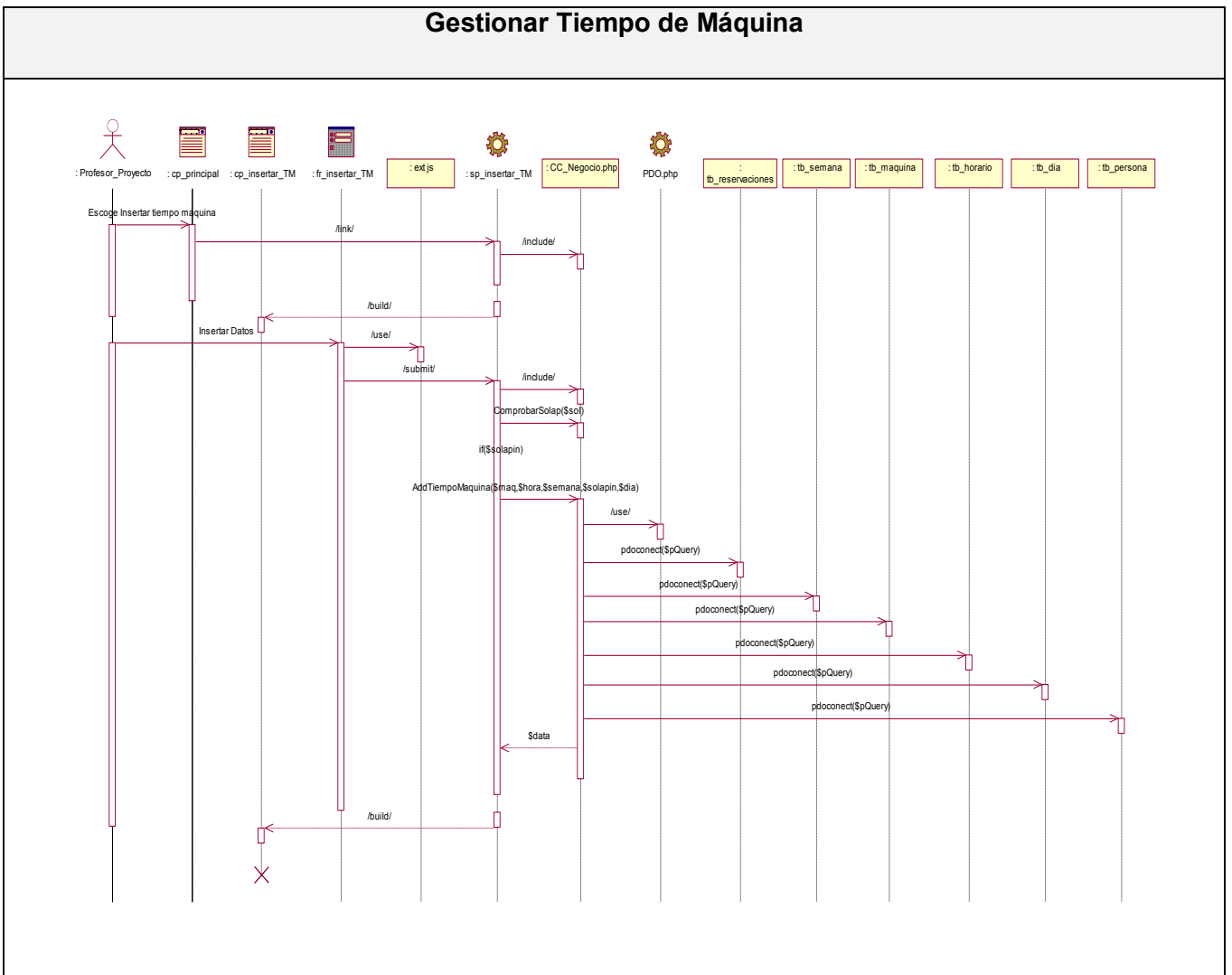


Figura 3. 11 Diagrama de Secuencia "Gestionar Tiempo de Máquina".

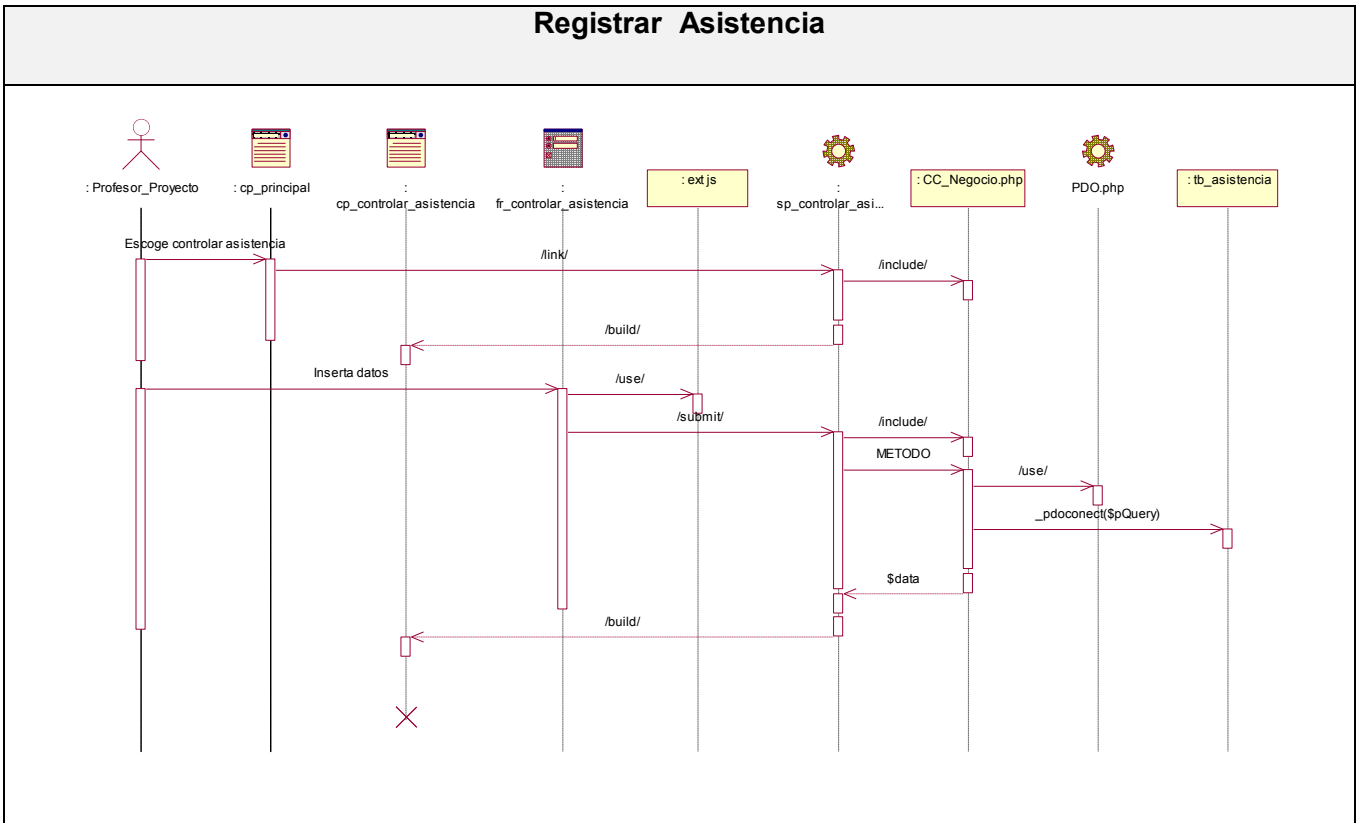


Figura 3. 12 Diagrama de Secuencia "Registrar Asistencia".

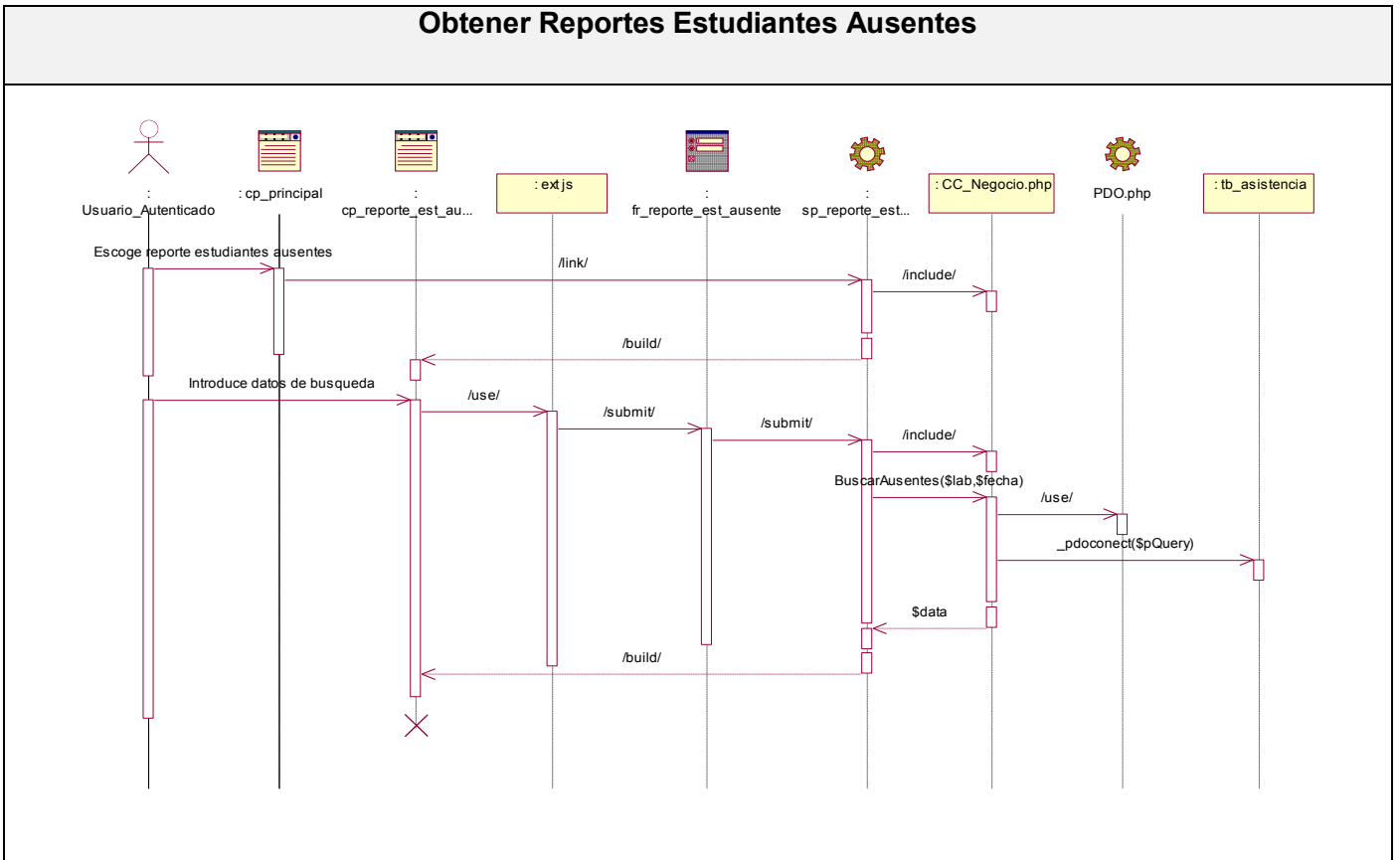


Figura 3. 13 Diagrama de Secuencia "Obtener Reportes Estudiantes Ausentes".

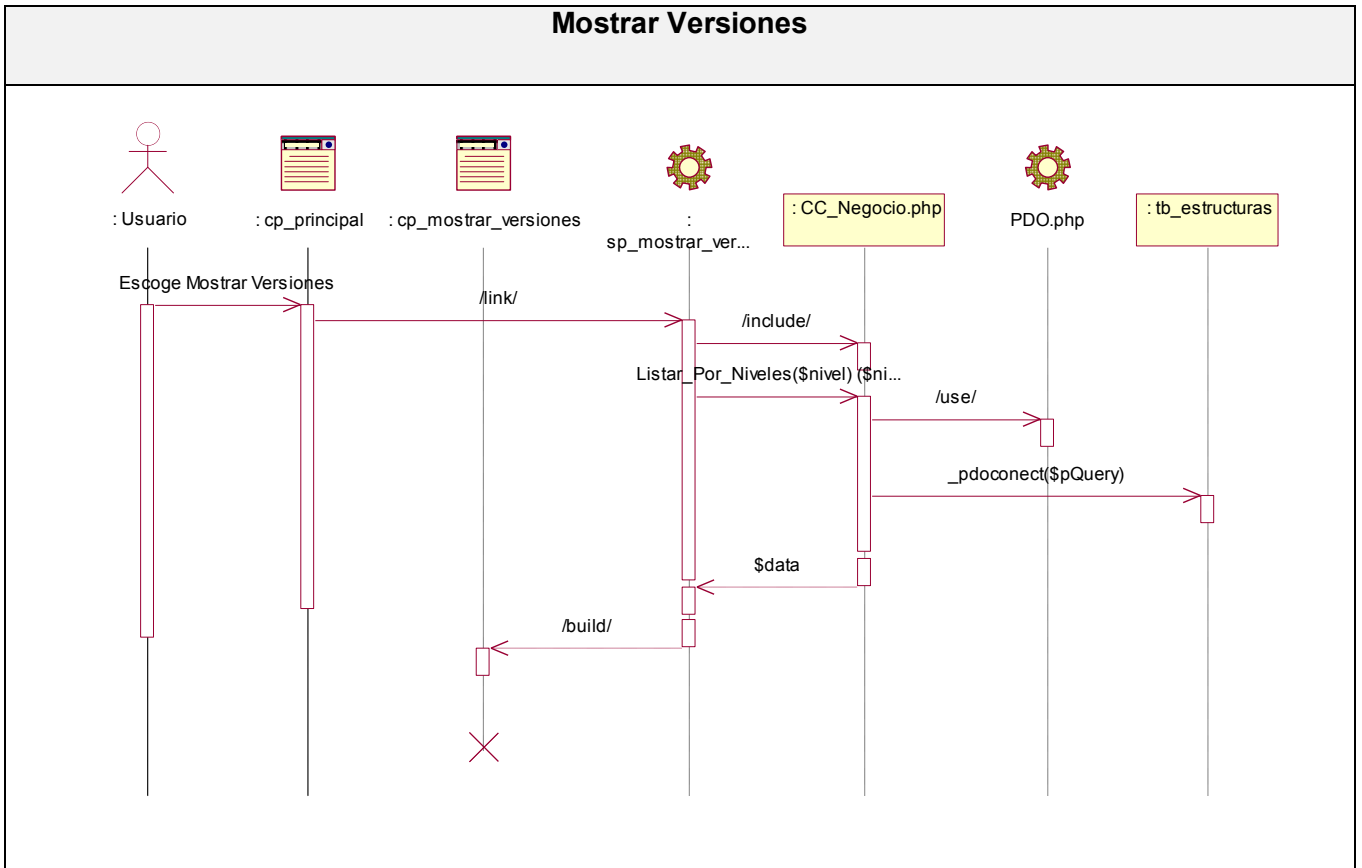


Figura 3. 14 Diagrama de Secuencia "Mostrar Versiones".

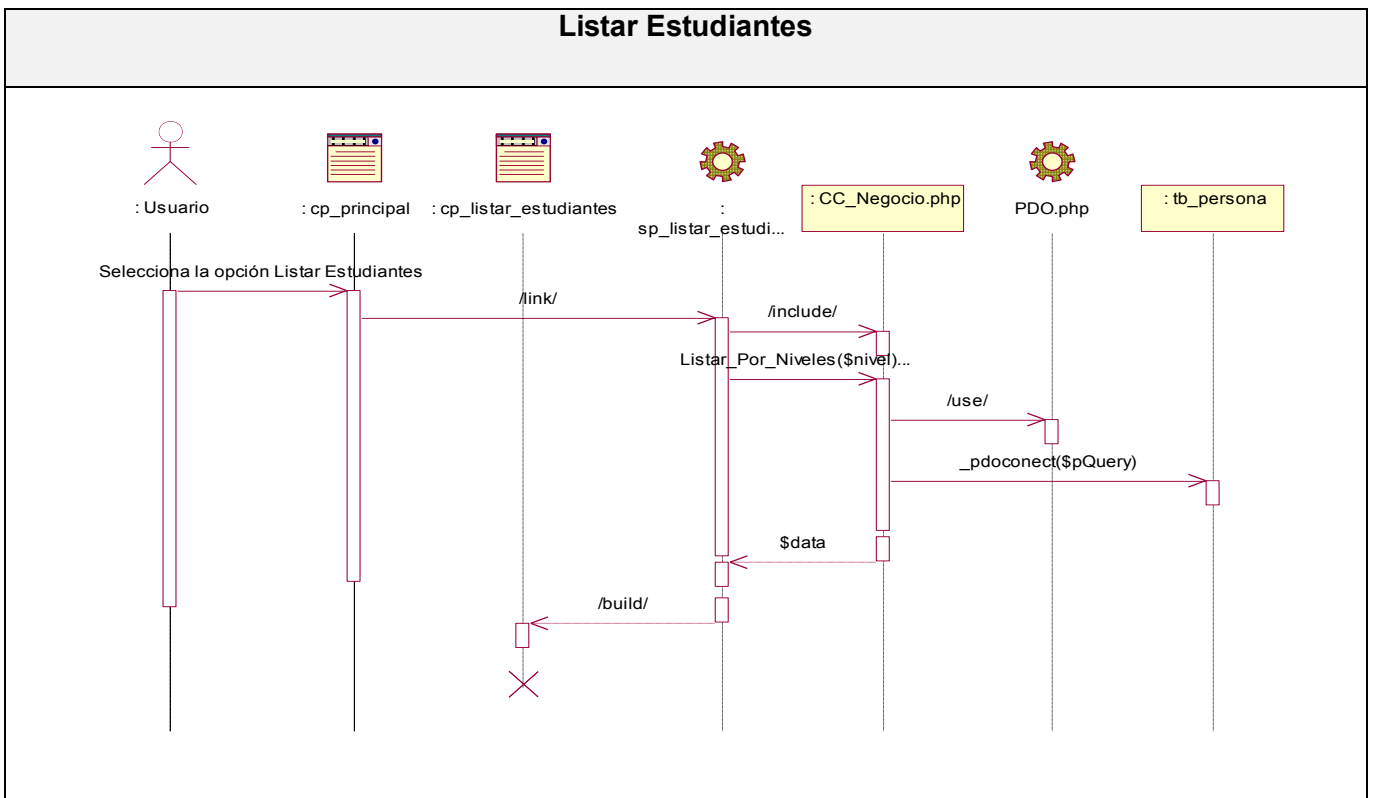


Figura 3. 15 Diagrama de Secuencia "Listar Estudiantes".

Conclusiones.

Con la elaboración del capítulo se obtuvo el Modelo de Análisis y de Diseño, de esta forma se describe la realización física de los casos de usos. Además, se tiene en cuenta como los requisitos funcionales y no funcionales; y otras restricciones relacionadas con la implementación impactan en el sistema desarrollado. Asimismo los resultados obtenidos, dan entrada a la realización del Flujo de Trabajo de Implementación.

CAPÍTULO 4: IMPLEMENTACIÓN.

Introducción.

En la realización del presente capítulo se tratan temas como la implementación y construcción de la aplicación partiendo del diseño antes elaborado. Además, se modela el diagrama de componente y el modelo de despliegue que se tiene en el sistema.

4.1 Modelo de Implementación.

El Modelo 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.

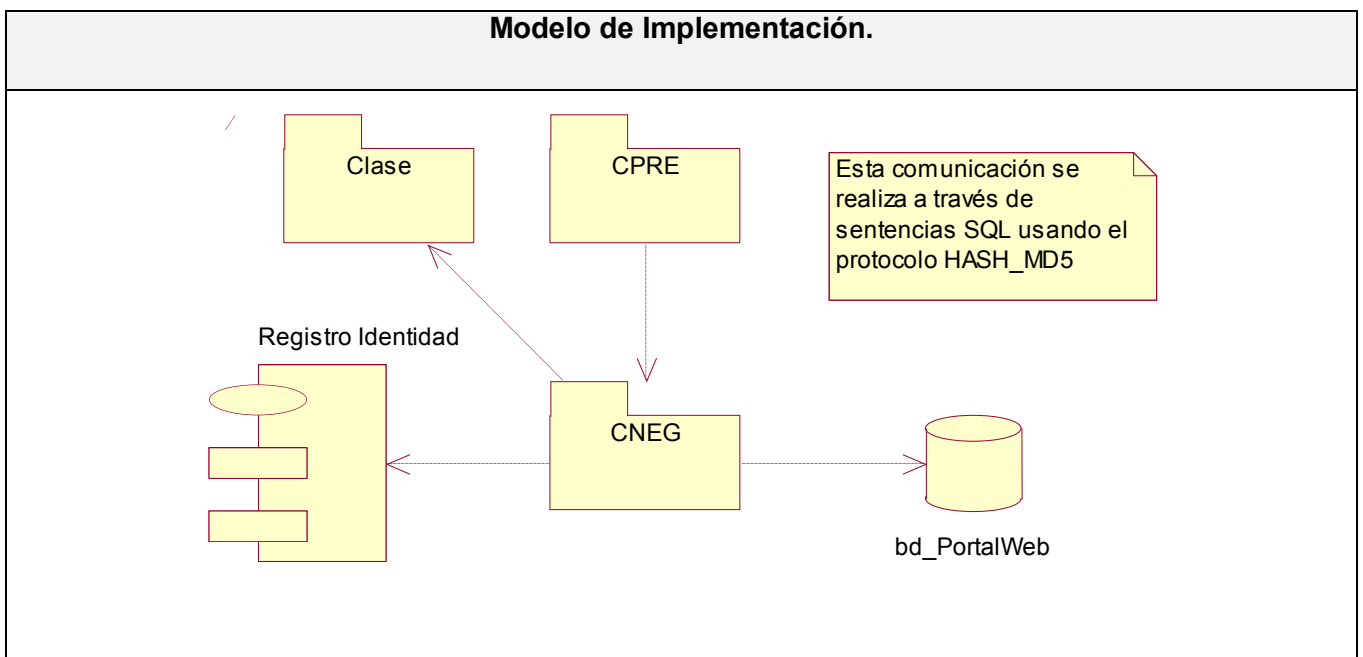


Figura 4. 1 Modelo de Implementación.

4.1.1 Diagrama de Componente.

Estos diagramas son los encargados de describir los elementos físicos del sistema y sus relaciones. Los componentes representan todos los tipos de software que tiene la aplicación. Es donde se presentan las opciones de ejecución conteniendo código fuente, binario y ejecutable. Se utilizan las relaciones de dependencia en dichos diagramas para especificar que un componente utiliza los servicios por otro componente.

Los diagramas de componentes son usados para estructurar el modelo de implementación en términos de subsistemas de implementación y mostrar las relaciones entre los elementos de

implementación. Un subsistema de implementación es una colección de componentes y otros subsistemas de implementación usados para estructurar el modelo de implementación y dividirlos en pequeñas partes que pueden ser integradas y probadas de forma separada.

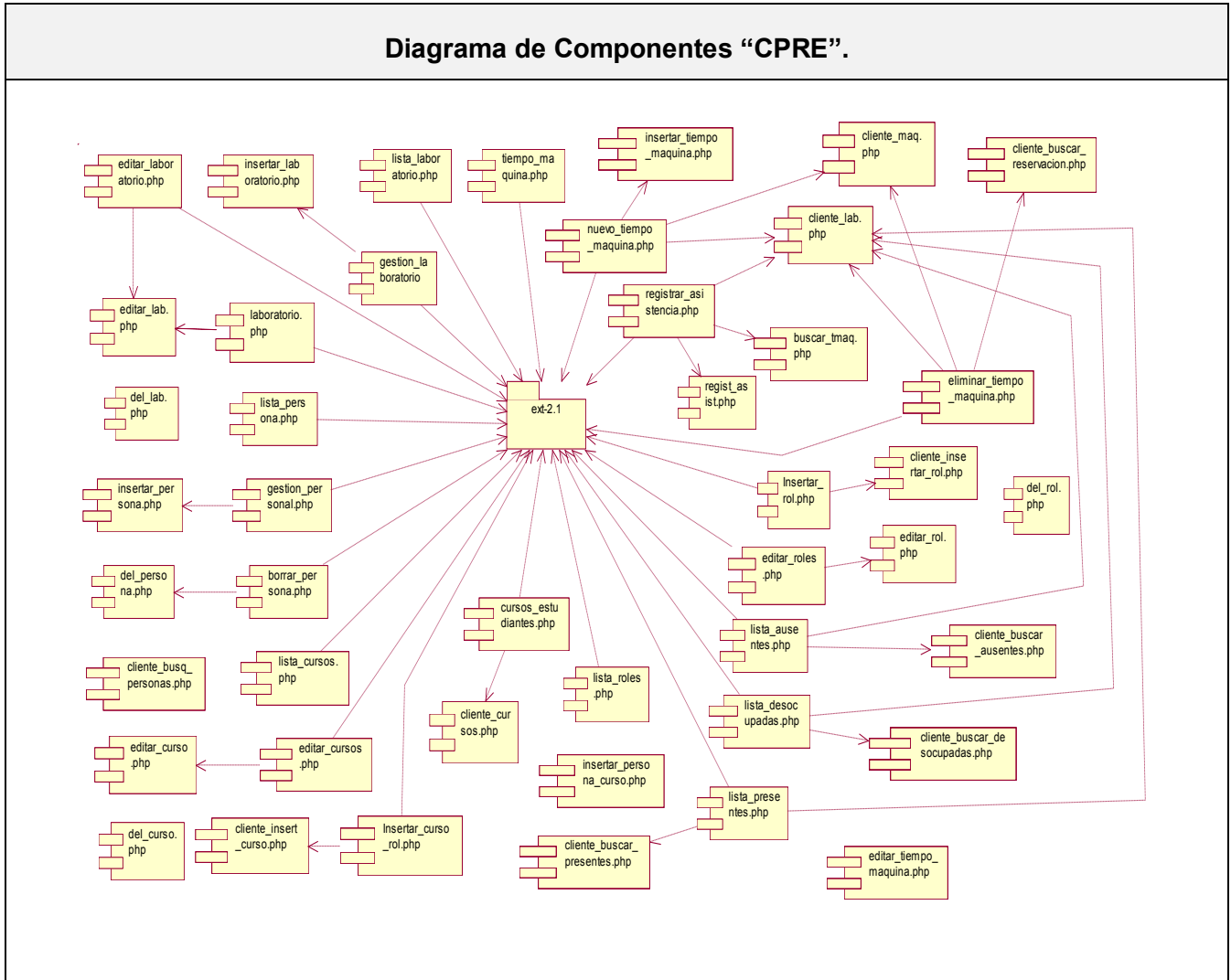


Figura 4. 2 Diagrama de Componente.

4.1.2 Modelo de Despliegue.

En un modelo de despliegue es donde se muestra la disposición física de los nodos existentes que conforman el sistema y el reparto de los componentes sobre dichos nodos. Un nodo es un recurso de ejecución un ejemplo de esto es: un computador, un dispositivo o una memoria.

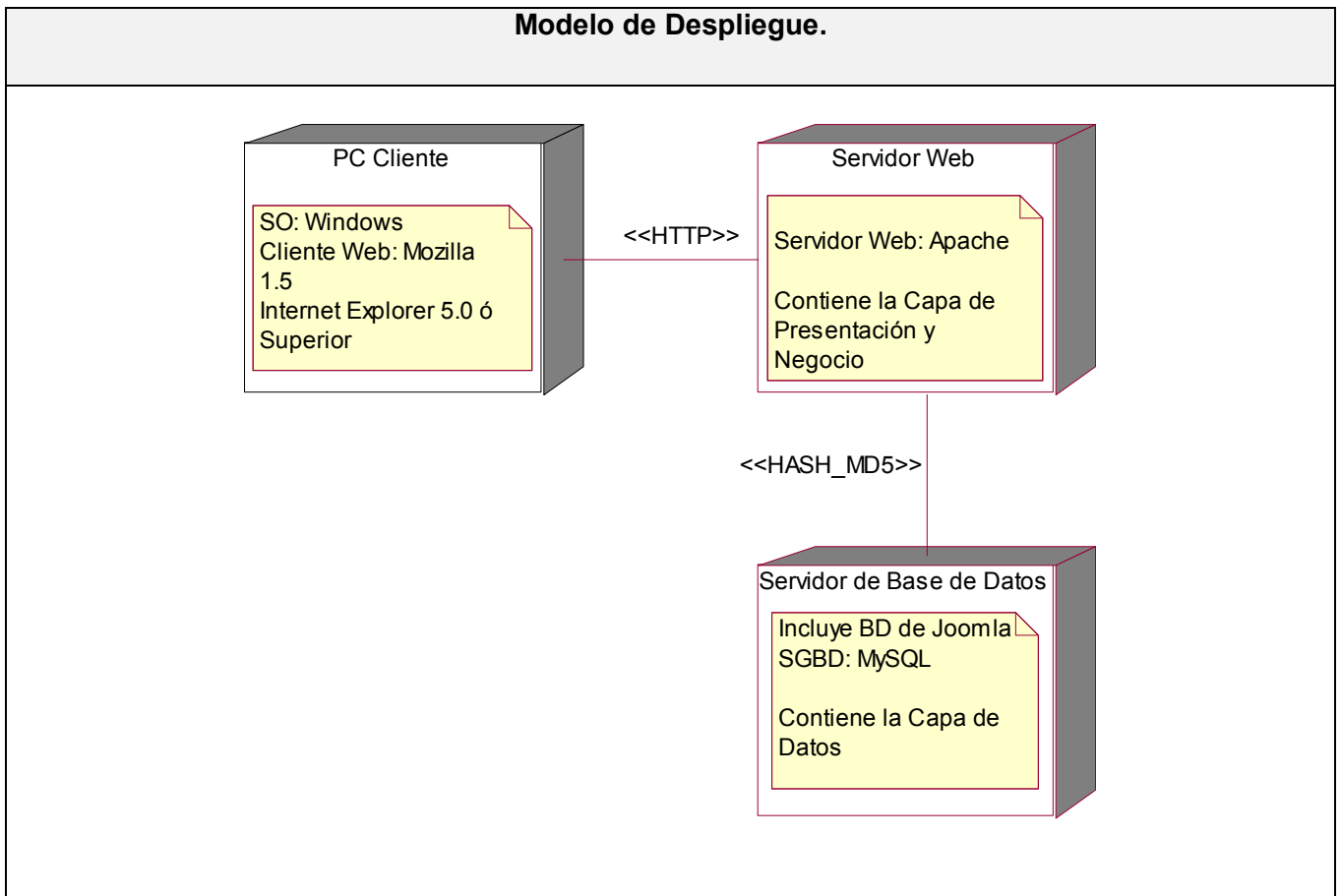


Figura 4. 3 Modelo de Despliegue.

4.2 Descripción de los métodos del negocio.

A continuación se presenta la descripción de algunos de los métodos más complejos para la elaboración del Portal Web de APS donde se utilizaron componentes del framework ext-js para la realización de la interfaz visual de dichos métodos.

insertar_tiempo_maquina

El usuario entra todos los datos necesarios por la interfaz, éstos son: laboratorio, máquina, solapín, tipo de semana, día y sección. Estos datos son capturados y se comprueba mediante una consulta a la base de datos si el solapín pasado por el usuario pertenece a alguna persona que se haya insertado anteriormente mediante el método insertar persona. Luego, si se encuentra a la persona se devuelve el id de la misma, de lo contrario, el método devuelve falso. Una vez realizado este procedimiento, se da paso a insertar el tiempo de máquina mediante un método llamado AddTiempoMaquina el cual registra en la base de datos los parámetros mencionados inicialmente.

registrar_asistencia

Primeramente se realiza una búsqueda de todos los tiempos de máquinas registrados para un laboratorio. Dicha búsqueda se hace teniendo en cuenta parámetros como el laboratorio, la semana,

la sección y el día. Para esta búsqueda se ejecuta el método `BuscarTiempoMaquina` el cual mediante una consulta a la Base de Datos devuelve todos los tiempos de máquinas encontrados que cumplan con los criterios anteriores. Luego se almacenan todos estos datos menos el nombre de la persona a la que corresponde el tiempo de máquina en un arreglo llamado `resultado`, dicho nombre viene como un `id` por lo cual se hace necesario hacer un ciclo anidado, o sea, un `foreach` dentro de otro `foreach` para poder desglosar este `id` y obtener el nombre.

En este momento se procede a guardar este nombre en el arreglo inicial, conjuntamente con todos los otros datos del tiempo de máquina y se muestra en la interfaz los datos que conforman el tiempo de máquina de forma completa, ya que tenemos el nombre de la persona a la cual corresponde el mismo.

Una vez que el usuario puede ver el listado de los tiempos de máquina con los respectivos nombres de quienes los ocupan. Se puede comenzar a pasar asistencia marcando en cada elemento si el estudiante está presente o ausente y si la computadora está o no vacía. Esto se hace mediante una llamada al método `regist_asist` lo cual es un diminutivo de registrar asistencia, donde se comienza verificando de qué forma han sido marcados los elementos. Esto se logra utilizando la función `explode` para separar la información seleccionada en el `radio button` y el `id` de la persona. Luego en un arreglo llamado `asis guardo` dicho `id` y si se marcó como ausente, presente o vacía utilizando para esto los valores `-1`, `0` y `1` respectivamente.

Cuando este procedimiento se haya realizado para todos los tiempos de máquinas encontrados. Se guarda en otro arreglo llamado `Asistencia` el cual debe insertarse en la Base de Datos mediante un procedimiento. En medio de toda esta ejecución, se declara un contador llamado `cont` el cual se inicializa en `0` y se incrementa si ocurre un error al insertar. Al final si el contador es distinto de `0` se dispara un cartel de error, de lo contrario significa que los datos han sido insertados correctamente.

estudiantes_presentes

El usuario introduce en la interfaz correspondiente el laboratorio y la fecha, lo cual es capturado con el método `BuscarPresentes` que tiene como fin, hacer una consulta a la Base de Datos y buscar allí mediante un procedimiento los estudiantes que estuvieron presentes en la fecha dada. Estos datos son devueltos y asignados a un arreglo llamado `resultado`, pero la persona viene como un `id` cuando realmente lo que necesitamos es el nombre. Por esto se utilizan los servicios del UDDI haciendo una llamada al método `ObtenerPersonaDadoldPersonaResult` para obtener verdaderamente el nombre que necesitamos. Luego estos datos ya con el nombre se van guardando en el arreglo llamado `devolver` y al final el `resultado`; que no es más que la información que contiene el nombre de la persona que estuvo presente en una fecha dada, se muestra en la interfaz.

Conclusiones.

Con la realización del capítulo se elaboraron los distintos modelos del flujo de trabajo de implementación, y las diversas pautas seguidas para el diseño de interfaz y para los métodos usados en los principios de codificación. Además, se brindó de forma breve y comprensible la descripción de algunos de los métodos implementados como son: Insertar un nuevo tiempo de máquina, registrar asistencia y un el reporte de los estudiantes presentes dado una fecha. Se pretende también dejar una idea clara de la fase de implementación y de todo lo que esto conlleva.

CONCLUSIONES.

Con la realización del presente trabajo se hizo un estudio del estado del arte de las herramientas de gestión de proyectos, control de versiones y portales web. También, se describieron y se emplearon las tecnologías necesarias para la implementación del Portal Web de APS con el fin de unificar las herramientas utilizadas en el mismo.

Finalmente se construyó el Portal Web que permite que los estudiantes y profesores del proyecto puedan acceder a la información relacionada con la producción. Así como llevar un registro más eficaz de los datos necesarios para el buen funcionamiento del mismo.

Para el desarrollo del sistema se utilizó el CMS Joomla con PHP como lenguaje de programación y MySQL como Sistema Gestor de Base de Datos.

La solución propuesta brindará rapidez en los mecanismos empleados por el Proyecto APS para llevar el control de los estudiantes y mejorar su organización. Superará los obstáculos físicos que limitan y demoran a los profesores y jefes del proyecto mantener un adecuado control de determinadas estadísticas como son: datos de los integrantes, reportes de asistencia al horario de producción, planificación y control del tiempo de máquina, entre otros.

RECOMENDACIONES.

Se recomienda:

- Desplegar el portal para su uso en el proyecto APS.
- Definir el plan de instrucción necesario para que el personal del proyecto APS, en especial los profesores, se familiaricen con el sistema.
- Mejorar el diseño del Portal.
- Migrar en un futuro hacia otras versiones del CMS utilizado (Joomla), que incorporan nuevas funcionalidades que la versión utilizada no posee.
- Adicionar funcionalidades al Portal de forma que todo lo que se gestione en el proyecto de forma manual pueda hacerse mediante el uso de las TIC.
- Agregar niveles de acceso de forma tal, que no solo existan usuarios públicos, registrados y administradores.

BIBLIOGRAFÍA

1. Wikipedia. [En línea] [Citado el: 19 de Junio de 2008.]
http://es.wikipedia.org/wiki/Portal_%28internet%29.
2. abcdelinternet. *Diccionario del Internet - Portales*. [En línea] [Citado el: 5 de Febrero de 2008.]
<http://www.abcdelinternet.com/abc1.htm>.
3. **Teruel, Prof. Alejandro**. lcd. *Arquitectura de capas*. [En línea] [Citado el: 17 de Mayo de 2008.]
<http://www ldc.usb.ve/~teruel/ci3715/clases/arqCapas.html>.
4. mitecnologico. *Arquitectura Aplicaciones Web*. [En línea] [Citado el: 25 de Junio de 2008.]
<http://www.mitecnologico.com/Main/ArquitecturaAplicacionesWeb>.
5. joomlaos. *¿Qué es Joomla?* [En línea] [Citado el: 10 de Abril de 2008.] <http://www.joomlaos.net/-que-es-joomla--4.php>.
6. joomlaos. *Características de Joomla*. [En línea] [Citado el: 10 de Abril de 2008.]
<http://www.joomlaos.net/caracteristicas-de-joomla.php>.
7. Wikipedia. *Lenguajes de Programación (PHP)*. [En línea] [Citado el: 8 de Febrero de 2008.]
<http://es.wikipedia.org/wiki/PHP>.
8. masadelante. *¿Qué significa php? - Definición de php*. [En línea] [Citado el: 8 de Febrero de 2008.]
<http://www.masadelante.com/faq-php.htm>.
9. desarrolloweb. *Java Script*. [En línea] [Citado el: 20 de Marzo de 2008.]
<http://www.desarrolloweb.com/articulos/490.php>.
10. dcc. *XML: ¿Otro Acrónimo Mas?* [En línea] [Citado el: 12 de Mayo de 2008.]
<http://www.dcc.uchile.cl/~rbaeza/inf/xml.html>.
11. **Franco, Jose Manuel Lopez**. trevinca. *Características XML*. [En línea] 15 de Octubre de 2001.
[Citado el: 2 de Julio de 2008.]
<http://trevinca.ei.uvigo.es/~txapi/espanol/proyecto/superior/memoria/node156.html> .
12. **Ortíz, Dr. Manuel Martín**. *Aplicaciones de Bases de Datos Cliente Servidor*. [En línea] Marzo de 2004. http://www.cs.buap.mx/~mmartin/notas/BD_CS2004_v3.pdf.
13. *EXT-Framework JavaScript*. [En línea] [Citado el: 12 de Mayo de 2008.] <http://pixelco.us/blog/ext-framework-javascript>.
14. **Esteban, Ginés Ángel, Grimaldos, José J. y Saorín, Antonio**. tldp. *Sirviendo web desde la escuela*. [En línea] [Citado el: 12 de Mayo de 2008.] <http://es.tldp.org/Tutoriales/doc-servir-web-escuela/sirviendo-web-escuela.pdf>.
15. Wikipedia. [En línea] [Citado el: 30 de Noviembre de 2007.]
http://es.wikipedia.org/wiki/Sistema_de_gesti%C3%B3n_de_base_de_datos#Objetivos.
16. **Suárez Bárzaga, Liusvani y Lorenzo, Leinier Lubián**. *Diseño y Servicios Web para el Registro de Enfermedades de Declaración Obligatoria del Sistema de Información para la Salud*. Habana . 2006 - 2007.

17. rational. *Rational Rose Enterprise*. [En línea] [Citado el: 2 de Julio de 2008.]
<http://www.rational.com.ar/herramientas/roseenterprise.html>.
18. **Autores, Colectivo de.** *Proyecto Vulcano Forja de proyectos software de calidad*. 2007.
19. **PRESSMAN, R.** *Ingeniería de Software, un enfoque practico, parte 1*. La Habana : Felix Varela, 2004.
20. —. *Ingeniería de Software, un enfoque practico, parte 2*. La Habana : Felix Varela, 2004.
21. **Jacobson, I., Booch, G y Rumbaugh, J.** *El Proceso Unificado de Desarrollo de Software*. s.l. : Addison-Wesley, 2000.

REFERENCIAS BIBLIOGRÁFICAS.

- [1]. Wikipedia. [En línea] [Citado el: 19 de Junio de 2008.] http://es.wikipedia.org/wiki/Portal_%28internet%29.
- [2]. abcdelinternet. *Diccionario del Internet - Portales*. [En línea] [Citado el: 5 de Febrero de 2008.] <http://www.abcdelinternet.com/abc1.htm>.
- [3]. **Teruel, Prof. Alejandro**. Icd. *Arquitectura de capas*. [En línea] [Citado el: 17 de Mayo de 2008.] <http://www ldc.usb.ve/~teruel/ci3715/clases/arqCapas.html>.
- [4]. mitecnologico. *Arquitectura Aplicaciones Web*. [En línea] [Citado el: 25 de Junio de 2008.] <http://www.mitecnologico.com/Main/ArquitecturaAplicacionesWeb>.
- [5]. joomlaos. *¿Qué es Joomla?* [En línea] [Citado el: 10 de Abril de 2008.] <http://www.joomlaos.net/que-es-joomla--4.php>.
- [6]. joomlaos. *Características de Joomla*. [En línea] [Citado el: 10 de Abril de 2008.] <http://www.joomlaos.net/caracteristicas-de-joomla.php>.
- [7]. Wikipedia. *Lenguajes de Programación (PHP)*. [En línea] [Citado el: 8 de Febrero de 2008.] <http://es.wikipedia.org/wiki/PHP>.
- [8]. masadelante. *¿Qué significa php? - Definición de php*. [En línea] [Citado el: 8 de Febrero de 2008.] <http://www.masadelante.com/faq-php.htm>.
- [9]. desarrolloweb. *Java Script*. [En línea] [Citado el: 20 de Marzo de 2008.] <http://www.desarrolloweb.com/articulos/490.php>.
- [10]. dcc. *XML: ¿Otro Acrónimo Mas?* [En línea] [Citado el: 12 de Mayo de 2008.] <http://www.dcc.uchile.cl/~rbaeza/inf/xml.html>.
- [11]. **Franco, Jose Manuel Lopez**. trevinca. *Características XML*. [En línea] 15 de Octubre de 2001. [Citado el: 2 de Julio de 2008.] <http://trevinca.ei.uvigo.es/~txapi/espanol/proyecto/superior/memoria/node156.html> .
- [12]. **Ortíz, Dr. Manuel Martín**. *Aplicaciones de Bases de Datos Cliente Servidor*. [En línea] Marzo de 2004. http://www.cs.buap.mx/~mmartin/notas/BD_CS2004_v3.pdf.
- [13]. *EXT-Framework JavaScript*. [En línea] [Citado el: 12 de Mayo de 2008.] <http://pixelco.us/blog/ext-framework-javascript>.
- [14]. **Esteban, Ginés Ángel, Grimaldos, José J. y Saorín, Antonio**. tldp. *Sirviendo web desde la escuela*. [En línea] [Citado el: 12 de Mayo de 2008.] <http://es.tldp.org/Tutoriales/doc-servir-web-escuela/sirviendo-web-escuela.pdf>.
- [15]. Wikipedia. [En línea] [Citado el: 30 de Noviembre de 2007.] http://es.wikipedia.org/wiki/Sistema_de_gesti%C3%B3n_de_base_de_datos#Objetivos.
- [16]. **Suárez Bárzaga, Liusvani y Lorenzo, Leinier Lubián**. *Diseño y Servicios Web para el Registro de Enfermedades de Declaración Obligatoria del Sistema de Información para la Salud. Habana* . 2006 - 2007.
- [17]. Ídem a la referencia 16.

[18]. Ídem a la referencia 16.

[19]. rational. *Rational Rose Enterprise*. [En línea] [Citado el: 2 de Julio de 2008.]
<http://www.rational.com.ar/herramientas/roseenterprise.html>.

[20]. **Autores, Colectivo de.** *Proyecto Vulcano Forja de proyectos software de calidad*. 2007.

[21]. Ídem a la referencia 20.

[22]. Ídem a la referencia 20.

[23]. Ídem a la referencia 20.

[24]. Ídem a la referencia 20.

GLOSARIO DE TÉRMINOS.

Acoplamiento: Es una medida de la interdependencia relativa entre los componentes. Minimizando el acoplamiento se evita el efecto “onda” en la propagación de errores.

Aplicación Web: Es una aplicación informática que los usuarios utilizan accediendo a un servidor Web a través de un navegador o browser. Estas son muy populares debido a la habilidad para actualizar y mantener la información manipulada sin distribuir e instalar el software en miles de potenciales clientes.

APS: Atención Primaria de Salud es el nivel asistencial que constituye la puerta de entrada del paciente al Sistema Nacional de Salud, donde debe darse solución alrededor del 90% de los problemas que afectan a la población. En este nivel se realizan acciones educativas, curativas y de Rehabilitación.

Beans: Componente de software que tiene la particularidad de ser reutilizable.

Bugs: Error indefinido.

Caso de Uso: Secuencias de acciones que el sistema puede llevar a cabo interactuando con sus actores, incluyendo alternativas dentro de las secuencias.

Caso de Uso arquitectónicamente significativos: Casos de uso que ayudan a mitigar los riesgos más importantes, aquellos que son los más importantes para los usuarios del sistema y aquellos que ayudan a cubrir todas las funcionalidades significativas.

Caso de uso que, dada las funcionalidades que agrupa, constituye pieza clave para la arquitectura del sistema. Esto puede estar dado por la dependencia que exista de otros casos de uso con él, que por su complejidad pueda constituir un riesgo para el sistema o por la cantidad de clases que puedan generarse del mismo.

CMS: En inglés Content Management System, abreviado (CMS), permite crear sitios web de alta interactividad, profesionalidad y eficiencia. Además de poder crear una estructura de soporte (framework) para la creación y administración de contenidos por parte de los participantes, principalmente en páginas web.

Cohesión: Es una medida de la fuerza relativa funcional de un componente. Un componente con cohesión realiza una sola tarea dentro de un procedimiento de software, requiriendo poca interacción con los otros componentes.

Informatizar: Proceso de aplicar sistemas o equipos informáticos al tratamiento de la información.

MINSAP: Ministerio de Salud Pública, órgano rector del Sistema Nacional de Salud.

PHP: El lenguaje PHP (acrónimo de "PHP: Hypertext Preprocessor") es un lenguaje de script interpretado utilizado para la generación de páginas Web dinámicas, embebido en páginas HTML y ejecutado en el servidor.

Soportado por una gran comunidad de desarrolladores, como producto de código abierto, PHP goza de la ayuda de un gran grupo de programadores, permitiendo que los fallos de funcionamiento se encuentren y reparen rápidamente. Con PHP se puede hacer el procesamiento de información en formularios, foros de discusión, manipulación de cookies y páginas dinámicas.

Plug-ins: Aditamento para agregar a un equipo.

Portal Web: Es un sitio web cuyo objetivo es ofrecer al usuario, de forma fácil e integrada, el acceso a una serie de recursos y de servicios, entre los que suelen encontrarse buscadores, foros, documentos, aplicaciones y compra electrónica.

Prototipo: Maqueta visual funcional o no de la futura aplicación. Este puede ser una imagen o una aplicación software que simule funcionalidades del software.

SISalud: Plataforma única para la gestión, procesamiento y transmisión de la información clínica en el Sistema Nacional de Salud.

Softel: Entidad del Ministerio de Informática y las Comunicaciones (MIC).