



**Universidad de las Ciencias Informáticas**  
**Facultad 10**



# **Sistema para el Control de Asistencia integrado al CMS Drupal**

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

**Autora:** Mirelys Torres Pérez

**Tutor:** Ing. Yoel González Díaz

**Co-tutor:** Ing. José Ramón Fernández Pérez

**Consultante:** DraC. Mireya Pérez Rodríguez

Ciudad de La Habana, 2010

“Año 50 de la Revolución”

## RESUMEN

En el actual siglo, cualquier actividad laboral donde esté presente la entrega de resultados y su evaluación, las personas vinculadas deben estar reguladas en cuanto a distribución de puestos de trabajo, horarios de entrada y salida, roles, evaluaciones y otras características propias de cada centro evaluador.

La Universidad de las Ciencias Informáticas (UCI) es una universidad productiva, cuya misión es producir software y servicios informáticos a partir de la vinculación estudio – trabajo como modelo de formación. Específicamente en el Departamento de Gestión y Representación de Contenidos Web (GRCW) de la facultad 10 de esta Universidad, se lleva a cabo un proceso donde trabajan alrededor de 130 estudiantes divididos en horarios de desarrollo de software con los cuales deben cumplir.

Debido a las deficiencias que conlleva consigo para el personal que trabaja en dicho Departamento, el hecho de que el control de la asistencia en este se lleve a cabo de manera manual o por simple observación. La presente investigación, consiste en el diseño e implementación de un módulo para el Sistema de Gestión de Contenidos (CMS) Drupal, que permite controlar los elementos de asistencia, incidencias laborales y es capaz de emitir evaluaciones parciales de acuerdo a los registros de asistencia almacenados. Para cumplir con los objetivos trazados, el sistema brinda la posibilidad de definir las sesiones de producción del Departamento y de establecer un horario de producción semanal (teniendo en cuenta las sesiones creadas) para cada estudiante del mismo. Así como de generar reportes en base a los registros de asistencia almacenados.

## ÍNDICE

CAPÍTULO I: FUNDAMENTOS TEÓRICOS DE LOS SISTEMAS ACTUALES PARA CONTROLAR LA ASISTENCIA E INCIDENCIAS LABORALES.....	7
1.1.    Sistemas de control de asistencia existentes.....	8
1.1.1.    Sistemas automatizados empleados en el mundo .....	9
1.1.2.    Sistemas automatizados utilizados en Cuba.....	18
1.1.3.    Sistemas automatizados empleados en la UCI.....	20
1.2.    Tendencia y tecnología actuales para el desarrollo de aplicaciones web. ....	23
1.2.1.    Lenguajes del lado del cliente.....	23
1.2.2.    Lenguajes del lado del servidor .....	24
1.2.3.    Gestores de Bases de Datos .....	24
1.2.4.    Los Sistemas de Gestión de Contenidos .....	25
1.3.    Metodologías de desarrollo de software.....	27
1.4.    Herramientas para el desarrollo del software. ....	30
CAPÍTULO II: CARACTERÍSTICAS DEL SISTEMA PARA EL CONTROL DE ASISTENCIA INTEGRADO AL CMS DRUPAL. ....	33
2.1.    Problema y situación problemática .....	33
2.2.    Propuesta del Sistema.....	34
2.3.    Modelo de dominio .....	35
2.4.    Levantamiento de requisitos .....	36
2.4.1.    Requerimientos funcionales.....	37
2.4.2.    Requerimientos no funcionales del sistema .....	37
2.4.3.    Definición de los actores del sistema.....	39
2.4.4.    Diagrama de Casos de Uso del Sistema.....	40

2.4.5. Descripción de los Casos de Uso del Sistema .....	40
CAPÍTULO III: ANÁLISIS Y DISEÑO DEL SISTEMA PARA EL CONTROL DE ASISTENCIA.....	52
3.1. Modelo de Análisis .....	52
3.1.1. Diagramas de interacción .....	52
3.2. Modelo de Diseño.....	55
3.2.1. Diagrama de clases del diseño Web.....	55
3.2.2. Diseño de la Base de Datos.....	61
3.2.3. Diagrama de clases persistentes .....	62
3.2.4. Modelo de datos de clases persistentes .....	62
CAPÍTULO IV: CONSTRUCCIÓN Y PRUEBA DE LA SOLUCIÓN.....	64
4.1. Diagrama de Componentes .....	64
4.2. Diagrama de Despliegue .....	65
4.3. Arquitectura y Patrones empleados (56) (57).....	66
4.4. Pruebas .....	68
CONCLUSIONES .....	72
RECOMENDACIONES .....	73
BIBLIOGRAFÍA.....	<b>¡Error! Marcador no definido.</b>

## INTRODUCCIÓN

Desde sus inicios el hombre se percató que su supervivencia dependía de la capacidad de adaptación a las condiciones que el entorno le imponía a un ritmo vertiginoso y que para poder responder a estos retos debía organizarse y controlar los recursos de que disponía. Cada vez con mayor frecuencia la humanidad se ha visto desafiada por la naturaleza y ha debido resolver los retos que esta le ha impuesto en cada ocasión con un menor plazo de tiempo. En la actualidad las investigaciones se incrementan tanto en número como en intensidad y el ritmo al que lo hacen, provoca que lo surgido recientemente, hasta ayer novedoso, sea hoy atrasado.

El éxito o fracaso de las entidades radica en el manejo del cambio que significa también el de las prioridades y las formas de movilización de los recursos humanos, técnicos y financieros. El hombre es llamado a configurar las nuevas organizaciones, buscar múltiples alternativas, reestructurar permanentemente, actuar preventivamente y ajustar las desviaciones.

En el actual siglo cualquier actividad laboral donde esté presente la entrega de resultados y su evaluación, las personas vinculadas deben estar reguladas en cuanto a distribución de puestos de trabajo, horarios de entrada y salida, roles, evaluaciones y otras características propias de cada centro evaluador.

El campo de la informática no está ajeno a este proceso, esto se ha evidenciado en diferentes países del planeta con el desarrollo de aplicaciones de escritorio y sistemas web, (1) (2) (3) que ayudan a las diferentes empresas productoras de software a tener un mayor control del trabajo de su personal calificado, de la distribución correcta de los medios tecnológicos y una supervisión más eficiente de todo lo que se desarrolla en estos centros. De acuerdo con el comportamiento mantenido en la asistencia a los horarios definidos para la producción e investigación y al rendimiento alcanzado por efectos de entrega de soluciones de software, se pueden emitir valoraciones cualitativas y cuantitativas.

En Cuba se usan software como: GREHU y FASTOS para gestionar los recursos humanos de instalaciones de CUBANACAN S.A, la Cadena Hotelera Gran Caribe, Softel y Desoft (4) (5). Estas herramientas proporcionan entre otras funcionalidades el control de asistencia de los trabajadores y múltiples reportes relacionados con su actividad.

La Universidad de las Ciencias Informáticas (UCI) es una universidad productiva, cuya misión es producir software y servicios informáticos a partir de la vinculación estudio – trabajo como modelo de formación.

La UCI cuenta con una Infraestructura Productiva formada por 16 centros que a su vez están compuestos por Departamentos ubicados en sus 10 Facultades (6). El Centro Gestión de Información y Tecnologías

Libres (GEITEL) de la Facultad 10, está compuesto por cuatro Departamentos: Sistema Operativo y Desarrollo de Tecnologías Libres (SODTL), Gestión Documental (GD), Soluciones Informáticas para Internet (SINI) y Gestión y Representación de Contenidos Web (GRCW).

Específicamente en el Departamento de GRCW se lleva a cabo un proceso, donde trabajan alrededor de 130 estudiantes divididos en horarios de desarrollo de software con los cuales deben cumplir. Actualmente el control del horario en este Departamento se lleva a cabo de manera manual o por simple observación, lo que provoca la toma de decisiones erróneas con los estudiantes a la hora de emitir una nota correspondiente a la asignatura Práctica Profesional. Además, dado un proceso de determinación del uso real de la tecnología, ya sea por horas de consumo eléctrico o aprovechamiento del tiempo de estudio individual, que constituyen elementos subjetivos importantes, no se reúnen los datos exactos debido a que no se cuenta con la estadística o el reporte de cuántas horas el estudiante trabaja en el laboratorio en un periodo determinado.

A partir de la necesidad social antes expresada, la presente investigación contribuye a solucionar el siguiente **problema científico**: ¿Cómo gestionar el control de asistencia vinculado a la producción de los estudiantes del Departamento de Gestión y Representación de Contenidos Web?

El diseño de la investigación permite a los autores declarar como **objeto de estudio**: Los sistemas de control de asistencia e incidencias laborales y se precisa como **campo de acción**: Drupal como Sistema de Gestión de Contenidos para controlar la asistencia e incidencias laborales.

En el contexto investigativo se define como **objetivo general**: Desarrollar un módulo de Drupal para el control de asistencia vinculado a la producción de los estudiantes del Departamento de Gestión y Representación de Contenidos Web.

Se **defiende la idea**, que la creación de un módulo de Drupal que gestione el control de asistencia vinculado a la actividad de desarrollo de software de los estudiantes del Departamento de Gestión y Representación de Contenidos Web, posibilitará a los directivos un adecuado manejo de criterios cuantitativos y cualitativos acerca del aprovechamiento de los recursos, tiempo de trabajo y evaluación de los estudiantes.

#### **Objetivos específicos:**

- Caracterizar el desarrollo de aplicaciones para el control de parámetros de horarios e incidencias laborales para entornos de escritorio y web.

- Implementar un módulo para Drupal que controle los elementos de asistencia, incidencias laborales y sea capaz de emitir evaluaciones parciales.
- Validar el sistema en el Departamento al cual pertenece este Trabajo de Diploma.

Para guiar el proceso investigativo se determinaron las siguientes **tareas científicas de la investigación**:

- Análisis de los sistemas actuales destinados al control de asistencia e incidencias laborales.
- Selección de una metodología apropiada para definir el desarrollo de la aplicación y que asegure su funcionamiento correcto.
- Levantamiento de requisitos.
- Implementación de las funcionalidades del sistema propuesto.
- Realización de pruebas a las distintas funcionalidades.

#### **Métodos y técnicas fundamentales de investigación utilizados.**

- **Método analítico-sintético:** se utilizó en la aplicación del conocimiento teórico referido en el marco conceptual en función de desarrollar el control de asistencia integrado al CMS Drupal.
- **Método dialéctico-materialista:** permitió desde el punto de vista filosófico, materialista y dialéctico conocer diferentes consideraciones de investigadores reconocidos en el tema con el fin de reconstruir el marco teórico conceptual a partir de conocimiento científico y precisar elementos relacionados con la importancia y novedad del problema investigado.
- **Método de inducción-deducción:** de importancia para la selección de una metodología apropiada para definir el desarrollo de la aplicación y asegurar su correcto funcionamiento y en la organización de los elementos del sistema para el control de asistencia integrado al CMS Drupal.
- **Método histórico-lógico:** en la determinación de los antecedentes del problema objeto de estudio así como en la determinación de las leyes generales que intervienen en el funcionamiento y desarrollo de los sistemas de control de asistencia.
- **Método sistémico-estructural:** en la modelación del objeto investigado y en el desarrollo del sistema de control de asistencia que se integre al CMS Drupal a utilizar en el Departamento de Gestión y Representación de Contenidos Web.
- **Método de la modelación:** en la elaboración del modelo teórico para el control de la asistencia y evaluación.

También fue utilizada la técnica de entrevista, para conocer y precisar criterios en vista al desarrollo del

Sistema para el Control de Asistencia.

El presente documento está estructurado, además de la presente Introducción, en los siguientes capítulos:

- **Capítulo I:** “Fundamentos teóricos de los sistemas actuales para controlar la asistencia e incidencias laborales”, donde se realiza un análisis de sistemas actuales destinados al control de asistencia e incidencias laborales. Se detallan las tendencias y tecnologías actuales, metodologías y herramientas para el desarrollo web, así como conceptos teóricos necesarios para comprender los temas tratados en el resto del documento.
- **Capítulo II:** “Características del Sistema para el Control de Asistencia integrado al CMS Drupal”, donde se exponen las características del sistema a implementar. Se determinan requisitos funcionales y no funcionales, actores, casos de uso del sistema y se realiza el modelo de dominio, de modo que facilite de manera visual mostrar al lector los principales conceptos que se manejan en el dominio del sistema en desarrollo.
- **Capítulo III:** “Análisis y diseño del Sistema para el Control de Asistencia”, donde se determinan las clases que se utilizarán para la implementación del sistema, la relación entre ellas, los diagramas de interacción y el diseño de la base de datos.
- **Capítulo IV:** “Construcción y prueba de la solución”, contempla la implementación del Sistema para el Control de Asistencia integrado al CMS Drupal, a partir de los diagramas de despliegue y de componentes. Se determinan y realizan las distintas pruebas a las funcionalidades.

Además se incluyen conclusiones, recomendaciones, bibliografía y anexos.



## **CAPÍTULO I: FUNDAMENTOS TEÓRICOS DE LOS SISTEMAS ACTUALES PARA CONTROLAR LA ASISTENCIA E INCIDENCIAS LABORALES.**

En el presente capítulo se abordan los fundamentos teóricos generales del proceso de control de asistencia, que sirven de punto de partida a la comprensión de la propuesta; junto a ello se realiza un análisis del estado del arte de los distintos sistemas para controlar la asistencia y su utilidad en las diferentes esferas en las que se emplean. Además, se detallan las tendencias y tecnologías actuales, metodologías y herramientas para el desarrollo web que darán solución al problema científico.

En la época actual donde el conocimiento es la base fundamental de cualquier fuente de competitividad en las organizaciones, cabe señalar que los conceptos actuales, juegan un papel estratégico fundamental.

### **¿Qué es un sistema de información?**

El concepto de sistema de información en el campo de la informática está muy ligado al de tecnología de información y existen diferentes concepciones acerca de la relación entre Tecnologías de Información (TI) y Sistemas de Información (SI), que pueden parecer sutiles; así para Cosialls (2000) las Tecnologías de Información son el conjunto de tecnologías (hardware, software, comunicaciones) aplicadas para implementar y dar soporte a los Sistemas de Información (7). El concepto de implementar se refiere a facilitar los instrumentos necesarios para la realización de alguna cosa (SALVAT, 1985) y la implementación, desde la perspectiva informática, implica la instalación y puesta en marcha de un sistema o conjunto de programas (ESPASA, 2000); relacionándose este último sólo con el funcionamiento. (8)

Para Barros (1998), el concepto de SI en computación se remonta a la década de los 70, haciendo énfasis en la integración de datos utilizando tecnología de sistemas de administración de bases de datos (SABD) y el apoyo a las decisiones de gestión (9). Este concepto persiste aun cuando ha evolucionado, dando lugar a concepciones de un mejor servicio a los usuarios con la disposición de información local en las computadoras para el apoyo en el control, evaluación y toma de decisiones, lo cual ofrece nuevas formas de realizar las actividades con el apoyo de las TI.

Para Cosialls (2000) un SI es un conjunto coherente de elementos relacionados, humanos, tecnológicos, organizativos y operativos, ordenados de alguna manera e interdependientes que constituyen, en una referencia temporal, una unidad funcional o un todo orgánico que captura (datos), procesa, almacena y distribuye información necesaria para el logro de un determinado propósito, que estará signado por el contexto donde se utilice. (7)

Los términos construcción, implementación, operación y soporte de los SI incluidos en el concepto de TI, permiten concebir la tecnología sólo como un elemento de los SI. Éstos se construyen con la utilización de aplicaciones (software) que se instalan en los elementos computacionales (hardware) para su operación; tal como afirman Cohen y Asín (2000) un sistema de información no necesariamente incluye un equipo electrónico y una aplicación computacional; pero no puede existir un SI sin tecnología, aunque sea en su forma más rudimentaria y manual (10). Estos autores definen un SI como un conjunto de elementos que interactúan entre sí con el fin de apoyar las actividades de una empresa o negocio, aplicable de hecho a cualquier organización o institución.

Para los efectos de este trabajo se asume, que un SI es un proceso que organiza elementos humanos y tecnológicos interdependientes, que interactúan entre sí con propósitos específicos de captura de datos internos y externos, su almacenamiento y procesamiento, para generar información, de modo que contribuyan al control, evaluación, planificación y toma de decisiones por los responsables de niveles gerenciales operativos y estratégicos, así como por clientes o beneficiarios de una determinada organización. (8)

#### **¿Qué es control de asistencia?**

Se entiende por control de asistencia a la inspección, vigilancia y aplicación de las medidas necesarias para el cumplimiento de las disposiciones establecidas en materia de asistencia del personal operativo de una entidad. (11)

#### **1.1. Sistemas de control de asistencia existentes.**

Empresas de todos los tamaños requieren contar con herramientas de gestión para controlar y cuantificar los distintos aspectos de su operación. Una de las áreas que más ha tardado en automatizarse ha sido la del Control de Asistencia y/o Permanencia del personal, debido en gran medida por la aún amplia utilización de sistemas tradicionales de control basados en los antiguos relojes mecánicos que utilizan tarjetas de cartón, o únicamente con “partes de asistencia” que no son otra cosa que hojas de papel con líneas numeradas en las que los empleados firman e indican (ellos mismos) la hora en la cual ingresan o salen de la empresa (12). Entre los instrumentos técnicos de registro y control del desarrollo del recurso humano más utilizados se encuentran los siguientes:

- La Ficha Personal.
- Ficha Social.

- Tarjeta de Control Diario de Asistencia.
- Parte Diario de Asistencia.
- Papeletas de Autorización de Salidas.
- Tarjetas de Control de Récord laboral.
- File Personal.
- Rol Vacacional.
- Rol de Cambio de Vigilancia.
- Cuadro de Asignación de Personal.
- Reglamento Interno de Traba.

Los Sistemas de Control de Asistencia modernos se basan en Tecnologías de Identificación Automática con Códigos de Barras, Banda Magnética, Tarjetas de Proximidad por radio frecuencia (RFID) e incluso Sistemas Biométricos de Huella Digital. Todos ellos solo forman una parte de la solución debido a que el componente fundamental es el Software de Control de Asistencia pues los datos capturados con los distintos modelos de lectores necesitan ser procesados para recién entonces llegar a convertirse en información (tardanzas, inasistencias, horas extras, etc.). (12) (13)

#### **1.1.1. Sistemas automatizados empleados en el mundo**

##### **E-time Software para Control de Tiempo y Asistencia (1) (14)**

Calcula automáticamente las horas trabajadas de sus empleados, así como sus tiempos extras, considerando horas de comida, ausentismos, vacaciones y retardos. Se encuentra disponible en internet \$16.95 (descuentos).

Características:

- Genera un archivo consolidado de pre Nómina, preconfigurado para su exportación a cualquier programa de nómina.
- Diseñado para resolver necesidades de operar de la empresa latinoamericana, a diferencia de los productos de origen estadounidense o europeo.
- Totalmente en español.
- Un traje a su medida: E-time puede ser modificado para necesidades específicas.

En un sistema típico de Gestión de Tiempo y Asistencia, los usuarios del mismo registran sus entradas y

salidas en terminales colectoras de datos (llamados también relojes electrónicos), usando tarjetas con código de barras, proximidad, banda magnética o dispositivos biométricos como lectores de huella digital y de geometría de mano.

Cada entrada o salida de un usuario, genera un registro que contiene, entre otras cosas:

- Fecha y hora en que se hizo el registro.
- Número de usuario o empleado.
- Número de la Terminal colectora donde se hizo el registro.
- Información adicional, por ejemplo el motivo de una salida (permiso sindical, permiso personal, etc.).

Estos registros son colectados a través de una red local, remota o vía Internet, por una computadora ejecutando E-time, quien tiene previamente en su propia base de datos, la información de todos los empleados, como es el horario y turno en que trabajan, su Departamento, fotografía digitalizada, etc.

A partir de los registros, se efectúan los cálculos de tiempo trabajando, tiempos extras, retardos, ausentismo, salidas a comer, salidas fuera de horario, y finalmente se genera un archivo de nómina, que contiene las variables que un programa de nómina requiere.

#### **Características relevantes de operación:**

- Rotación automática de turnos en base a una programación.
- Cambios globales de turnos de trabajo, por emergencias.
- Exportación de reportes a múltiples formatos (Excel, Word, CSV, sdf, etc.).
- Archivo de nómina con formato configurable.
- Archivo de registros de entradas y salidas (“chequeadas”) con formato configurable.
- Maneja hasta 10 clasificaciones del sistema (Ej., área, centro de costos, categorías, tipos de empleados) configurables por el usuario, las cuales se pueden agrupar hasta en cinco jerarquías.
- Programación de movimientos directamente desde la pantalla del Empleado.
- Manejo de fotografía digitalizada como parte de los datos de empleado.
- Módulo opcional integrado de diseño e impresión de credenciales, multiusuario y multiestación, usa la base de datos de E-time, y maneja impresoras de tarjetas de PVC o Poliéster Teslin. El gafete puede imprimirse directamente desde la pantalla de consulta de un empleado.
- Uso de motivos en autorizaciones de tiempo extra, bajas de empleados, incapacidades, permisos, suspensiones para reportes estadísticos.

- Seguridad a nivel módulo, acción (agregar, modificar y borrar), tiempo de inactividad y caducidad de contraseña.
- Módulo de comedor (consumo y platillos).
- Manejo de ligas entre supervisores y trabajadores.
- Almacena y administra las huellas digitales en equipos Bioscrypt a través del software V-Harvest.

#### **Características técnicas:**

- Totalmente 32 bits-opera bajo Windows 95, 98, NT.
- Arquitectura cliente-servidor.
- Gran capacidad de atención a usuarios simultáneos, haciéndolo ideal para empresas con múltiples localidades y usuarios en LAN.
- Opera con Microsoft SQL Server o MSDE (Microsoft Data Engine, motor de base de datos, compatible con SQL Server y actualizable a SQL Server).
- Su código fuente en Visual Basic, está disponible opcionalmente con costo adicional
- Servidor E-time:
  - Desconexión remota de clientes.
  - Envío de mensajes a clientes.
  - Procesos centralizados (actualización de datos, generación de archivos para validación).

#### **SICON Software Avanzado de Control de Asistencia (3)**

##### **Características:**

- **Acceso al sistema:** cada usuario ingresa al SICON mediante su contraseña, la cual le permite acceder únicamente a los módulos previamente autorizados por el administrador del sistema.
- **Consulta de marcadores y asistencias:** consulta de la asistencia de un empleado directamente en una sola pantalla, indicando la hora del mes actual o meses anteriores, desde aquí también se pueden realizar las justificaciones respectivas de un empleado.
- **Edición Manual de Marcadores:** a efectos de auditoría, las marcaciones ingresadas manualmente tienen un marcador para indicar que proceden de un ingreso manual del operador y no desde los lectores.
- **Programación de vacaciones y saldo vacacional:** controla los días solicitados a cuenta de vacaciones para el año en curso o a futuro. Vacacional indicando número de días disponibles de

tomar.

- **Impresión configurable de listados:** los reportes se imprimen directo a la pantalla, impresora o hacia archivo de texto ASCII.
- **Gestión de permisos y Movimientos:** impresión de listados indicando múltiples criterios de selección. SICON permite llevar un control de los distintos movimientos y permisos del empleado, permitiendo topes máximos para cada uno de ellos en el año indicando si son con o sin Goce.
- **Captura de datos:** lee los datos provenientes de cualquiera de los lectores que DESIGN SOFT S.A.C (empresa peruana formada en el año 2001) comercializa. Entre los dispositivos compatibles se encuentran: lectores de barras, de huellas digitales, 4030, DR519 y FP4500.
- **Sistema operativo:** Windows XP.

#### **Doce versión 2 (15)**

Es un programa de control de asistencia (US\$40.) que incorpora un registro de datos del personal de una empresa o institución. El control de asistencia se efectúa usando el teclado de una computadora en el cual el personal registra un código individual de la misma manera en que se podría hacer con un reloj tarjetero. Permite utilizar una cámara web para que tome fotografías de la persona que marca su entrada y salida de manera que, posteriormente, a través de una revisión manual sea posible identificar algún fraude cometido por personas que marcan a cuenta de otras.

Los reportes que el programa puede generar son:

- Varias nóminas.
- Historial.
- Reporte de atrasos y faltas.
- Tiempo no trabajado.
- Control de marcas en el reloj.
- Selección manual.
- Asistencia de personal por día.
- Planilla mensual.
- Cuadro mensual exportado a Excel.

#### **SmartClk Control de Asistencia (16)**

Es un programa diseñado para asistir en la gestión de control horario y de asistencia de personal que permite evitar errores de cálculo producidos en el control de asistencia por métodos no informatizados de

forma fácil, eficaz y eficiente. Además de la incorporación de registros horarios de cualquier reloj electrónico en sus bases de datos ayudando al operador en la depuración de las marcas de asistencia y el control de los horarios mediante la incorporación de herramientas efectivas y el ingreso de registros de justificación de falta por distintos conceptos. Los mismos pueden crearse en el sistema especificando el tipo de concepto para la justificación de falta. Ejemplos de estos son licencias, faltas con aviso, etc.

#### **Características y Funciones:**

- **Seguridad:** incorpora elementos de seguridad que permiten el registro de los operadores y todas las operaciones realizadas en el programa. Esta información se almacena en un registro de auditoría el cual puede analizarse posteriormente a los efectos de controlar la buena gestión de los datos. El ingreso a la aplicación es mediante usuario y contraseña.
- **Horarios:** permite el control de horarios fijos y rotativos fijos, los cuales se debe especificar la hora de entrada, de salida y los días o el período que se debe controlar; horarios libres, los cuales no se especifica más que una carga horaria y los días que debe cumplir esa carga horaria; y horarios presencia, donde el sistema controla que esas personas por lo menos cuente con una marca indicando su presencia. En todos los casos el sistema soporta horarios que pasen la media noche.
- **Informes:** posibilita emitir informes de inconsistencias, inasistencias, flash de asistencia, diferencias de horario (entradas tardes / salidas anticipadas), cálculo de horas trabajadas y planilla resumen. Los informes permiten gran flexibilidad para la selección, filtro de registros (filas y columnas), impresión y vista previa de impresión.
- **Carga de Datos Externos:** proporciona la carga de los datos iniciales del sistema a partir de archivos de datos en formato de texto externos. Soporta la carga de funcionarios, Secciones y Horarios. Adicionalmente puede leer los registros de asistencia de archivos de texto generados por cualquier reloj de control.
- **Interfaz con otros sistemas:** incluye una salida de datos en formato de texto con cuatro tablas: registros horarios, Inconsistencias, Inasistencias y Horas. Esta interfaz se utiliza para entregar la información de horas trabajadas a un sistema de liquidación de haberes.
- **Asignación de Marcas:** permite desdoblar un código de marca creado en un reloj de control, en dos códigos correspondientes a Entrada y Salida. Esta asignación se realiza teniendo en cuenta el tipo de horario asignado al funcionario y la secuencia de marcas que el mismo ha generado.
- **Soporte para equipos biométricos:** a partir de la versión 3.11 se incluye la lectura de marcas y la

administración de las huellas para verificación biométrica para terminales Identix Finger Scan V20 directamente en la aplicación. Para otros equipos se puede integrar con las aplicaciones Smart Lan para terminales F4/A4 y relojes de tarjetas magnéticas y de proximidad RTA600.

- Requerimientos Mínimos del Sistema:
  - Windows 98 SE / NT SP6 / 2000 / XP.
  - Procesador Pentium 500Mhz.
  - 64 Mb de memoria RAM.

#### **Exactus Pro 2010 (17) (18)**

Es un efectivo sistema de control de asistencia basado en un ordenador que registra la marcación del personal a través de un código de barra, una tarjeta magnética u otro medio de lectura. El programa brinda una capacidad manejo ilimitado tanto de turnos y/o horarios por empleado, como de empleados por estación, adaptación inmediata a sistemas multiusuario y emisión de múltiples reportes de asistencia/ausencias en línea, sin necesidad de procesos previos. El costo de la licencia adicional de Exactus PRO 2010 es de \$55 US y del sistema de \$250 USD.

Características y funciones:

- Ofrece diversos informes: tiempo trabajado, ausencias, tardanzas, sobre tiempo y todos aquellos reportes que se deseen adicionar. El sistema registra las marcaciones a través de la lectura de un código de barra (o lector magnético) impreso en tarjetas de marcación personalizadas llamadas Exactus Pro Cards.
- Es compatible con relojes biométricos U300 (US\$580), TFT 500(US\$695), F7 (US\$595), F4, entre otros.
- Puede mantener un turno definido para cada día del mes por cada empleado. Mantiene los datos de todos los empleados que trabajan en su empresa y la historia de todas las marcaciones registradas por si se requiriese alguna comprobación posterior.
- Capacidad de migrar datos de tiempo a cualquier programa de planilla gracias a la flexibilidad de su interfaz. Permite revisar e importar las marcaciones desde un Exactus Pro 2010 a otra terminal dentro de una red o remotamente desde otra sucursal a través de módem.
- Controles de acceso que brindan la seguridad necesaria para salvaguardar la información. Ningún empleado podrá marcar una vez vencido su contrato de trabajo e igualmente no podrá hacerlo antes de su fecha de ingreso.



- Permite administrar el tiempo que su personal toma como *Coffee Break* así como los constantes permisos para asuntos personales.
- Permite ver en tiempo real todos los informes del sistema.
- Posee generador de reportes, calculadora, calendario, verificador de archivos. Mensajería Interna.
- Sistemas operativos soportados: Windows 2000, Windows XP y Window 7.

#### **Aplicación Web del Control Horario - Universidad de Zaragoza (19)**

Esta aplicación de control del horario se emplea en la actualidad en la Universidad de Zaragoza, España para lograr un adecuado cumplimiento de la normativa en materia de calendario laboral, jornada y horario de los trabajadores de este centro.

Se utiliza el término fichaje como sinónimo de marcaje, ambos términos definen una incidencia de una persona en una fecha y hora determinada, realizada desde un lector o desde la propia aplicación. El personal de administración y servicios, en general, puede realizar fichajes mediante un ordenador personal, desde su puesto de trabajo, ver sus fichajes y visualizar su cuadro mensual de bloques de trabajo una vez revisado por el responsable de su unidad.

Los responsables de las unidades se encargarán de la supervisión del cumplimiento de la jornada y horario del personal a su cargo, así como del control de las diferentes causas de inasistencia al trabajo. Existen dos tipos fundamentales de incidencias:

1. Incidencias dentro de la jornada laboral: el fichaje de las entradas y salidas y demás incidencias es responsabilidad de cada persona y pueden ser realizadas desde el lector o desde la web (los errores u olvidos deben ser corregidos por los responsables de unidad).
2. Ausencias planificadas: estas incidencias pueden corresponder, en algunos casos, a días completos y es el responsable de unidad quien debe introducir este tipo de incidencias en la aplicación.

Para el desarrollo de esta aplicación se emplearon tecnologías basadas en la web, con una interface y navegación homogénea con el resto de las aplicaciones que viene desarrollando el Servicio de Informática, se deberá contar con un navegador web (Microsoft Internet Explorer o Netscape Navigator) para poder utilizarla. Los desarrolladores recomiendan la instalación de la última versión disponible del navegador seleccionado.

Entre las principales funcionalidades que brinda la aplicación se encuentran las tres siguientes:

1. **Fichar ahora:** esta opción permite fichar desde cada ordenador seleccionando el código correspondiente a la incidencia en un menú desplegable. La aplicación informática identifica la máquina desde la que se realiza la operación de marcaje.
2. **Ver mis fichajes:** desde esta opción se consultan los fichajes realizados por cada usuario entre dos fechas determinadas. La modificación de los fichajes es función de los responsables de la unidad. Se podrán ver los fichajes realizados entre las fechas seleccionadas indicando: el día de la semana, la fecha, la hora, la duración de la jornada de ese día si es una ausencia planificada, el tipo de marcaje mediante un icono (lector o web), la procedencia (una etiqueta que identifica el lector o la dirección IP (*Internet Protocol*) del ordenador desde el que se efectuó el fichaje), quién realizó ese fichaje mediante un icono (el propio usuario o el administrador) y el código y la incidencia asociada.
3. **Cuadros mensuales:** desde esta opción se consultan los horarios de entradas, salidas e incidencias en un mes determinado para cada usuario. No estará disponible si existe algún error (por esa razón no podrá verse dentro de la jornada de trabajo, hasta que se cierre el bloque de salida aunque todo este correcto). Se indica: día de la semana, fecha, hora de entrada y salida, el número de horas de presencia real (Hp), el número de horas de incidencias (Hi), el número de horas de ausencia por motivos particulares (Haap), el número de horas teóricas justificadas ( $Ht=Hp+Hi$ ), los códigos de incidencia usados en el bloque (Ci), las horas trabajadas el sábado (Hs), las horas extraordinarias (He), las horas en exceso de jornada irregular (Hx) y las horas de recuperación de ausencias particulares (Hrap).

#### **Sistema de Control de Asistencia de la Fundación para el Desarrollo de la Ciencia y la Tecnología del estado Mérida, Venezuela (FUNDACITE) (20)**

Sistema utilizado para el control de la asistencia de todo el personal de la FUNDACITE. Cuenta con la autenticación de usuarios manejado por sesiones para consultar los datos, utiliza como entrada de datos un dispositivo llamado dedómetro Modelo, además presenta las siguientes características:

- Ambiente: web environment.
- Estado de Desarrollo: 5 - *Production/Stable*.
- Idioma Utilizado: English, Spanish.
- Intendencia de audiencia: End Users/Desktop, System Administrators.
- Lenguaje de Programación: Javascript, PHP.

- Licencia: GNU *General Public License* (GPL).
- Sistema Operativo: OS *Independent*, Linux.
- Tópicos: Software Development.

Se necesita tener instalado como mínimo:

- Un servidor web: el sistema se ha probado y funciona correctamente en Apache 2.0.52.
- Un servidor MySQL: versión 3.23.58 o mayor.
- PHP Versión 4.3.11.

Hay que tomar en cuenta que este sistema está instalado en Fedora (GNU&Linux). Sin embargo en la mayoría de los casos estos archivos pueden tener configuraciones iguales tanto para GNU&Linux como para Windows. Es posible que el sistema funcione para otras versiones, sin embargo las que se mencionaron son las versiones en las que está funcionando en la actualidad.

El sistema permite:

- Consultar la asistencia del personal en un periodo o mes determinados
- Autenticación de usuarios para el manejo del sistema.
- Mostrar todas las notas incluidas a la asistencia, incluir o consultar una nota.

### **Análisis de los sistemas empleados en el mundo**

Los software SICON (1), E-time (3), SmartClk (16) y Exactus Pro 2006 (17) brindan muchas funcionalidades relacionadas con el control de la asistencia, pero requieren leer los datos provenientes de terminales colectoras de datos, usando tarjetas con código de barras, proximidad, banda magnética, dispositivos biométricos u otros similares. Debido al elevado costo que representa en la actualidad instalar este tipo de dispositivos, el Departamento no tendría la posibilidad de adquirir esta tecnología, esto unido al hecho de que todos presentan licencia privativa, imposibilitaría dar solución al problema de la presente investigación empleando uno de estos sistemas.

Por otra parte Doce versión 2 (15) no necesita contar con lectores de datos para la captura de los mismos, pues en este sistema el control de asistencia se efectúa usando el teclado de una computadora en la cual el personal registra un código individual de la misma manera en que se podría hacer con un reloj tarjetero. La autora de esta tesis es del criterio que de implantarse este sistema en el Departamento de GRCW pondría de manifiesto la limitación de que cada estudiante tendría que marcar su entrada sólo desde una computadora. Esto haría imposible poder determinar cuántas horas el estudiante trabaja en el laboratorio

en un periodo determinado en su computadora por lo que tampoco se podría calcular el uso real de la tecnología. Este software al igual que la Aplicación Web del Control Horario de la Universidad de Zaragoza (19), posee ciertas funcionalidades que se adaptan a las necesidades de la presente investigación; pero se imposibilita la adecuación a los requerimientos de la situación problemática porque ambos poseen licencia privativa.

En relación con el Sistema de Control de Asistencia de la (FUNDACITE) (20) hay que tomar en cuenta que aunque este sistema posee licencia GNU/GPL permitiendo su modificación y aunque los autores plantean que los archivos pudieran tener configuraciones iguales tanto para GNU&Linux como para Windows, se intentó configurarlos tanto para Windows XP como para Ubuntu 9.0 (GNU&Linux) Sistema Operativo empleado en el Departamento, sin tener éxito.

#### **1.1.2. Sistemas automatizados utilizados en Cuba**

##### **Sistema de Recursos Humanos (Fastos) (4)**

El sistema de Recursos Humanos (Fastos), está formado por los módulos Configuración, Personal, Capacitación y Cuadros, permite controlar las informaciones fundamentales de los empleados de una entidad, también realizar varios procesos y operaciones que son inherentes al área de recursos humanos, tales como:

- Registro de los empleados: se guardan los datos de los empleados, así como informaciones referentes a los reportes de vacaciones, certificados médicos, licencias, resolución.
- Control de la plantilla: permite establecer la estructura organizativa de las plazas de la entidad.
- Control de asistencia: incorpora el control de claves de asistencias, turnos de trabajos, horarios, tarjetas de asistencia e incidencias de cada empleado. Además, posibilita acoplar relojes (RTA 600) para actualizar la información de la tarjeta de asistencia de forma automática.
- Informes y modelos: proporcionan la obtención de un total de 56 informes, por ejemplo cierre del periodo, análisis de fondo de tiempo, estadísticos, entre otros, y 11 modelos entre los que se encuentran: el acta de entrega del expediente laboral, contratos de trabajo, guía de documentos archivados, convenio de trabajadores, etc.
- Control de la capacitación respecto a: acciones de capacitación, estudios realizados, cursos, eventos, experiencia docente, publicaciones, conocimientos, idiomas extranjeros, plan de desarrollo, informes y otros aspectos.
- Control de la información de los cuadros: se establece el registro de los cuadros, dirigentes y

reserva, referente a evaluaciones, inspecciones, sanciones, necesidades de capacitación, entre otros.

**El sistema tiene las siguientes características generales:**

- Aplicación cliente servidor, con base de datos de SQL 2000 Server.
- Ayuda incorporada.
- Protección contra copias ilegales.
- Control y registro de acceso al sistema.
- Facilidades para la exportación de información.

**Requerimientos para la explotación del producto:**

- Las PC (*Personal Computer*) deben tener instaladas el sistema operativo Windows 98 o superior, 256MB RAM como mínimo.
- Sistema de redes estable y confiable pues este permitirá tener acceso a la información con mayor rapidez y seguridad.
- Gestor de Base de Datos SQL 2000.

**“GREHU: Un sistema integral para gestionar los Recursos Humanos” (5)**

Este software constituye el resultado de un profundo estudio realizado en la temática de Gestión de los Recursos Humanos, obteniéndose una herramienta de software que permite registrar, procesar y gestionar de forma integrada las principales funciones que se desarrollan en la dirección de Recursos Humanos de una entidad laboral entre ellas la del control de la asistencia de los trabajadores.

El software desarrollado permite realizar de forma automatizada e integrada las principales funciones que se realizan en la Dirección de Recursos tales como: el inventario de personal, el control de las sanciones y amonestaciones, la selección y contratación, la evaluación del desempeño, el trabajo con los dirigentes y cuadros y el procesamiento de las nóminas.

El sistema informatizado, conjuntamente con los procedimientos metodológicos propuestos se está explotando desde hace algunos años en las Direcciones de Recursos Humanos y de Personal de varias instalaciones de Cubanacan S.A y de la Cadena Hotelera de Gran Caribe, obteniéndose resultados satisfactorios, según opiniones de los funcionarios de estas instalaciones. Con su utilización, se controla, almacena y obtiene una gran cantidad de información sobre los recursos humanos en particular, pueden realizar múltiples inferencias, para la toma de decisiones oportunas con datos actualizados. (5)

Este sistema, con sus innumerables salidas tanto estadísticas como gráficas, hace posible que en mayor o menor medida, los directivos relacionados con la gestión de personal, puedan conocer y prever las posibles promociones, necesidades de formación y capacitación, los reclutamientos futuros, el comportamiento de la disciplina laboral, el desempeño del personal, así como el desarrollo y un control efectivo de la asistencia a la empresa. El sistema fue programado en FoxPro para WINDOWS versión 2.6.

#### **Análisis de los sistemas utilizados en Cuba**

El Sistema de Recursos Humanos Fastos (4), permite controlar las informaciones fundamentales de los empleados de una entidad, también realizar varios procesos y operaciones que son inherentes al área de recursos humanos. Para el control de asistencia incorpora el manejo de claves de asistencias, turnos de trabajos, horarios, tarjeta de asistencia e incidencias de cada empleado; pero necesita acoplar relojes (RTA 600) para actualizar la información de las tarjetas de asistencia de forma automática, lo que presupone inversiones adicionales en tecnología.

GREHU (5): un sistema integral para gestionar los Recursos Humanos brinda la posibilidad de realizar de forma automatizada e integrada las principales funciones que se llevan a cabo en la Dirección de Recursos tales como: el inventario de personal, el control de las sanciones y amonestaciones, la selección y contratación, la evaluación del desempeño, el trabajo con los dirigentes y cuadros y el procesamiento de las nóminas. La autora de este trabajo de diploma considera que la principal deficiencia de este sistema estriba en que no captura de forma automática las inasistencias e impuntualidades, por lo que en los centros donde está implantado este sistema se debe realizar por medio de un libro de firmas. Además, ambos sistemas presentan la limitación de que sólo pueden operar sobre sistema operativo Windows.

#### **1.1.3. Sistemas automatizados empleados en la UCI.**

##### **Sistema de Control de Acceso SAM (21)**

Este sistema se emplea para controlar el acceso a los laboratorios del módulo del MININT en la UCI. Entre las funcionalidades que brinda este sistema se encuentran:

- Registros históricos de entrada y salida de usuarios por fecha.
- Listado de personal fuera y dentro de la entidad.
- Bloqueo y desbloqueo de usuarios.
- Registrar y eliminar visitantes.

Características:

- Simple: desarrollada en un solo módulo con dos niveles de autenticación.

- Administrador: registro, visualización, actualización y eliminación de personas. Visualización de reportes y administración del sistema.
- Usuario: limitado a visualizar una interfaz para registrar la entrada o salida del personal.
- Captura de Datos: mediante un lector de códigos de barras o escribiendo el número de solapín en el sistema.
- Modelo - Vista - Controlador MVC.Net.
- Visualización - EXTJS 3.0.
- Acceso y control de datos - ASP.Net C#.
- Gestor BD –ORACLE.
- Servidor: IIS (*Internet Information Server*) en Windows XP.

#### **Sistema de Gestión Integral ASSETS NS (22) (23)**

Es un sistema integral modular concebido para el control de la actividad económica empresarial. Este software es empleado en la UCI para calcular las nóminas y controlar los recursos laborales de que dispone esta entidad. Entre las funcionalidades que brinda ASSETS NS, se encuentra la de emitir reportes de impuntualidades e inasistencias de cada trabajador de un día en específico. Un detalle a destacar es que para que el sistema pueda emitir estos reportes, los datos deben de ser introducidos manualmente a partir de los modelos de asistencia que son recogidos.

Desde el punto de vista de hardware, requiere ser instalado en un servidor de bases de datos, normalmente ubicado en una red de computadoras, el cual emplea los recursos del servidor en atender simultáneamente, las solicitudes de un elevado número de clientes, brindándoles la información solicitada en un tiempo mínimo y garantizando la integridad máxima de sus datos.

#### **Requerimientos del sistema:**

##### **Servidor**

- Equipo: Intel® o compatible Pentium III 800 MHz o superior.
- Memoria: 128 MB RAM como mínimo (se recomienda 256 o más).
- Disco duro: 1 GB como mínimo.
- Microsoft Windows NT Server 4.0 (Service pack 5) o superior.
- Microsoft SQL Server 2000.

#### **Cliente**

- Equipo: Intel® o compatible Pentium 166 MHz o superior.
- Memoria: 64 MB RAM como mínimo (se recomienda 128 o más).
- Disco duro: 60 MB como mínimo.
- Microsoft Windows'98 o superior.

#### **Módulo de control de acceso de UCI-Lab (24)**

Este módulo se emplea para controlar el acceso a los laboratorios de producción del docente 1 (viejo) en la UCI, pero puede ser configurado para que controle el acceso tanto para un laboratorio en específico, como para todo un bloque de laboratorios. El sistema también muestra los reportes de la cantidad y el por ciento de estudiantes que han accedido a un laboratorio en un día y hora determinados.

Actualmente se está trabajando en una nueva versión de este sistema que incluirá entre los elementos adicionales del nuevo proyecto: la reservación y el sistema del estado del equipamiento. También próximamente se implantará este sistema en los laboratorios de *Enterprise Resource Planning* (ERP).

#### **Análisis de los sistemas empleados en la UCI**

El Sistema de Control de Acceso SAM y el módulo de control de acceso de UCI-Lab poseen las ventajas de que previenen el acceso de personal ajeno al área, permiten valorar el aprovechamiento de los recursos y saber en todo momento quien se encuentra dentro del local. No brindan la posibilidad de emitir reportes de impuntualidades e inasistencias, simplemente se limitan a generar reportes de quienes han estado en el laboratorio en una fecha u horario determinado. Además, para su implementación es necesario tener en la entrada del local una persona que verifique en todo momento que el acceso de los usuarios al sistema ha sido correcto y que a su vez los usuarios marquen la salida de la aplicación. En los locales donde se ha implantado este sistema, se ha presentado la dificultad de que en múltiples ocasiones los usuarios al salir de la entidad no marcan la salida de la aplicación, lo que provoca que los registros que se obtengan no sean los reales.

Por otra parte el Sistema de Gestión Integral ASSETS NS no posee una captura de los datos automática, por lo que los mismos deben ser introducidos manualmente. Esto origina una gran masa de datos en papel, gastos innecesarios de recursos, duplicidad y manejo ineficiente de la información, entre otros inconvenientes. También se debe señalar que los sistemas antes mencionados sólo funcionan sobre sistema operativo Windows.



En la mayoría de los Departamentos de la UCI la asistencia es tomada diariamente por los líderes o profesores que atienden a los estudiantes en sus líneas de trabajo, para luego ser registrada en una hoja de cálculo de Excel.

## **1.2. Tendencia y tecnología actuales para el desarrollo de aplicaciones web.**

Teniendo en cuenta las necesidades anteriormente planteadas y las características del entorno donde se aplicará la solución propuesta, se realizó un estudio de las tendencias y tecnologías actuales posibles a emplear, puntualizadas a continuación.

### **1.2.1. Lenguajes del lado del cliente**

#### **HTML**

*Hyper Text Markup Language* o Lenguaje de Marcas de Hipertexto es el lenguaje de marcado predominante para construir páginas web y usado normalmente en la *World Wide Web*. Define la estructura y el contenido de las páginas permitiendo combinar textos, imágenes, sonidos, videos y enlaces a otras páginas. Su nivel de complejidad es bajo y además permite embeber dentro de su código otros script escritos en lenguajes como PHP y Javascript. (25)

#### **CSS**

*Cascade Style Sheet* u Hoja de Estilos es un lenguaje formal que define cómo se va a mostrar un documento escrito en HTML o XML. Establece la separación definitiva de la lógica (estructura) y el físico (presentación) del documento. El *World Wide Web Consortium (W3C)* es el encargado de formular la especificación de las hojas de estilo que servirán de estándar para los navegadores. (26) (27)

#### **Javascript**

Es un lenguaje orientado a objetos, interpretado, que se ejecuta del lado del cliente. Utilizado principalmente en páginas Web y con sintaxis semejante a Java o C, sigue la programación basada en prototipos. Además usa DOM (Document Object Model) para acceder y modificar el contenido, estructura y estilo de los documentos HTML y XML. Dentro de las principales tecnologías para interactuar con DOM que usa Javascript se encuentran AJAX y DHTML. (28)

#### **Ajax**

Asynchronous Javascript and XML (Javascript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas o RIA (*Rich Internet Applications*). Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el

servidor en segundo plano. De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargarlas, lo que significa aumentar la interactividad, velocidad y usabilidad en las aplicaciones.

Es una tecnología asíncrona, en el sentido de que los datos adicionales se requieren al servidor y se cargan en segundo plano sin interferir con la visualización ni el comportamiento de la página. Javascript es el lenguaje interpretado (*scripting language*) en el que normalmente se efectúan las funciones de llamada de Ajax mientras que el acceso a los datos se realiza mediante XML Http Request, objeto disponible en los navegadores actuales. En cualquier caso, no es necesario que el contenido asíncrono esté formateado en XML.

#### 1.2.2. Lenguajes del lado del servidor

##### PHP

PHP (*Hypertext Preprocessor*) es un lenguaje de “código abierto” interpretado, de alto nivel, embebido en páginas HTML y ejecutado en el servidor. Contiene infinidad de bibliotecas con funciones orientadas al manejo de bases de datos, imágenes, películas flash, directorio, entre muchas otras. Soporta además la programación orientada a objetos y es una alternativa potente para la creación de portales web muy dinámicos. PHP puede ser utilizado en cualquiera de los principales sistemas operativos del mercado, incluyendo GNU&Linux, muchas variantes Unix (HP-UX, Solaris y Open BSD), Microsoft Windows, Mac OS X y RISC OS.

Soporta la mayoría de servidores Web de hoy en día, incluyendo Apache, *Microsoft Internet Information Server* (Microsoft IIS), Personal Web Server, Netscape e iPlanet, Oreilly Website Pro server, Caudium, Xitami, OmniHTTPd y muchos otros. Quizás la característica más destacable de PHP es su soporte para una gran cantidad de bases de datos. Destacan entre ellas MySQL, PostgreSQL, Oracle (OCI7 y OCI8), Sybase, ODBC, FrontBase, Unix dbm, Adabas D, entre otras.

#### 1.2.3. Gestores de Bases de Datos

Un Sistema Gestor de Base de Datos (SGBD) es una herramienta que permite crear y mantener una base de datos, asegurando su integridad, confidencialidad y seguridad. Permiten definir una base de datos (especificar tipos, estructuras y restricciones de los datos), construirla (guardar los datos en algún medio controlado por el propio SGBD) y manipular su contenido (realizar consultas, actualizarla y generar informes). Partiendo de que la propuesta es un módulo para el CMS Drupal y que por ende soporta los Gestores de Base de Datos tanto PostgreSQL 8.4 como MySQL 5.0, a continuación se abordan ambos.

##### PostgreSQL 8.4

Gestor de base de datos distribuido bajo la licencia BSD, relacional y multiplataforma. Sostenida por una activa comunidad de desarrolladores denominada PostgreSQL Global *Development Group*. Presenta una amplia gama de características como *triggers*, vistas, integridad transaccional, variedad de tipos de datos (texto de largo ilimitado, *arrays*, direcciones IP y MAC, figuras geométricas) y con posibilidad de que los usuarios puedan crear los suyos propios. Lenguajes como C, C++, Java, Perl, Python, Ruby, PHP, entre otros contienen librerías que permiten usar este gestor. (29)

#### **MySQL 5.0**

Es el SGBD más usado en Internet, su licencia es dual debido a que se ofrece bajo la licencia GNU/GPL para cualquier uso compatible con la misma, pero las empresas que deseen desarrollar software privativo deben pagar una licencia específica que les permita su uso. Por otra parte es importante señalar que consume pocos recursos y es utilizado en aplicaciones de diferentes complejidades y además en plataformas como Windows/Linux-Apache-MySQL-PHP/Perl/Python.

Varios lenguajes contienen APIs (*Application Programming Interface*) con las cuales pueden acceder a las bases de datos en MySQL, por ejemplo C++, C#, Java, Lisp, Perl, Python, Ruby y PHP. MySQL es una base de datos muy rápida en la lectura cuando utiliza el motor no transaccional MyISAM, pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. En aplicaciones web hay baja concurrencia en la modificación de datos, en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones (30). La validación de la propuesta se realizó en este Gestor de Base de Datos.

#### **1.2.4. Los Sistemas de Gestión de Contenidos**

Un Sistema de Gestión de Contenidos (*Content Management System* o CMS) son aplicaciones altamente configurables, que cubren el ciclo de vida completo de las páginas de un sitio. Se utilizan principalmente para publicar, editar, eliminar, actualizar y administrar la información de un portal web sin conocimiento alguno de HTML, ya sea en Internet o en una intranet, proporciona la capacidad para gestionar la estructura del sitio y la navegación a los usuarios. Basado en un entorno cien por ciento web, la finalidad de un CMS es la creación de aplicaciones y sitios web dinámicos de forma rápida y eficiente. (31)

El mayor beneficio de los CMS es que proporcionan apoyo a las metas de negocio y estrategias de los clientes. Por ejemplo, los CMS pueden ayudar a mejorar las ventas, aumentar la satisfacción de los usuarios, o ayudar en la comunicación con el público. Las funcionalidades de un sistema de gestión de contenido se pueden desglosar en varias categorías principales: creación, administración, publicación y

presentación del contenido (32). Esta investigación se basa en el CMS Drupal, por lo que a continuación se describe el mismo.

#### **Drupal**

Existen muchos CMS de código abierto de gran calidad y con muy buena aceptación en el mercado, uno de ellos es Drupal. El mismo despunta por su robustez, su flexibilidad, su código immaculado, su apariencia y capacidad infinita de extensión y personalización.

“Drupal”, es el equivalente fonético en inglés a la palabra neerlandesa ‘*druppel*’ que significa ‘gota’, de ahí el logo que lo identifica.

Drupal es un sistema de gestión de contenido modular y muy configurable. Es un programa de código abierto, con licencia GNU/GPL, escrito en PHP, desarrollado y mantenido por una activa comunidad de usuarios. Se destaca por la calidad de su código y de las páginas generadas, el respeto de los estándares de la web y un énfasis especial en la usabilidad y consistencia de todo el sistema.

El diseño es especialmente idóneo para construir y gestionar comunidades en Internet. No obstante, su flexibilidad y adaptabilidad, así como la gran cantidad de módulos adicionales disponibles, hace que sea adecuado para realizar diferentes tipos de sitio web.

Dicho CMS presenta potencialidades que hacen de este uno de los más utilizados a nivel mundial:

- Comunidad de desarrollo: Drupal cuenta con una amplia comunidad de usuarios, los cuales se mantienen implementando nuevas versiones y módulos para lograr el crecimiento del CMS.
- Carácter modular: basado en la inclusión de módulos elaborados por la comunidad de desarrolladores que proporcionan las más disímiles funcionalidades.
- Personalización: un robusto entorno de personalización está implementado en el núcleo de Drupal. Tanto el contenido como la presentación pueden ser individualizados de acuerdo a las preferencias definidas por el usuario.
- Control de versiones: el sistema de control de versiones de Drupal permite seguir y auditar totalmente las sucesivas actualizaciones del contenido.
- Plantillas (*Templates*): el sistema de temas de Drupal separa el contenido de la presentación permitiendo controlar o cambiar fácilmente el aspecto del sitio web. Se pueden crear plantillas con HTML y/o con PHP.
- Sindicación del contenido: Drupal exporta el contenido en formato RDF/RSS para ser utilizado por otros sitios web.

- Independencia del gestor de base de datos: aunque la mayor parte de las instalaciones de Drupal utilizan MySQL, existen otras opciones. Drupal incorpora una “capa de abstracción de base de datos” que actualmente está implementada y mantenida para MySQL y PostgreSQL, aunque permite incorporar fácilmente soporte para otras bases de datos.
- Multiplataforma: Drupal ha sido diseñado desde el principio para ser multiplataforma. Puede funcionar con Apache o Microsoft IIS como servidor web y en sistemas como GNU&Linux, BSD, Solaris, Windows y Mac OS X. Por otro lado, al estar implementado en PHP, es totalmente portable.
- Administración vía web: la administración y configuración del sistema se puede realizar enteramente con un navegador y no precisa de ningún software adicional. (33) (34) (35)

### 1.3. Metodologías de desarrollo de software

Se define como metodología al conjunto de estrategias, procedimientos, métodos o actividades intencionadas, organizadas, secuenciadas e integradas, que permitan el logro de aprendizajes significativos y de calidad. (36)

A su vez, una metodología para el desarrollo del proceso de software es el conjunto de actividades necesarias para transformar los requisitos de los usuarios en un sistema software. En el vertiginoso mundo de la informática, por lo riesgoso y difícil de controlar que es un proceso de software, el uso de metodologías que apoyen el mismo, es un tema muy discutido. Una correcta elección evitaría entre otros problemas, pérdida de tiempo, capital, insatisfacción en clientes y desarrolladores o peor aún, el colapso mismo del producto. En realidad no existe una metodología estándar, sino que las características de cada proyecto, las de un equipo de desarrollo, recursos disponibles y tiempo para su elaboración exigen la flexibilidad del proceso, adaptándose el mismo al entorno y teniendo como objetivo alcanzar la máxima calidad en lo que se produce. Hoy en día, dentro de las más conocidas encontramos: *Rational Unified Process* (RUP), *eXtreme Programming* (XP) y *Feature Driven Development* (FDD); en su traducción al español, Proceso Unificado de Desarrollo, Programación Extrema y Desarrollo Guiado por Funcionalidad, respectivamente. (37) (38)

#### **XP** (37) (39)

Se encuentra dentro del grupo de metodologías ágiles. Intenta reducir la complejidad del software por medio del trabajo orientado directamente al objetivo, basado en las relaciones interpersonales y la velocidad de reacción.

XP intenta minimizar el riesgo de fallo del proceso por medio de la disposición permanente de un representante competente del cliente a disposición del equipo de desarrollo. El mismo debe estar apto para contestar de forma rápida y concreta cualquier duda que pueda surgir por parte del equipo, de manera que la toma de decisiones no se retrase.

Defiende la idea del trabajo en parejas (dos programadores, un ordenador), por lo que se espera que la calidad del mismo aumente en el momento en que se escribe. El código pertenece al equipo completo y no solo a unos pocos, de manera que cualquier programador pueda cambiar cualquier parte del código en cualquier momento si se necesita. Esto se logra debido a que cada componente del equipo debe trabajar al menos una vez con cada uno de los demás integrantes y con cada componente de software, de manera que el conocimiento de la aplicación lo posea todo el equipo.

#### **FDD (37)**

Esta metodología está pensada para proyectos con tiempo de desarrollo relativamente cortos (menos de un año). Basada en un proceso con iteraciones que producen un software funcional que el cliente y la dirección de la empresa pueden ver y monitorizar.

FDD se divide en cinco fases:

- Desarrollo de un modelo general.
- Construcción de la lista de funcionalidades.
- Plan de *releases* en base a las funcionalidades a implementar.
- Diseñar en base a las funcionalidades.
- Implementar en base a las funcionalidades.

El trabajo (tanto de modelado como de desarrollo) se realiza en grupo, aunque siempre habrá un responsable último (arquitecto jefe o jefe de programadores en función de la fase en la que se encuentre), con mayor experiencia, que tendrá la última palabra en caso de no llegar a un acuerdo. Al hacerlo en grupo se consigue que todos formen parte del proyecto y que los inexpertos aprendan de las discusiones de los más experimentados, y al tener un responsable último, se asignan las responsabilidades que todas las empresas exigen.

#### **RUP (37) (40)**

Es un proceso de desarrollo de software que junto con el Lenguaje Unificado de Modelado, se emplean para implementar y documentar sistemas orientados a objetos. El mismo está dirigido

por casos de uso, centrado en la arquitectura e iterativo e incremental. RUP propone el desarrollo de productos en cuatro fases: Inicio, Elaboración, Construcción y Transición.

En cada fase se ejecutarán una o varias iteraciones en dependencia del tamaño del proyecto. Además propone nueve flujos de trabajo, los seis primeros ingenieriles y el resto de soporte. Los mismos se llevan a cabo en cada una de las fases anteriormente expuestas. Ellos son: Modelado del negocio, Requisitos, Análisis y diseño, Implementación, Pruebas, Despliegue, Gestión de cambio y configuraciones, Gestión de proyecto, Entorno.

RUP es uno de los procesos más generales de los existentes actualmente, ya que en realidad está pensado para adaptarse a cualquier proyecto. El proceso define una serie de roles que se distribuyen entre los miembros del proyecto y que definen las tareas de cada uno y el resultado (artefactos según la terminología de RUP) que de ellos se espera.

#### **Análisis de las metodologías de desarrollo de software.**

Teniendo en cuenta las características del proyecto y las metodologías antes mencionadas, se decidió que en el desarrollo del sistema no se utilizará la metodología XP debido a que el alcance del proyecto no se define completamente al comienzo del mismo y a que la metodología XP es expresamente abierta a los cambios durante todo el proceso hace que se torne sumamente difícil estimar un presupuesto previo. Además define poca documentación, lo que constituye una desventaja al pensar en el posterior mantenimiento del sistema, porque si bien durante el desarrollo de proyecto el equipo tiene en mente todas sus particularidades, hay que prever que pasará luego de entregado, cuando el equipo se disuelva y sea necesario realizar algún cambio o mejora. XP propone la recodificación permanente, para asegurar la claridad y simplicidad del código. Sin embargo, es posible que aún un código simple y claro no baste cuando otro equipo de trabajo tenga que tomar el sistema y cambiarlo, lo que hace necesario mantener cierta documentación. (41)

Por otra parte el principal problema de FDD está representado por la necesidad de contar con miembros experimentados en el equipo, miembros que puedan definir los roles y funciones de cada uno, que marquen el camino a seguir desde el principio con la elaboración del modelo global.

Por tanto la metodología más adecuada para dirigir el proceso de desarrollo del software es RUP, ya que define claramente actividades realizadas por roles generando a su paso artefactos que sustentan el proceso de desarrollo del producto. RUP constituye una metodología adaptable al proyecto, utilizada para el análisis, implementación y documentación de sistemas a través del UML (*Unified Modeling Language*), que implementa el Paradigma Orientado a Objetos.

Además por ser una de sus características principales el desarrollo incremental posee las ventajas siguientes:

- Los clientes no necesitan esperar hasta el fin del desarrollo para utilizar el sistema, pueden empezar a usarlo desde el primer incremento.
- Se disminuye el riesgo de fracaso de todo el proyecto, ya que se puede distribuir en cada incremento.
- Las partes más importantes del sistema son entregadas primero, por lo cual se realizan más pruebas en estos módulos y se disminuye el riesgo de fallos.

Es importante señalar que la metodología recomienda, que en dependencia de las características del proyecto y de la organización, se seleccionen los artefactos, actividades y roles que van a ser utilizados, hace mayor énfasis en la planificación, control del proyecto y en la especificación precisa de los requisitos.

#### **1.4. Herramientas para el desarrollo del software.**

Las herramientas de desarrollo son un cúmulo de software que permiten al equipo de desarrollo una serie de ventajas, comodidades y recursos para poder dar respuesta a un problema dado. A continuación se detallan brevemente las que se utilizaron y permitieron planear, desarrollar, probar y documentar la aplicación web.

##### **Planner (42)**

Es un proyecto de código abierto que permite la planeación de las tareas. Dentro de sus características principales cuentan:

- Definición de tareas y subtareas con fecha de inicio y fin y su tiempo de duración.
- Definir la dependencia de tareas.
- Definir recursos y asignarlos a las distintas actividades.
- Guardar los horarios en formato XML.
- Producir Diagramas de Gantt en formato HTML.
- Controlar el estado de cumplimiento de las actividades.

##### **Quanta Plus 3.2.1 (43)**

Quanta Plus es un entorno de desarrollo web rico y estable. Es un software destinado a la maquetación HTML/CSS, que no es más que convertir los esbozos creados por diseñadores en una plantilla HTML, con su respectiva hoja de estilos e imágenes usadas.

Destacan rasgos como:



- Resaltado de sintaxis de HTML, Javascript, CSS y varios más.
- Contiene un analizador que informa acerca de la correcta creación de las páginas.
- Los documentos pueden ser previsualizados dentro de la aplicación usando el motor KHTML. Es posible preprocesar los documentos a través de un servidor web antes de visualizar.
- A pesar de ser una aplicación para el entorno KDE, funciona también en el entorno GNOME.

#### **NetBeans IDE 6.8 (44)**

Primer IDE que brinda soporte completo para Java EE 6 y *Sun Glass Fish Enterprise Server v3* y ofrece PHP mejorado, soporte para JavaFX y C/C ++. Además ofrece soporte completo para Java™ Platform Enterprise Edition 6 (Java EE 6) y *Sun Glass Fish™ Enterprise Server v3*, así como otras funcionalidades innovadoras.

Junto con el soporte para Java EE 6 y *Glass Fish v3*, el NetBeans IDE 6.8 ofrece otras nuevas características y mejoras que incluyen:

- Soporte PHP Ampliado: expande el soporte de los lenguajes dinámicos con apoyo para PHP 5.3 y el esquema de Symfony acelera el desarrollo de aplicaciones web PHP.
- Una integración más ajustada con Project Kenai: Project Kenai, un entorno de colaboración para acoger proyectos Open Source, ofrece ahora soporte completo para JIRA así como mensajería instantánea mejorada y una integración de seguimiento de issues.
- Mejora de C / C + + *Profiling*: perfila y sintoniza aplicaciones C / C + + con el nuevo indicador *Microstate Accounting*, supervisor de uso I/O.
- JavaFX™: código de finalización mejorado, sugerencias y navegación para JavaFX en el editor NetBeans.

#### **Firebug 1.53 (45)**

Firebug es una extensión de Firefox creada y diseñada especialmente para desarrolladores y programadores web. Es un paquete de utilidades con el que se puede analizar (revisar velocidad de carga, estructura DOM), editar, monitorizar y depurar el código fuente, CSS, HTML y Javascript de una página web de manera instantánea y online.

Firebug no es un simple inspector como DOM Inspector, además edita y permite guardar los cambios. Su atractiva e intuitiva interfaz, con pestañas específicas para el análisis de cada tipo de elemento (consola, HTML, CSS, Script, DOM y red), permite al usuario un manejo fácil y rápido. Firebug está encapsulado en forma de plug-in o complemento de Mozilla, es *Open Source* (aunque no Free Software) y de distribución gratuita.

#### **Servidor Web Apache 2.0 (46) (47)**

Apache es un servidor web flexible, rápido y eficiente, continuamente actualizado y adaptado a los nuevos protocolos (HTTP 1.1). Entre sus características destacan:

- Multiplataforma.
- Es un servidor de web conforme al protocolo HTTP/1.1.
- Modular: puede ser adaptado a diferentes entornos y necesidades, con los diferentes módulos de apoyo que proporciona, y con las API de programación de módulos, para el desarrollo de módulos específicos.
- Basado en hebras.
- Incentiva la realimentación de los usuarios, obteniendo nuevas ideas, informes de fallos y parches para la solución de los mismos.
- Se desarrolla de forma abierta.
- Extensible: gracias a ser modular se han desarrollado diversas extensiones entre las que destaca PHP, un lenguaje de programación del lado del servidor.

#### **Visual Paradigm 6.4 (48)**

Visual Paradigm para UML es una herramienta profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El software de modelado UML agiliza la construcción de aplicaciones de calidad y a un menor costo. Permite dibujar todos los tipos de diagramas de clases, realizar ingeniería inversa, generar documentación y código desde diagramas, importación desde Rational Rose, integración con MS Visio, integración IDE con Visual Studio, IntelliJ IDEA, Eclipse, NetBeans y otros. La herramienta UML CASE también proporciona abundantes tutoriales, demostraciones interactivas y proyectos UML.

## **CAPÍTULO II: CARACTERÍSTICAS DEL SISTEMA PARA EL CONTROL DE ASISTENCIA INTEGRADO AL CMS DRUPAL.**

Este capítulo expone las características del sistema a implementar. Se determinan requisitos funcionales y no funcionales, actores, casos de uso del sistema y se realiza el modelo de dominio, de modo que facilite de manera visual mostrar al lector los principales conceptos que se manejan en el dominio del sistema en desarrollo.

### **2.1. Problema y situación problemática**

En el Departamento de Gestión y Representación de Contenidos Web al cual pertenece el presente trabajo de diploma, los controles de horario productivo se han basado fundamentalmente en la toma de la asistencia de manera manual, apoyándose en una hoja de cálculo de Excel o por simple observación. Recientemente con los cambios organizativos de la actividad de desarrollo de software en la Universidad, el horario se divide en cuatro sesiones: mañana (8:00 am-12:00pm), tarde (1:00pm-5:00pm), noche (7:00pm-11pm) y madrugada (11:00pm-2:00pm). El control de asistencia consiste en que al arribar el estudiante al laboratorio, su profesor tutor le registra la hora de entrada en una hoja de cálculo de Excel y el mismo proceso se repite para la salida.

Al no existir un marcado de la asistencia de forma automática, esto demanda que los datos tengan que ser introducidos manualmente por el profesor tutor, quien debe marcar la hora de entrada y la salida de los estudiantes bajo su responsabilidad. Además, se dificulta llevar un seguimiento de los registros de asistencia para cada estudiante, procesar los datos obtenidos y producir información, reportes y otras salidas para los directivos del Departamento.

El hecho de no contar con un sistema que gestione el control de asistencia vinculado a la producción de los estudiantes del Departamento, incide en la toma de decisiones erróneas con los estudiantes a la hora de emitir una nota correspondiente a la asignatura Práctica Profesional; esto a su vez puede repercutir en los niveles de motivación de los estudiantes vinculados a la producción. Además, dado un proceso de determinación del uso real de la tecnología, ya sea por horas de consumo eléctrico o aprovechamiento del tiempo de estudio individual, que constituyen elementos subjetivos importantes, no se reúnen los datos exactos debido a que no se cuenta con la estadística o el reporte de cuántas horas el estudiante trabaja en el laboratorio en un periodo determinado.

## 2.2. Propuesta del Sistema

La solución propuesta en la presente investigación consiste en el diseño e implementación de un módulo para el Sistema de Gestión de Contenidos (CMS) Drupal, que permite controlar los elementos de asistencia, incidencias laborales y es capaz de emitir evaluaciones parciales de acuerdo a los registros de asistencia almacenados. Para cumplir con los objetivos trazados, el sistema brinda la posibilidad de definir las sesiones de producción del Departamento y de establecer un horario de producción semanal (teniendo en cuenta las sesiones creadas) para cada estudiante.

Entre las ventajas que posee el sistema que se propone se encuentran:

- Posee código abierto, con licencia GNU/GPL.
- Es multiplataforma, por lo que puede ser utilizado desde cualquier sistema operativo.
- Necesita requerimientos mínimos de hardware para su funcionamiento.
- Es totalmente portable, en este sentido puede ser usado en otro sistema basado en Drupal, en su versión 6.
- Puede ser empleado en los gestores de base de datos PostgreSQL y MySQL.
- Cada usuario puede marcar la asistencia de forma automática desde su computadora de trabajo (sin necesidad de dispositivos colectores de datos), lo que brinda la posibilidad de determinar cuál es el uso real de la tecnología.
- Presenta controles de acceso que brindan la seguridad necesaria para salvaguardar la información y garantizar la fiabilidad de los datos obtenidos. Entre los que se destacan:
  - Ningún estudiante puede marcar la asistencia una vez terminado su horario de trabajo e igualmente no podrá hacerlo antes de su hora de ingreso.
  - Un estudiante sólo puede marcar la asistencia desde una sola computadora en una misma sesión de trabajo.

Cada entrada o salida de un usuario genera un registro que contiene:

- Fecha y hora en que se realizó el registro.
- Nombre de usuario.
- Dirección IP desde donde se hizo el registro.
- Información adicional que es introducida por los usuarios administrativos que se dividen justificaciones de ausencias e indisciplinas cometidas durante el horario de producción.

A partir de los datos obtenidos, el sistema permite emitir reporte de un estudiante o de todos los

estudiantes, dado un periodo de tiempo determinado. Entre los reportes se pueden encontrar los siguientes:

- Cantidad de horas trabajadas.
- Cantidad de ausencias, impuntualidades y salidas anticipadas.
- Cantidad de incidencias.
- Evaluaciones.
- Estudiantes que se encuentran trabajando en el Departamento.

Además el sistema brinda la posibilidad de imprimir los reportes y exportarlos como pdf. Otra funcionalidad que proporciona la propuesta es la notificación a los estudiantes que se ausentaron al horario de producción. Esto se realiza de forma diaria al concluir la última jornada de producción mediante el envío de un correo que contiene las sesiones a las que se ausentó el estudiante.

### 2.3. Modelo de dominio

Después de haber realizado un estudio de los procesos que se van a efectuar, se llegó a la conclusión de que el negocio estudiado tiene muy bajo nivel de estructuración, donde los flujos de información se encuentran difusos y cuando se desea realizar una actividad múltiples personas intervienen en la misma, lo que implica un solapamiento de responsabilidades, además es difícil establecer las reglas de funcionamiento. Por lo anteriormente expresado, se entendió que era factible elaborar un modelo de dominio, de modo que se facilitara de manera visual mostrar al usuario los principales conceptos que se manejan en el dominio del sistema en desarrollo. Esto ayuda a los usuarios, clientes, desarrolladores e interesados a utilizar un vocabulario común para poder entender el contexto en que se enmarca el sistema.

Es necesario tener un vasto conocimiento de cómo debe funcionar el proceso en cuestión, para poder capturar correctamente los requisitos y así poder construir un sistema con las características que el Departamento necesite. Este modelo va a contribuir posteriormente a identificar algunas clases que se utilizarán en el sistema.

A continuación aparecen varios conceptos que serán de utilidad para entender la lógica del modelado.

**Departamento:** estructura productiva dentro de la Facultad que organiza a profesores y estudiantes y que posee un Jefe de Departamento como directivo.

**Jefe de Departamento:** máxima autoridad en el Departamento.

**Profesores:** profesores de la Facultad vinculados a la producción en el Departamento y que son tutores

de un grupo de estudiantes. Entre sus responsabilidades se encuentran la de tomar la asistencia de los estudiantes bajo su tutela.

**Estudiantes:** estudiantes de 3ro, 4to y 5to año de la Facultad, vinculados a la producción en el Departamento.

**Registros:** información obtenida dentro de la jornada de producción, entre ellos se encuentran los datos obtenidos a partir de los fichajes de entradas y salidas como: fecha, hora, nombre de usuario y dirección IP desde donde se efectuaron. Los registros a su vez están compuestos por las Incidencias.

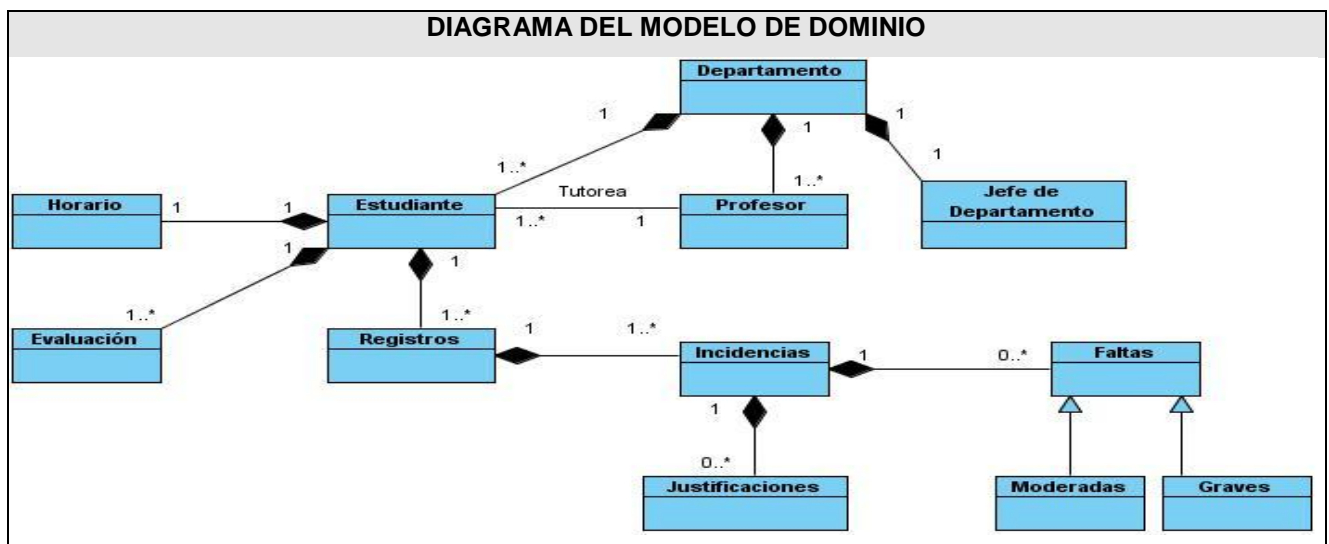
**Incidencias:** acontecimientos de carácter negativo que tienen lugar durante la producción y que son detectados por el profesor tutor, o ausencias que son reportadas con antelación por los estudiantes. Estas se agrupan en Faltas y Justificaciones.

- Faltas: violaciones cometidas por los estudiantes que son clasificadas en graves o moderadas, de acuerdo a la relevancia que tengan.
- Justificaciones: motivos que excusan las ausencias de los estudiantes al horario de producción.

**Evaluación:** nota del estudiante en base a los registros de asistencia obtenidos.

**Horario:** horario de producción de los estudiantes del Departamento. Se encuentra dividido entre los días de la semana y las sesiones de producción.

### Diagrama del Modelo de Dominio



### 2.4. Levantamiento de requisitos

Es el segundo flujo de trabajo propuesto por la metodología RUP. En la que se define cuáles son las

condiciones o capacidades que deben ser alcanzadas o poseídas por el sistema que se desea construir. El usuario final debe tener total conocimiento y comprensión de los requisitos y aceptar los mismos. Se dividen en dos grupos: los requisitos funcionales y los requisitos no funcionales.

#### **2.4.1. Requerimientos funcionales**

Los requisitos funcionales son capacidades o condiciones que el sistema debe cumplir. Constituyen una base para poder identificar lo que realmente debe hacer el sistema. Los mismos deben ser comprendidos tanto por los desarrolladores como por los clientes. A continuación se muestran los que la aplicación a desarrollar debe cumplir.

##### **RF1 Sesiones de producción.**

RF1.1. Mostrar sesiones de producción.

RF1.2. Editar horario de las sesiones de producción.

##### **RF2 Horario de estudiante.**

RF2.1. Crear horario de estudiante.

RF2.2. Mostrar horario de estudiante.

RF2.3. Editar horario de estudiante.

RF2.4. Eliminar horario de estudiante.

##### **RF3 Buscar registros de asistencia de estudiante.**

##### **RF4 Buscar registros personales.**

##### **RF5 Registrar asistencia.**

RF5.1. Registrar entrada.

RF5.2. Registrar salida.

##### **RF6 Reportar incidencias de estudiante.**

RF6.1. Reportar faltas de estudiante.

RF6.2. Reportar justificaciones de estudiante.

##### **RF7 Mostrar estudiantes que se encuentran produciendo.**

##### **RF8 Registrar jornada de producción.**

#### **2.4.2. Requerimientos no funcionales del sistema**

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener como restricciones del entorno o de la implementación, rendimiento, dependencias de la plataforma, facilidad de

mantenimiento, entre otras. (49)

Los requisitos no funcionales identificados en el sistema propuesto se detallan a continuación.

#### **Requisitos de Usabilidad**

- El sistema podrá ser usado por cualquier persona que posea conocimientos básicos en el manejo de la computadora y de un ambiente web en sentido general.

#### **Requisitos de Fiabilidad**

- La información contenida en el sistema deberá ser totalmente confiable.
- Deberán implementarse mecanismos para garantizar la respuesta ante posibles fallos lo más rápido posible.
- Ante cualquier falla en el sistema se deberán mostrar los errores sin dar detalles de información que puedan comprometer la seguridad e integridad del mismo.

#### **Requisitos de Rendimiento**

El sistema demora en una transición aproximadamente 3 segundos como máximo. Por detrás de la aplicación web se realizarán consultas a bases de datos y validaciones de formularios que contengan información, por lo que se requiere de cierto software y hardware para cumplir con los tiempos de transición de la interfaz de usuario.

#### **Requisitos de Software**

- Intérprete de aplicaciones web (Navegador).
- Servidor apache (versión 1.5).
- Sistema Gestor de Bases de Datos (MYSQL 5.0 o PostgreSQL 8.4).
- Versión de PHP 5.0.

#### **Requisitos de Hardware**

##### **Cliente**

- 512 Mb de memoria RAM.
- 40 Gb de Disco duro.
- Procesador Pentium VI a 3,00 GHz.

##### **Servidor**

- 1 Gb de memoria RAM.
- 80 Gb de Disco Duro.
- Procesador Intel Dual Core a 1,86 GHz.

#### **Requisitos de Soporte**

- Servidor apache versión 2.0 o superior.



- Sistema Gestor de Bases de Datos MYSQL 5.0 (o superior) o PostgreSQL 8.4 (o superior).
- Versión de PHP 5.0 o superior.
- El sistema debe dar la posibilidad de ser mejorado, así como de incorporarle nuevos servicios en caso de ser necesarios.

#### Restricciones de diseño

- Versión de PHP 5.0 o superior.
- Por parte del cliente se requiere un navegador capaz de interpretar Javascript.

#### Requisitos de Seguridad

- Se deberán establecer los permisos para el acceso a las funcionalidades de administración del sistema y garantizar que las mismas se visualicen de acuerdo al nivel de usuario que está activo.
- El producto deberá tener protección contra acciones no autorizadas o que puedan afectar la integridad de los datos.

#### Requisitos de Portabilidad

- El sistema deberá ser multiplataforma y permitir que se acceda a él desde los navegadores más comunes.

#### Requisitos de Interfaz

- El producto deberá ser legible.
  - El Diseño deberá estar orientado a una navegación sencilla.

#### Legales

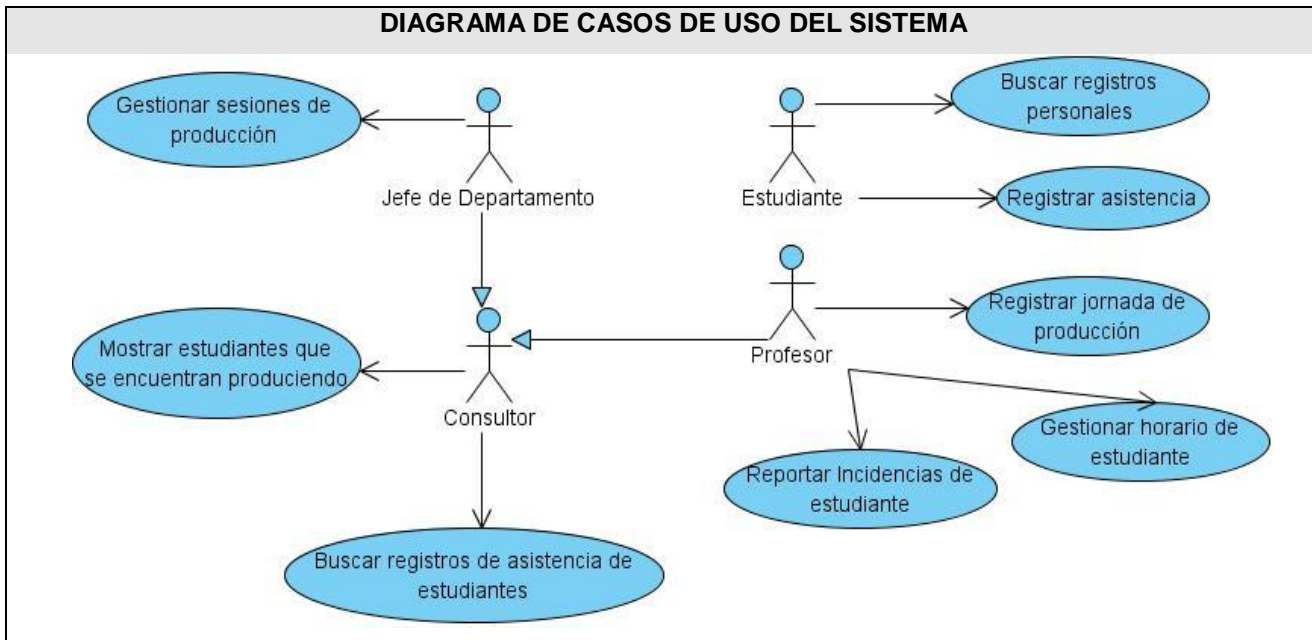
- La plataforma escogida para el desarrollo del producto deberá estar basada en la licencia GNU/GPL.

#### 2.4.3. Definición de los actores del sistema

Actor	Descripción
Jefe de Departamento	Define las sesiones de producción.
Profesor	Es el encargado de la supervisión del cumplimiento del horario de producción de los estudiantes bajo su responsabilidad, reportar incidencias y registrar las sesiones de producción.
Consultor	Accede a buscar los registros de asistencia de los estudiantes y además, puede ver los estudiantes que se encuentran trabajando en el proyecto.

Estudiante	Es el usuario que accede al sistema para marcar su asistencia o consultar sus registros de asistencia.
------------	--

**2.4.4. Diagrama de Casos de Uso del Sistema**



**2.4.5. Descripción de los Casos de Uso del Sistema**

Caso de Uso	Gestionar sesiones de producción
Actores:	Jefe de Departamento
Resumen:	El Jefe de Departamento una vez que haya entrado en el sistema accede a la interfaz de “Sesiones de Producción” e introduce los datos necesarios.
Referencias	RF1.1., RF1.2.
Prioridad	Crítico
Precondiciones:	Estar logueado
Flujo Normal de Eventos	
Sección “General”	
Acción del Actor	Respuesta del Sistema

<p>1. El Jefe de Departamento accede al sistema para gestionar las sesiones de producción.</p>	<p>1.1. El sistema muestra las distintas opciones:</p> <p>a) Si desea ver una sesión de producción ir a la sección “Mostrar sesiones de producción”.</p> <p>b) Si desea modificar una sesión ir a la sección “Editar sesiones de producción”.</p>
<p>Flujo Normal de Eventos</p>	
<p>Sección “Mostrar sesiones de producción”</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
<p>1. El Jefe de Departamento selecciona el menú “Sesiones de Producción”.</p>	<p>1.1. El sistema despliega una tabla donde se muestran las sesiones de Producción que tiene el Departamento con su hora de entrada y salida en cajas de texto editables.</p>
<p>Flujo Normal de Eventos</p>	
<p>Sección “Editar sesiones de producción”</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
<p>2. El Jefe de Departamento selecciona el menú “Sesiones de Producción”.</p> <p>3. El Jefe de Departamento modifica el horario de las sesiones que desee y presiona el botón “Aceptar”.</p>	<p>2.1. El sistema despliega una tabla donde se muestran las sesiones de Producción que tiene el Departamento con su hora de entrada y salida en cajas de texto editables.</p> <p>3.1. Queda modificado el horario de la sesión y muestra el mensaje “Las sesiones de producción han sido modificadas exitosamente”.</p>
<p>Flujo alterno</p>	

Acción del Actor		Respuesta del Sistema
3. a)	Si al intentar modificar la hora de entrada o salida de las secciones, las horas no se encuentran con el formato adecuado, al presionar el botón "Aceptar".	3.1. a) El sistema muestra el mensaje "El formato de la hora es incorrecto". Volver a la acción 2.1.
3. b)	Si al intentar modificar la hora de entrada o salida de las secciones, las horas no coinciden con el orden de las sesiones.	3.1. b) El sistema muestra el mensaje "El orden de las horas de entrada y salida es incorrecto". Volver a la acción 2.1.
Poscondiciones	Quedan gestionadas las sesiones de producción.	

Caso de Uso	Gestionar horario de estudiante	
Actores:	Profesor	
Resumen:	El Profesor una vez que haya entrado en el sistema accede a la interfaz "Horarios de Usuarios" e introduce los datos necesarios.	
Referencias	RF2.1., RF2.2., RF2.3., RF2.4.	
Prioridad	Crítico	
Precondiciones:	Estar logueado	
Flujo Normal de Eventos		
Sección "General"		
Acción del Actor		Respuesta del Sistema
1.	El Jefe de Departamento accede al sistema para gestionar horarios de estudiantes.	1.1. El sistema muestra las distintas opciones: a) Si desea adicionar un horario ir a la sección "Crear horario de estudiante". b) Si desea ver un horario ir a la sección "Mostrar horario de estudiante". c) Si desea modificar un horario ir a la

	<p>sección “Editar horario de estudiante”.</p> <p>d) Si desea suprimir un horario ir a la sección “Eliminar horario de estudiante”.</p>
Flujo Normal de Eventos	
Sección “Crear horario de estudiante”	
Acción del Actor	Respuesta del Sistema
<p>1. El Profesor selecciona el menú “Horarios de Usuarios”.</p> <p>2. El profesor introduce el nombre del usuario y establece el horario del estudiante dando clic en la intersección entre los días y sesiones en la tabla. Finalmente selecciona la opción “Aceptar”.</p>	<p>1.1. El sistema muestra un campo de texto para introducir el usuario del estudiante al que se le establecerá el horario y debajo una tabla con los días de la semana/sesiones de trabajo del Departamento.</p> <p>2.1. Queda establecido el horario de producción del estudiante y se muestra el mensaje “El horario de producción para usuario ha sido creado”.</p>
Flujo Normal de Eventos	
Sección “Mostrar horario de estudiante”	
Acción del Actor	Respuesta del Sistema
<p>3. El Profesor selecciona el menú “Horarios de Usuarios”.</p> <p>4. El Profesor introduce el nombre de usuario del estudiante y selecciona la</p>	<p>3.1. El sistema muestra un campo de texto para introducir el usuario del estudiante y debajo una tabla con los días de la semana/sesiones de trabajo del Departamento.</p> <p>4.1. a) El sistema actualiza la tabla, destacando los checkbox que</p>

opción "Aceptar".	corresponden con el horario de producción del estudiante en cuestión. 4.1. b) Si el usuario no tiene un horario asignado se muestra el mensaje: "El usuario no tiene horario asignado".
<b>Flujo Normal de Eventos</b>	
<b>Sección "Editar horario de estudiante"</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
5. El Profesor selecciona el menú "Horarios de usuarios".	5.1. El sistema muestra un campo de texto para introducir el usuario del estudiante y debajo una tabla con los días de la semana/sesiones de trabajo del Departamento.
6. El Profesor introduce el nombre de usuario del estudiante y selecciona la opción "Aceptar".	6.1. El sistema actualiza la tabla, destacando los checkbox que corresponden con el horario de producción del estudiante en cuestión.
7. El profesor modifica el horario de producción del estudiante haciendo clic en los checkbox que se encuentran. Finalmente selecciona la opción "Aceptar".	7.1. Queda modificado el horario de producción del estudiante y se muestra el mensaje "El horario de producción para usuario ha sido creado".
<b>Flujo Normal de Eventos</b>	
<b>Sección "Eliminar horario de estudiante"</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
8. El Profesor selecciona el menú "Horarios de usuarios".	8.1. El sistema muestra un campo de texto para introducir el usuario del estudiante y debajo una tabla con los días de la semana/sesiones de trabajo del

<p>9. El Profesor introduce el nombre de usuario del estudiante y selecciona la opción "Aceptar".</p> <p>10. El Profesor desmarca todos los checkbox que estaban seleccionados en la tabla y finalmente selecciona la opción "Aceptar".</p>	<p>Departamento.</p> <p>9.1. El sistema actualiza la tabla, destacando los checkbox que corresponden con el horario de producción del estudiante en cuestión.</p> <p>10.1. Queda eliminado el horario de producción del estudiante y se muestra el mensaje "El horario de producción para usuario ha sido eliminado".</p>
<b>Poscondiciones</b>	Queda gestionado el horario del estudiante.

<b>Caso de Uso</b>	Registrar asistencia	
<b>Actores:</b>	Estudiante	
<b>Resumen:</b>	El Estudiante una vez que haya entrado en el sistema accede a la interfaz de "Control de Asistencia".	
<b>Referencias</b>	RF5.1., RF5.2.	
<b>Prioridad</b>	Crítico	
<b>Precondiciones:</b>	Estar logueado	
<b>Flujo Normal de Eventos</b>		
<b>Sección "Registrar Entrada"</b>		
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>	
<p>1. El Estudiante accede al sistema en un horario en que le corresponde trabajar.</p> <p>2. El Estudiante presiona el botón "Registrar Entrada" a la hora establecida.</p>	<p>1.1. El sistema muestra un bloque titulado "Control de Asistencia" con la opción "Registrar Entrada".</p> <p>2.1. El botón desaparece y aparece en su lugar el botón de "Registrar Salida" y se muestra el mensaje "Ha registrado su entrada correctamente".</p>	
<b>Flujo Alternativo</b>		

Acción del Actor		Respuesta del Sistema
1. Si cuando el Estudiante accede al sistema, no le corresponde trabajar según su horario.		1.1. El sistema no muestra el bloque titulado "Control de Asistencia".
Flujo Normal de Eventos		
Sección "Registrar Salida"		
Acción del Actor		Respuesta del Sistema
3. El Estudiante accede al sistema en un horario en que le corresponde trabajar y ya previamente ha registrado la salida.		3.1. El sistema muestra un bloque titulado "Control de Asistencia" con la opción "Registrar Salida".
4. El Estudiante presiona el botón "Registrar Salida" a la hora establecida.		4.1. El botón desaparece y se muestra el mensaje "Ha registrado su salida correctamente".
Flujo Alternativo		
Acción del Actor		Respuesta del Sistema
3. Si cuando el Estudiante accede al sistema, no ha marcado previamente la entrada y le corresponde trabajar en ese día y sesión.		3.1. El sistema muestra el bloque titulado "Control de Asistencia" con la opción "Registrar Entrada".
4. Si el Estudiante presiona el botón "Registrar Salida", en un horario menor que la hora de salida de la sesión.		4.1. El botón desaparece y se muestra el mensaje "Ha registrado su salida anticipadamente".
Poscondiciones	Queda registrada la asistencia.	

Caso de Uso	Registrar jornada de producción
Actores:	Profesor
Resumen:	El Profesor una vez que haya entrado en el sistema accede al botón "Registrar Jornada de Producción".
Referencias	RF8.



Prioridad	Crítico	
Precondiciones:	Estar logueado	
Flujo Normal de Eventos		
Sección “Registrar jornada de producción”		
	Acción del Actor	Respuesta del Sistema
	<p>1. El Profesor accede al sistema en un horario en que ya ha terminado la última sesión.</p> <p>2. El Profesor presiona el botón “Registrar Jornada”.</p>	<p>1.1. El sistema muestra un bloque titulado “Registrar Jornada” con el botón “Registrar Jornada”.</p> <p>2.1. El botón desaparece y se muestra el mensaje “Se han registrado los datos de la jornada correctamente”.</p> <p>3.1. El sistema envía una notificación a los estudiantes que se ausentaron injustificadamente ese día, mediante el envío de un correo que contiene las sesiones a las que este faltó.</p>
Flujo Alternativo		
	Acción del Actor	Respuesta del Sistema
	1. Si cuando el Profesor accede al sistema, ya se ha registrado previamente esa jornada de producción.	1.1. El sistema no muestra el bloque titulado “Registrar Jornada”.
Poscondiciones	Quedan registrados los datos referentes a la asistencia para ese día.	

Caso de Uso	Buscar registros de asistencia de estudiante
Actores:	Consultor
Resumen:	El Consultor una vez que hayan entrado en el sistema accede al Bloque de “Reportes” e introduce los datos necesarios.
Referencias	RF3.
Prioridad	Secundario

Precondiciones:	Estar logueado	
Flujo Normal de Eventos		
Sección “Buscar registros de asistencia de estudiante”		
Acción del Actor	Respuesta del Sistema	
1. El Consultor selecciona el reporte que desea ver del bloque “Reportes”.	1.1. El sistema despliega un formulario para que el Consultor introduzca los criterios de búsqueda y muestra encima del formulario las opciones: “Versión para impresora” y “Versión para PDF”.	
2. El Consultor introduce los criterios de búsqueda y selecciona la opción “Ver Reportes”.	2.1. El sistema muestra una tabla debajo del formulario con la información solicitada.	
Flujo alternativo		
Acción del Actor	Respuesta del Sistema	
2. a) Si al introducir los criterios de búsqueda, se deja algún campo requerido en blanco y se presiona el botón “Ver Reportes”.	2.1. a) El sistema muestra el mensaje “El campo “nombre del campo” es requerido”. Volver a la acción 1.1.	
2. b) Si cuando el Consultor introduce los criterios de búsqueda y presiona el botón “Ver Reportes”, no se encuentran registros que correspondan con los solicitados.	2.1. b) El sistema muestra el mensaje “No hay resultados que mostrar para los parámetros seleccionados”.	
Poscondiciones	Quedan mostrados los registros solicitados.	

Caso de Uso	Buscar registros personales
Actores:	Estudiante
Resumen:	El Consultor una vez que hayan entrado en el sistema accede al Bloque de “Reportes” e introduce los datos necesarios.

Referencias	RF4.	
Prioridad	Secundario	
Precondiciones:	Estar logueado	
Flujo Normal de Eventos		
Sección "Buscar registros personales"		
Acción del Actor	Respuesta del Sistema	
1. El Estudiante selecciona el reporte que desea ver del bloque "Reportes".	1.1. El sistema despliega un formulario para que el Estudiante introduzca los criterios de búsqueda.	
2. El Estudiante introduce los criterios de búsqueda y presiona el botón "Ver Reportes".	2.1. El sistema muestra una tabla debajo del formulario con la información solicitada.	
Flujo alternativo		
Acción del Actor	Respuesta del Sistema	
2. a) Si al introducir los criterios de búsqueda, se deja algún campo requerido en blanco y se presiona el botón "Buscar".	2.1. a) El sistema muestra el mensaje "El campo "nombre del campo" es requerido". Volver a la acción 1.1.	
2. b) Si cuando el Estudiante introduce los criterios de búsqueda y presiona el botón "Ver Reportes", no se encuentran registros que correspondan con los solicitados.	2.1. b) El sistema muestra el mensaje "No hay resultados que mostrar para los parámetros seleccionados".	
Poscondiciones	Quedan mostrados los registros personales solicitados.	

Caso de Uso	Mostrar estudiantes que se encuentran produciendo.
Actores:	Consultor
Resumen:	El Consultor una vez que hayan entrado en el sistema accede a la interfaz de "Trabajando Actualmente".

Referencias	RF7.	
Prioridad	Secundario	
Precondiciones:	Estar logueado	
Flujo Normal de Eventos		
Sección “Mostrar estudiantes que se encuentran produciendo”		
Acción del Actor	Respuesta del Sistema	
1. El Consultor selecciona el menú “Trabajando Actualmente”.	1.1. El sistema muestra una tabla con el nombre de los estudiantes y la dirección IP de la computadora en la que cada uno se encuentra en ese momento.	
Flujo alterno		
Acción del Actor	Respuesta del Sistema	
3. Si al acceder a la interfaz de “Trabajando Actualmente” no se encuentra ningún estudiante trabajando en ese momento.	2.1. El sistema muestra el mensaje “No hay estudiantes trabajando en estos momentos”.	
Poscondiciones	Quedan mostrados los estudiantes que se encuentran trabajando.	

Caso de Uso	Reportar incidencias de estudiante.	
Actores:	Profesor	
Resumen:	El Profesor una vez que haya entrado en el sistema accede a la interfaz de “Reportar Incidencias” e introduce los datos necesarios.	
Referencias	RF6.1., RF6.2.	
Prioridad	Secundario	
Precondiciones:	Estar logueado	
Flujo Normal de Eventos		
Sección “Reportar Faltas de estudiante”		
Acción del Actor	Respuesta del Sistema	
1. El Profesor selecciona el menú “Reportar	1.1. El sistema despliega un formulario para	

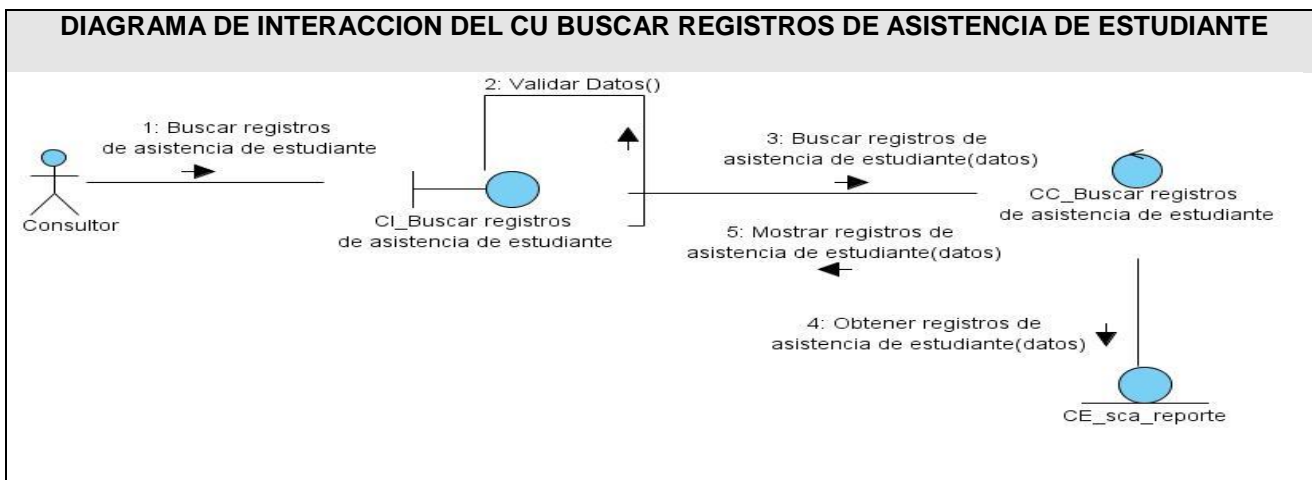
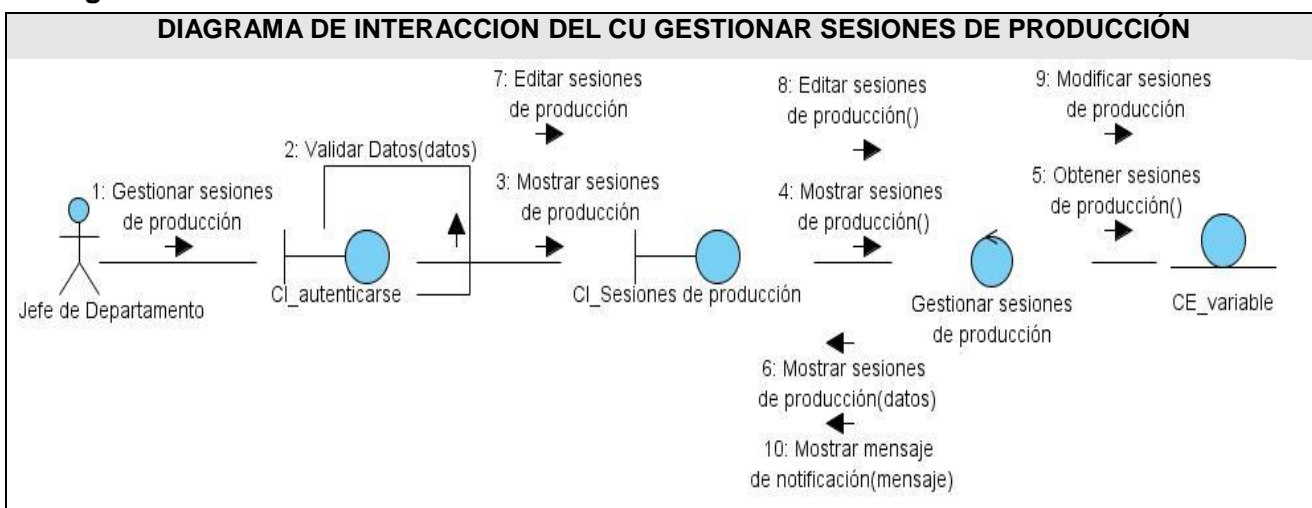
incidencias”.	que el Profesor introduzca los datos requeridos de la incidencia.
2. El Profesor introduce los datos de la incidencia y en tipo de incidencia selecciona: “Faltas”. Luego al finalizar, presiona el botón “Aceptar”.	2.1. El sistema comprueba la validez de los datos. 2.2. La aplicación crea la incidencia de tipo Falta.
<b>Flujo alternativo</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
2. Si al introducir los datos solicitados, la fecha de la falta es diferente de la fecha actual.	2.1. El sistema muestra el mensaje “Las faltas solo se registran en el día actual”. Volver a la acción 1.1.
<b>Flujo Normal de Eventos</b>	
<b>Sección “Reportar Justificaciones de estudiante”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
3. El Profesor selecciona el menú “Reportar incidencias”.	3.1. El sistema despliega un formulario para que el Profesor introduzca los datos requeridos de la incidencia.
4. El Profesor introduce los datos de la incidencia y en tipo de incidencia selecciona: “Justificaciones”. Luego al finalizar, presiona el botón “Aceptar”.	4.1. El sistema comprueba la validez de los datos. 4.2. La aplicación crea la incidencia de tipo Justificación.
<b>Flujo alternativo</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
4. Si al introducir los datos solicitados, la fecha de la Justificación es menor que la fecha actual.	4.1. El sistema muestra el mensaje “Las justificaciones no se registran en fechas anteriores a la actual”. Volver a la acción 3.1.
<b>Poscondiciones</b>	Quedan introducidas las incidencias.

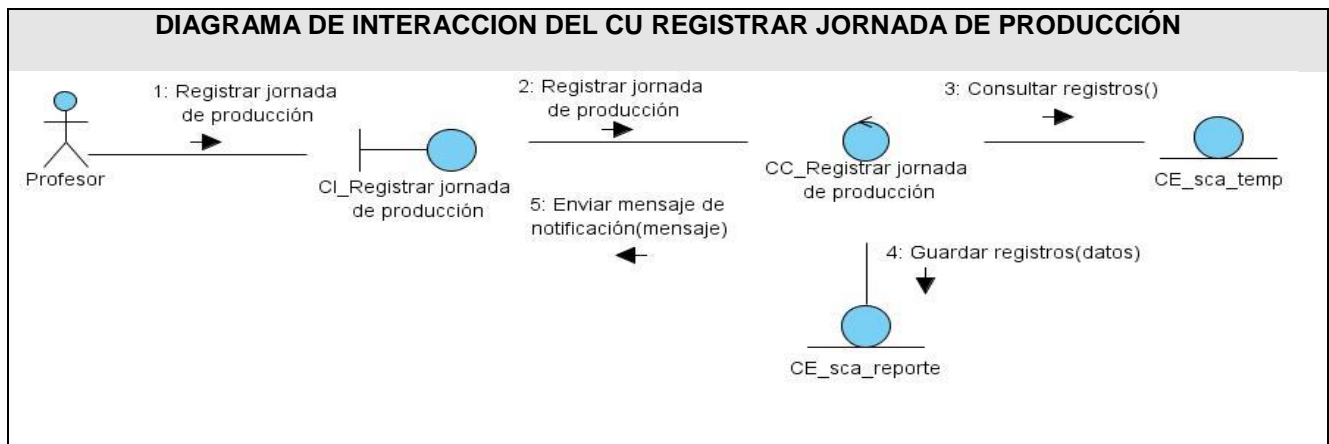
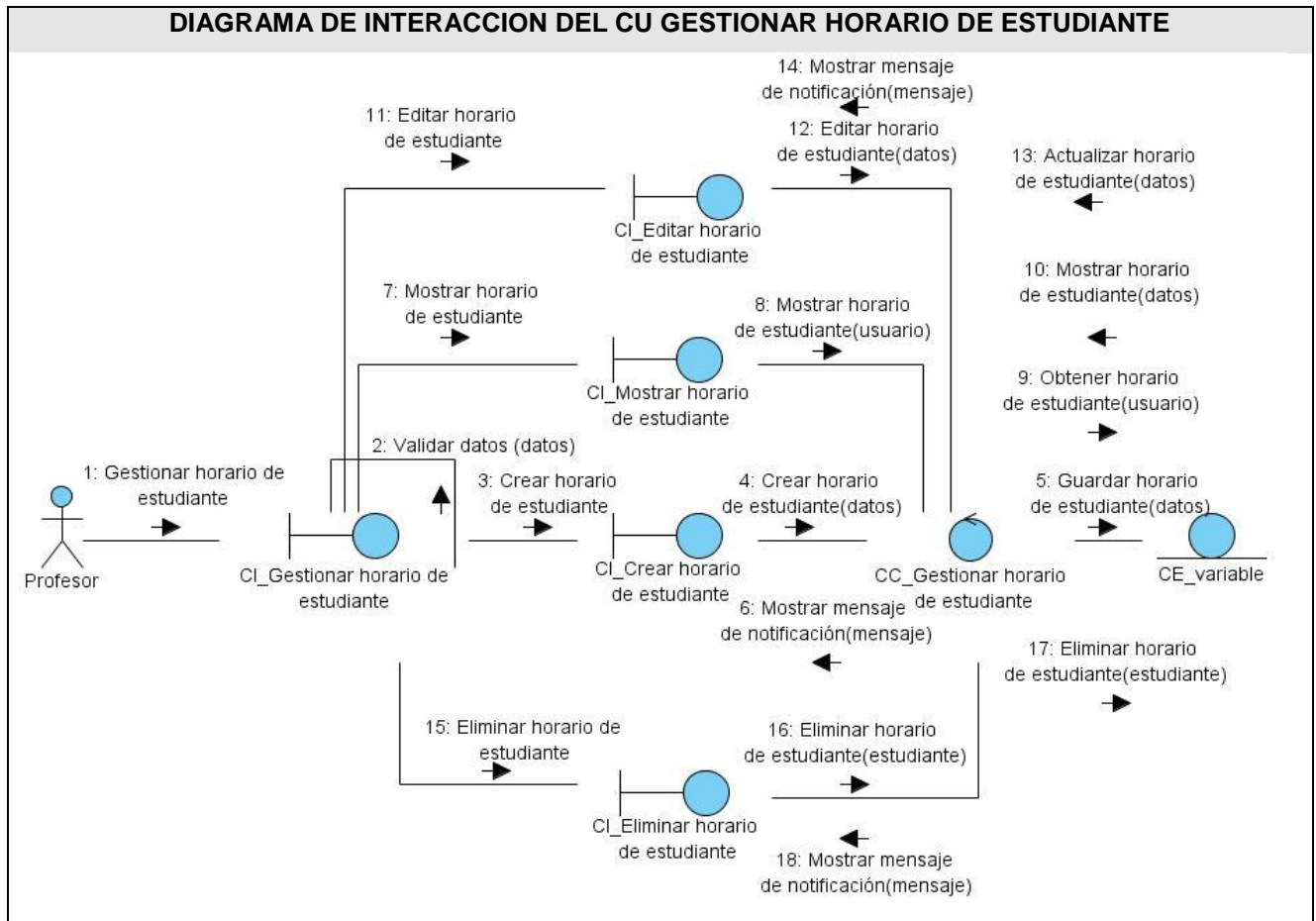
## CAPÍTULO III: ANÁLISIS Y DISEÑO DEL SISTEMA PARA EL CONTROL DE ASISTENCIA.

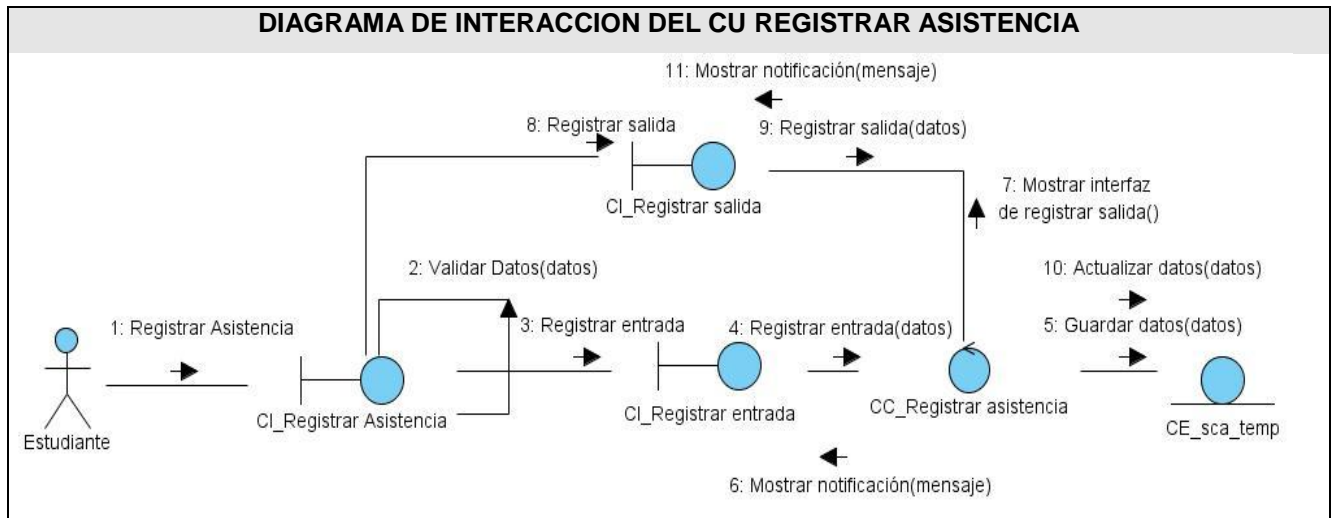
### 3.1. Modelo de Análisis

El Modelo de análisis es un modelo conceptual, ya que en él se identifican una serie de clases y relaciones que conectadas entre sí, permiten una mejor comprensión de los requisitos de la aplicación que se está modelando. Contiene clases del análisis y sus objetos organizados en paquetes que colaboran.

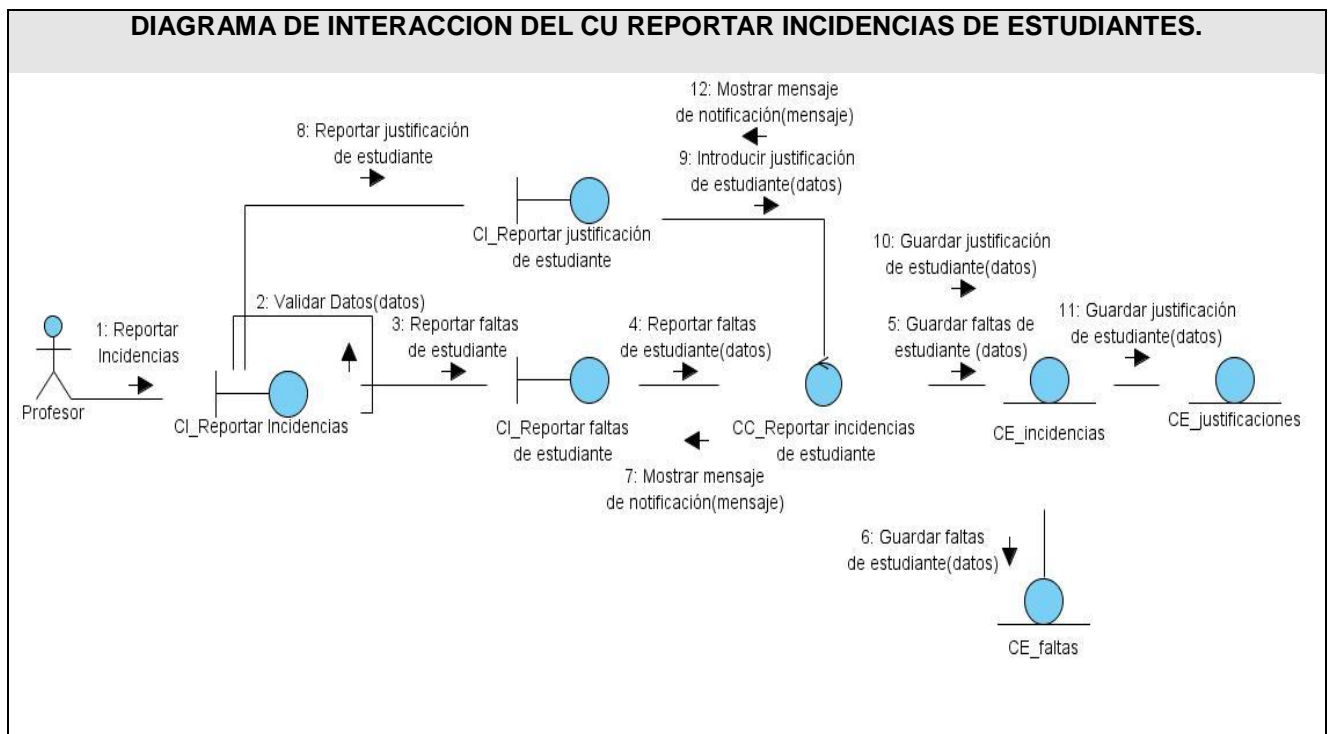
#### 3.1.1. Diagramas de interacción











### 3.2. Modelo de Diseño

A continuación se desarrolla el flujo de trabajo de diseño, a través del cual se modela el sistema que se pretende construir para que este sea capaz de soportar todos los requisitos. El diseño es el centro de atención final de la fase de elaboración y el comienzo de las iteraciones de construcción, no es más que el refinamiento del análisis, el cual contribuye a una arquitectura estable y sólida, además de crear un plano del modelo de implementación. Más tarde durante la fase de construcción, cuando la arquitectura es estable y los requisitos están bien entendidos, el centro de atención se desplaza a la implementación.

#### 3.2.1. Diagrama de clases del diseño Web

Drupal es un CMS que brinda una gran cantidad de funcionalidades, genéricas a cualquier tipo de proyecto, las cuales generan sus respectivos casos de usos. La propuesta se apoya en estas para lograr satisfacer los requisitos funcionales planteados. Para comprender el diagrama de clases del diseño web y cómo será implementado e integrado el módulo para el control de la asistencia, se explica la estructura de paquetes de Drupal, así como su funcionamiento. Drupal integra un conjunto de subpaquetes que conforman su núcleo (*Drupal Core*) y a la vez implementan sus funcionalidades básicas. Los mismos

están organizados en dos grupos principales:

Las **interfaces horizontales**: orientadas a la implementación de funcionalidades genéricas que son comunes para la propuesta y utilizadas horizontalmente en el resto de los módulos a través de una interfaz de aplicación (API):

- **Theme**: genera la interfaz gráfica que le proporciona el *Core* de Drupal y cada uno de los módulos utilizados, procesa la información y la visualiza en el portal.
- **Form**: integra las funciones para generar y controlar dinámicamente los formularios de la aplicación.
- **Database**: es la capa de acceso a datos, tramita el acceso y las acciones realizadas sobre la base de datos.

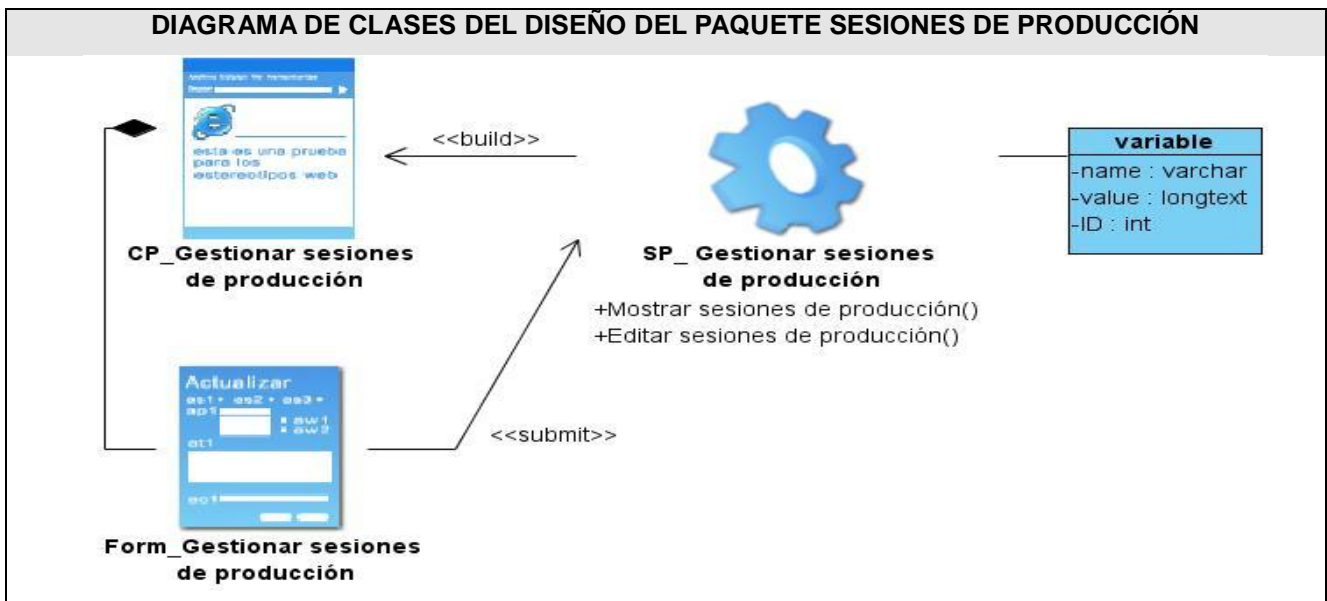
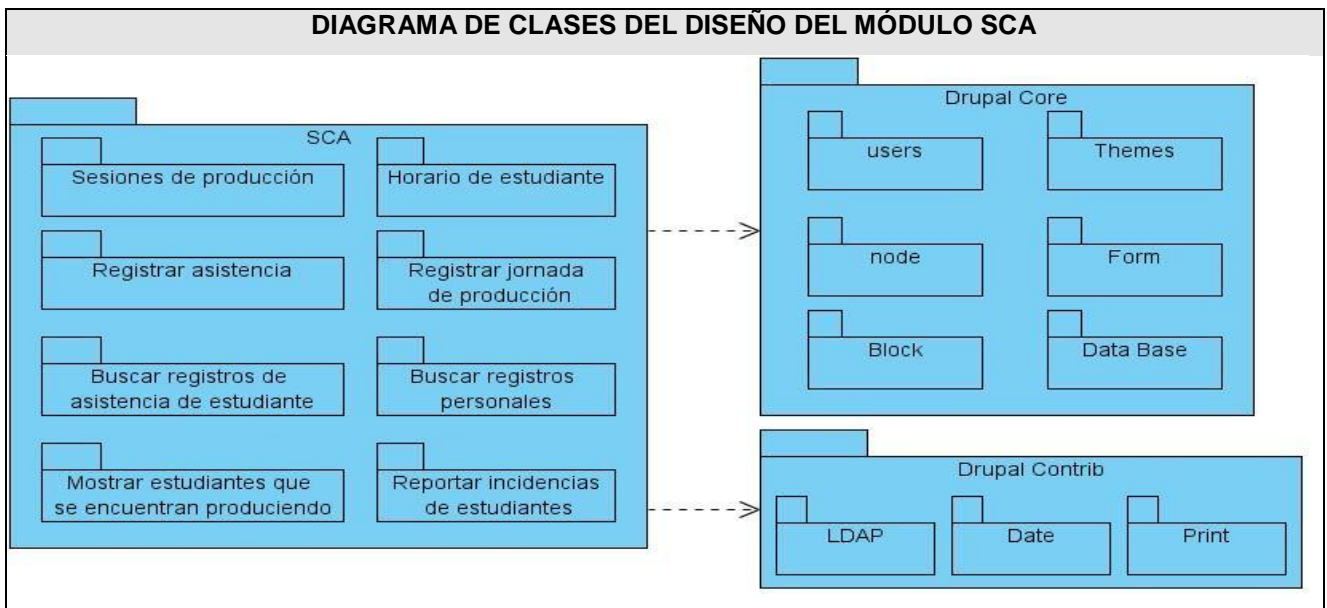
Los **módulos verticales**: orientados a la implementación de funcionalidades específicas:

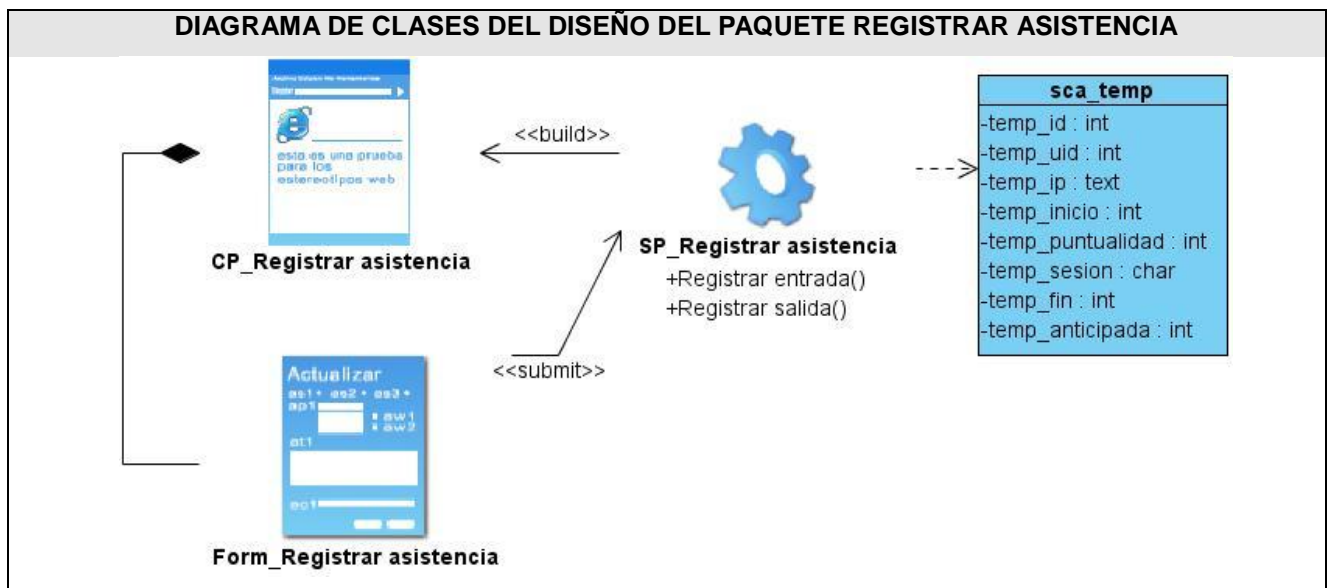
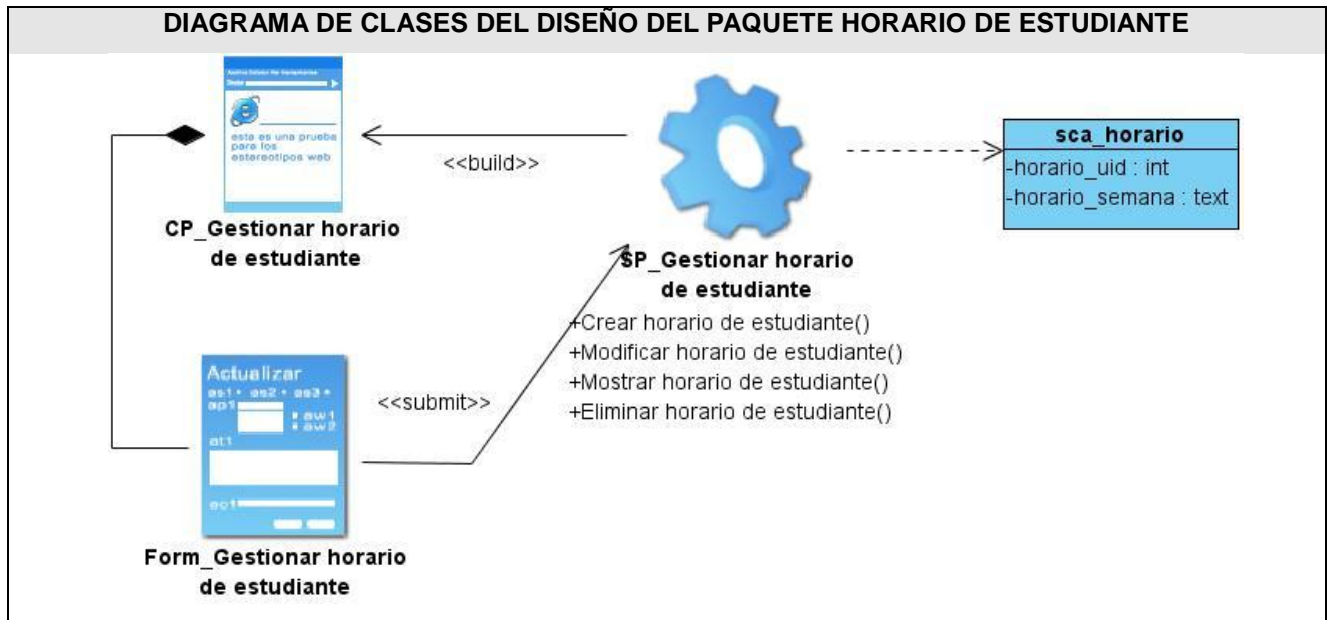
- **User**: gestiona los usuarios y roles del sistema, así como la autenticación de los mismos y sus permisos. (50)
- **Node**: provee una plataforma base para el proceso de gestión de contenidos, que puede ser extendida por otros módulos. (51)
- **Block**: permite configurar el contenido que aparece en bloques localizados en las barras laterales del sitio y otras regiones. (52)

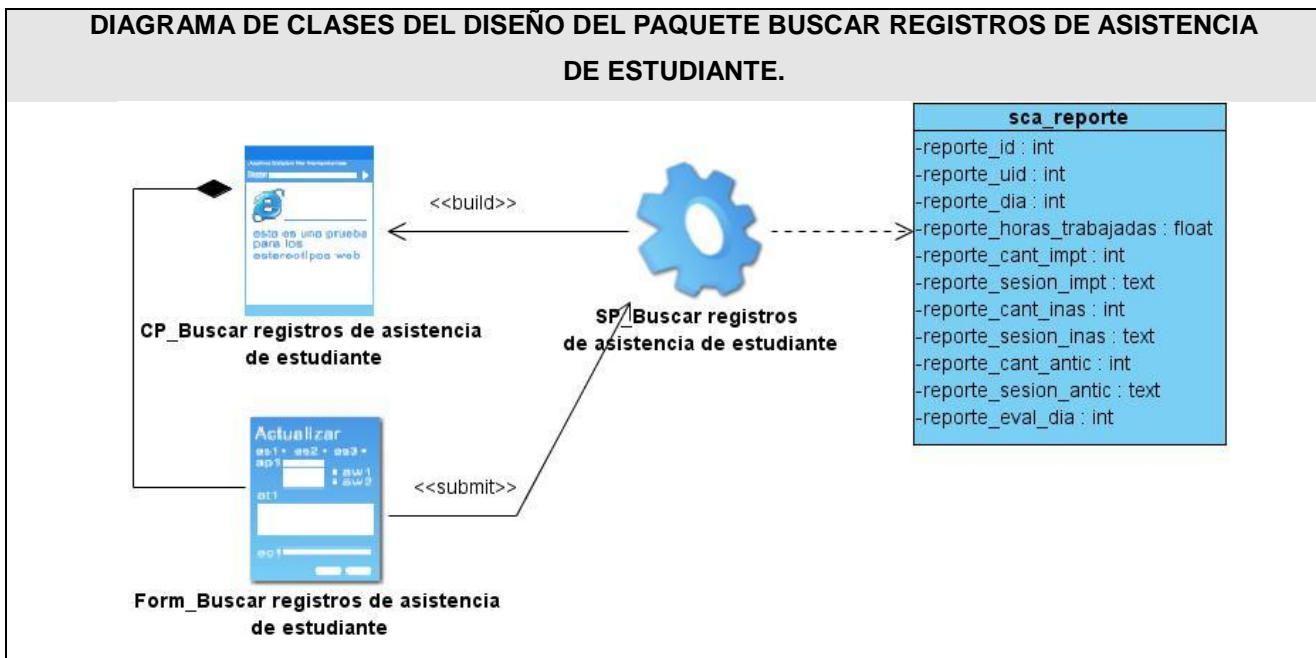
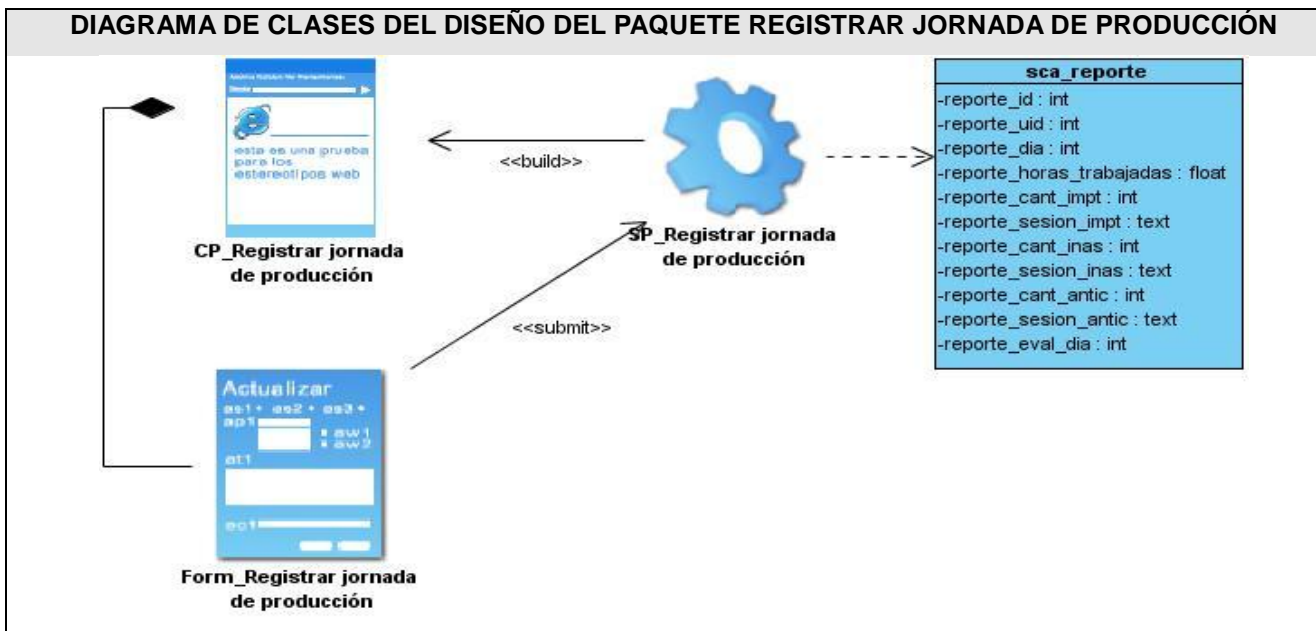
A su vez, los módulos que son adicionados al núcleo con el fin de incluir nuevas funcionalidades constituyen subpaquetes agrupados en el paquete *Drupal Contrib*. El sistema en cuestión hace uso de dos módulos en dicho paquete; los cuales son indispensable para el desarrollo de los aspectos novedosos de la aplicación, esto son:

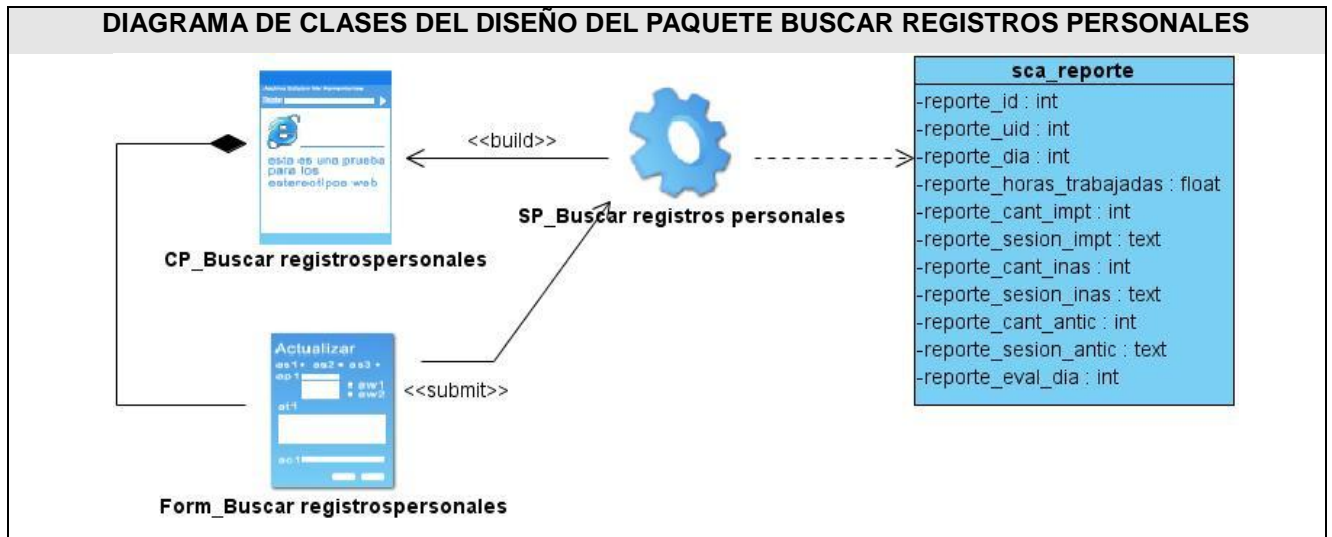
- **LDAP**: posibilita la integración con un servidor LDAP (*Lightweight Directory Access Protocol*), brindando la posibilidad a los usuarios de autenticarse en múltiples servidores LDAP o AD (*Active Directory*). (53)
- **Date**: permite mostrar cualquier campo de fecha en formato de calendario, incluidos los campos de fecha de CCK, propiciando de esta forma una factible selección del año, mes y día. (54)
- **Print**: añade un enlace de versión para impresora a las páginas con contenido y de administración, además de la capacidad de exportar páginas en forma de PDF. Para poder crear una versión en pdf se debe tener instalada una de las siguientes herramientas: wkhtmltopdf, TCPDF o dompdf, la propuesta hace uso de esta última. (55)

El módulo propuesto se nutre de funciones ya existentes en los paquetes *Drupal Contrib* y *Drupal Core*, para comprender de forma clara lo antes planteado se muestra a continuación, el diagrama de clases de diseño web de la aplicación y luego los diagramas de clases del diseño de los CU relacionados directamente con el Sistema para el Control de Asistencia.

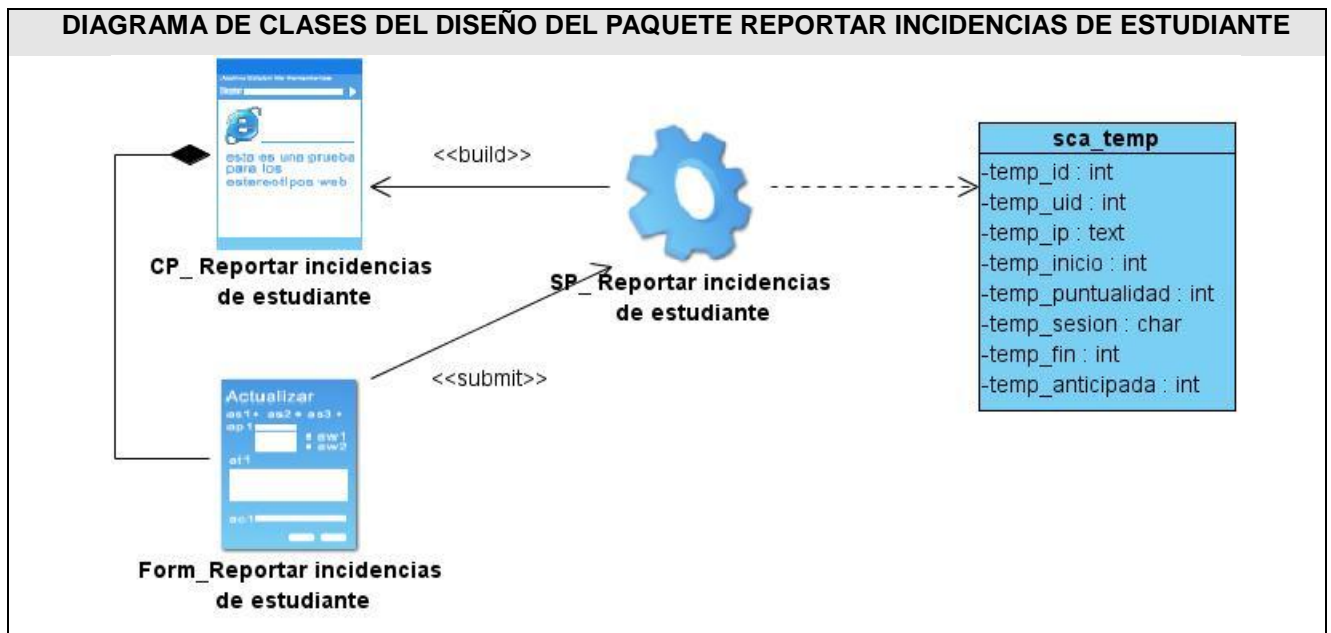












### 3.2.2. Diseño de la Base de Datos

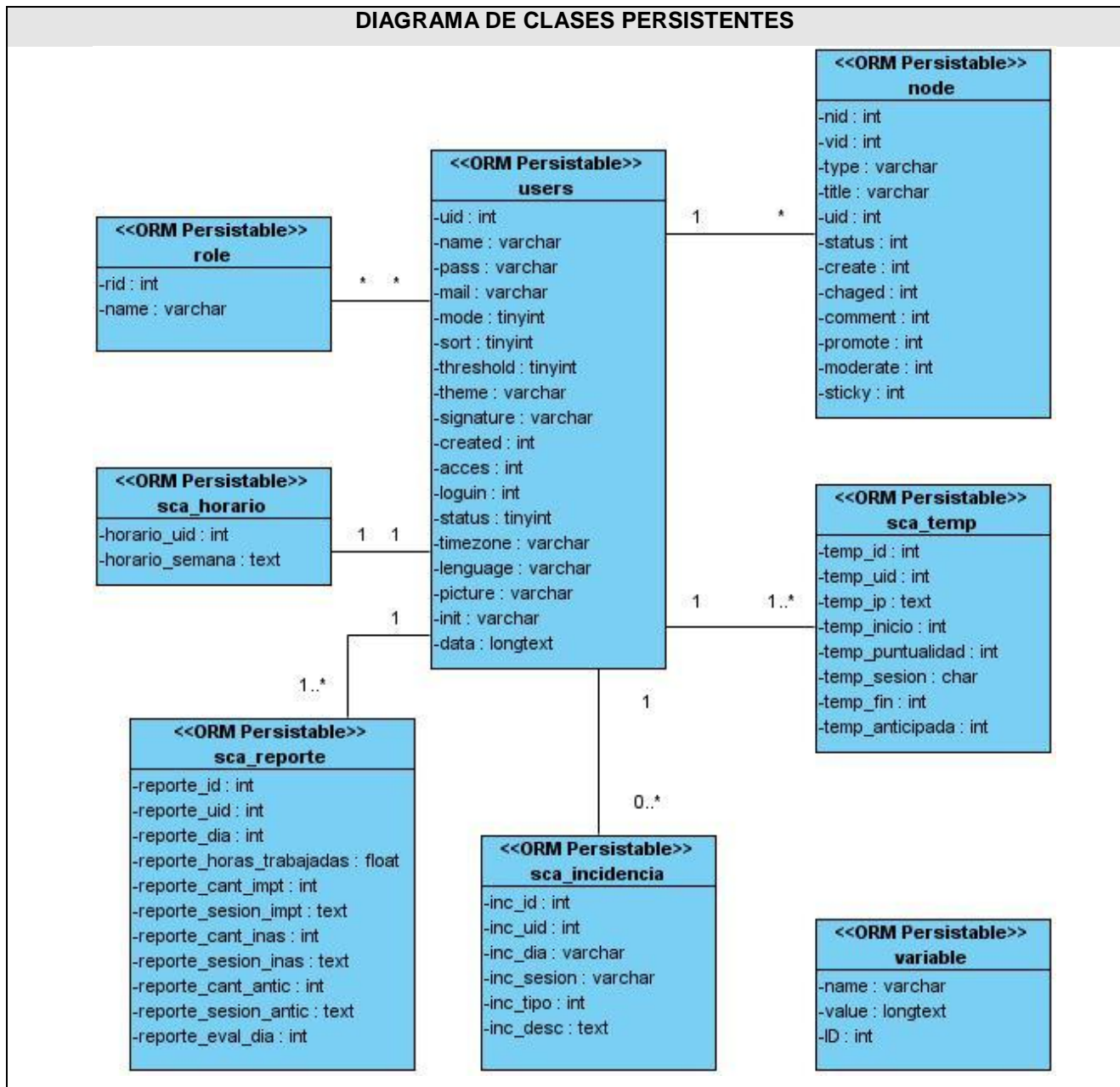
Las metas más importantes que se persiguen al diseñar un modelo de bases de datos, es dividir la información en tablas organizadas en temas para reducir los datos redundantes, proporcionar el acceso a la información necesaria para reunir la información de las tablas cuando así se precise, ayudar a garantizar la exactitud e integridad de la información y satisfacer las necesidades de procesamiento de los datos y de generación de informes. (Saldivar, G 2005)

Una base de datos correctamente diseñada permite obtener acceso a la información exacta y actualizada, por lo que una estructura adecuada es esencial para lograr los objetivos fijados para el sistema en cuestión. Independientemente de que la propuesta está basada en el CMS Drupal, el cual consta con una base de datos ya diseñada y probada por una comunidad internacional de varios miles de usuarios, se hará una representación de los principales aspectos para lograr un mejor entendimiento del funcionamiento de la aplicación.

En el diagrama de clases persistentes solo se contemplan las entidades de significativa relevancia para la aplicación, o sea, las clases que van a persistir en el tiempo. No será necesario el análisis de todas las entidades pues el CMS se encarga de la gestión de las mismas.

Para diseñar la Base de Datos es necesario primeramente realizar el diagrama de clases persistentes con sus atributos y relaciones.

### 3.2.3. Diagrama de clases persistentes



### 3.2.4. Modelo de datos de clases persistentes

A partir del diagrama de clases persistentes se define el modelo de datos que se muestra a continuación.





## CAPÍTULO IV: CONSTRUCCIÓN Y PRUEBA DE LA SOLUCIÓN.

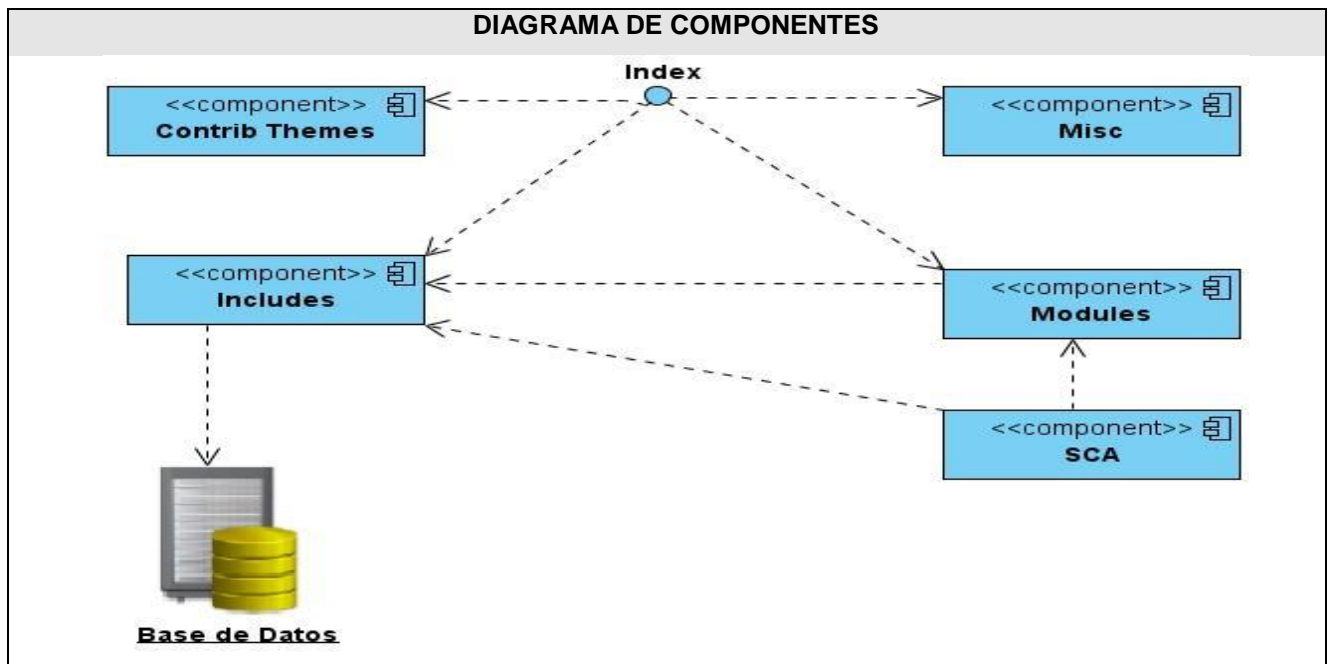
El presente capítulo contempla la implementación del Sistema para el Control de Asistencia integrado al CMS Drupal, a partir de los diagramas de despliegue y de componentes. Además, se determinan y realizan las distintas pruebas a las funcionalidades.

### 4.1. Diagrama de Componentes

Un diagrama de componentes muestra las organizaciones y dependencias lógicas entre componentes software, sean éstos componentes de código fuente, binarios o ejecutables.

A continuación se describe en una tabla los detalles del diagrama de componentes del sistema desarrollado:

Componente	Descripción
Contrib. Themes	Son los temas contribuidos de Drupal que no vienen incluidos por defecto en la instalación.
Misc	Aquí se encuentran una serie de archivos Javascript y CSS necesarios para el funcionamiento del sistema.
Modules	Aquí se localizan todos los módulos de Drupal.
Includes	Aquí se encuentran funciones y scripts necesarios para el funcionamiento de los módulos y el sistema Drupal, incluyendo las APIs de conexión a base datos.
Base de datos	Es la base datos de Drupal, dentro de la cual se incluyen las tablas del SCA.
SCA	Es el módulo para el control de asistencia a la producción.



#### 4.2. Diagrama de Despliegue

El diagrama de despliegue es un modelo de objetos que describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos de cómputo. Es una colección de nodos y arcos; donde cada nodo representa un recurso de cómputo, normalmente un procesador o un dispositivo de hardware similar. Muestra la configuración de los componentes hardware, los procesos, los elementos de procesamiento en tiempo de ejecución y los objetos que existen en tiempo de ejecución. En este tipo de diagramas intervienen nodos, asociaciones de comunicación, componentes dentro de los nodos y objetos que se encuentran a su vez dentro de los componentes. Un nodo es un objeto físico en tiempo de ejecución, es decir una máquina que se compone habitualmente de, por lo menos, memoria y capacidad de procesamiento, a su vez puede estar formada por otros componentes.

El nodo **Servidor de Base de Datos** representa un servidor MySQL, en el cual se ubica toda la información persistente del sistema, almacenándose los datos que son actualizados y consultados por los usuarios del Sistema para el Control de la Asistencia.

En el **Servidor web**, se ubican íntegramente las capas de presentación, lógica del negocio y de acceso a datos del sistema, así como los servicios que se brindan.

La **PC cliente** representa el conjunto de computadoras a través de las cuales los usuarios pueden actualizar y consultar la información que se encuentra en el Servidor web. Para acceder al Sistema de

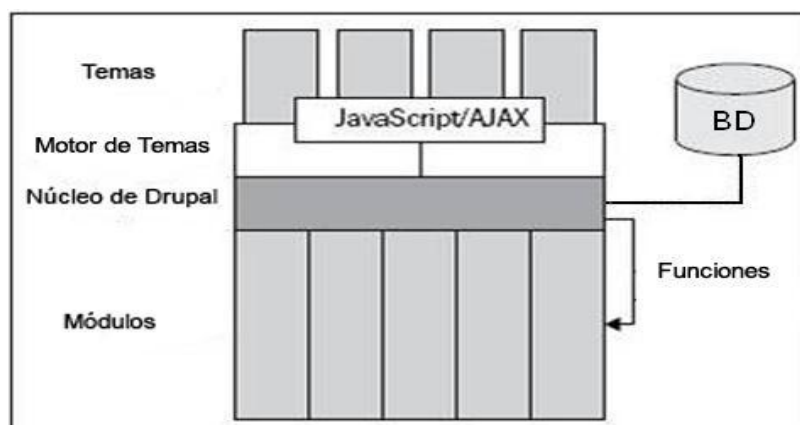
Control de Asistencia, las PC Clientes utilizan una aplicación cliente, en este caso, un navegador web. La comunicación entre las PC Clientes y el Servidor Web se establece utilizando el conjunto de protocolos de comunicación HTTP.



### 4.3. Arquitectura y Patrones empleados (56) (57)

El sistema implementado es un módulo para el CMS Drupal, por lo que la arquitectura y los patrones usados se heredan del mismo. A continuación se detalla la arquitectura y algunos de los patrones de Drupal. La arquitectura del CMS Drupal utiliza el patrón Modelo Vista Controlador (MVC). Este patrón de arquitectura de software separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. El patrón MVC se ve frecuentemente en aplicaciones Web, donde la vista es la página HTML y el código que provee de datos dinámicos a la página; el modelo es el Sistema de Gestión de Base de Datos y la Lógica de negocio; y el controlador es el responsable de recibir los eventos de entrada desde la vista.

Para lograr una mejor comprensión de cómo está estructurado arquitectónicamente el CMS Drupal se presenta la siguiente imagen:



Img 1. Patrón Modelo Vista Controlador del CMS Drupal.

Donde el modelo es la “Base de Datos (BD)”, la vista son los “Temas”, “Motor de Temas” y “Javascript/Ajax” y el controlador está compuesto por el “Núcleo de Drupal” y “Módulos”, donde “Funciones” son funciones que facilitan la comunicación entre el “Núcleo de Drupal” y los “Módulos”.

Drupal, a pesar de no utilizar muchas de las características de la Programación Orientada a Objetos, se debe aclarar que si hace un uso extensivo del Paradigma Orientado a Objetos, ya que es muy común pensar esto al no ver la palabra *class* en su código. El diseño de Drupal está altamente conformado a través de una orientación a objetos, mostrando características como: objetos y abstracción, además del encapsulamiento, polimorfismo y herencia, siendo estas tres últimas características las que hacen únicos a los Sistemas Orientados a Objetos. Para una mejor argumentación, a continuación se ejemplifica la forma en que estas características están presentes en Drupal:

**Objetos:** algunos de los más prominentes componentes de Drupal que pueden ser considerados objetos, son los módulos, temas, nodos y usuarios.

**Abstracción:** el sistema de ganchos de Drupal (*hook system*) es la base de su interfaz de abstracción.

**Encapsulamiento:** dado que el código de Drupal está basado en funciones y todas comparten el mismo *namespace*, este está dividido por el uso de prefijos. Siguiendo esta simple convención, cada módulo puede declarar sus propias funciones y variables sin la preocupación de los conflictos con los demás.

**Polimorfismo:** los nodos son polimórficas en el sentido clásico. La representación real de un nodo, sin embargo, dependerá de qué tipo de nodo se pasa a la función, que es directamente análoga a tener un objeto de la clase y determinar su comportamiento cuando se envía un mensaje a ella.

**Herencia:** los módulos y los temas pueden definir la función que deseen, pero a la vez pueden heredar su comportamiento. En los temas, si no sobrescriben una función definida, se utiliza la que posee por defecto.

Por otra parte, Drupal hace uso de patrones de diseño tales como: *singleton*, *decorador*, *observer*, *bridge*, *chain of responsibility* y *command*. Por solo mencionar uno, el caso del patrón **Singleton**: si se piensa en los módulos y temas como objetos, entonces ellos siguen este patrón. En general estos objetos no encapsulan datos, lo que separa un módulo de otro es el conjunto de funciones que contienen, por lo que se debe pensar en una clase con una única instancia. Cumpliendo así con este patrón, ya que su objetivo consiste en garantizar que una clase sólo tenga una instancia y proporcionar un punto de acceso global a ella. Se hace uso además, del patrón de distribución Cliente-Servidor, el cual define una estructura para el modelo de despliegue, de uso muy extendido en aplicaciones Web.

#### 4.4. Pruebas

El desarrollo del *software* implica una serie de actividades de producción, donde las posibilidades de que aparezcan errores son muy comunes. Los errores pueden comenzar desde el momento en que los objetivos son asimilados erróneamente, por tales motivos el desarrollo del *software* debe ir acompañado por una actividad que garantice la calidad. Las pruebas son una actividad en la cual un sistema o componente es ejecutado bajo ciertas condiciones, los resultados son observados y registrados, y una evaluación es hecha de algún aspecto del sistema o componente.

Con el objetivo de evaluar la calidad del producto desarrollado y garantizar que el módulo diseñado e implementado cumpla con las funcionalidades se diseñaron casos de prueba para realizar Pruebas de Caja Negra. Estos casos de prueba corresponden a los casos de uso que necesitan la entrada de datos y cada uno está dividido en tantas secciones como escenarios tenga el caso de uso, incluyendo los flujos de datos alternos.

##### Caso de Prueba para el caso de uso: Gestionar sesiones de producción

Entrada	Respuesta del Sistema	Resultados
El Jefe de Departamento deja algún campo de entrada o salida de las sesiones, vacío.	El sistema emite un mensaje de error para que el Jefe de Departamento no deje ningún campo vacío.	Satisfactorio.
El Jefe de Departamento introduce la hora de las sesiones en un formato incorrecto.	El sistema muestra un mensaje de error haciéndole saber al Jefe de Departamento que la hora está en un formato incorrecto.	Satisfactorio.
El Jefe de Departamento introduce las sesiones en un orden incorrecto.	El sistema muestra un mensaje de error notificando al Jefe de Departamento que el orden de las horas de entrada y salida es incorrecto.	Satisfactorio.
El Jefe de Departamento introduce la configuración correcta.	El sistema muestra un mensaje para notificar que la configuración se estableció con éxito.	Satisfactorio.

##### Caso de Prueba para el caso de uso: Gestionar horario de estudiante

Entrada	Respuesta del Sistema	Resultados
---------	-----------------------	------------



El Profesor deja el campo Usuario vacío.	El sistema emite un mensaje de error notificando al Profesor de que el campo Usuario es obligatorio.	Satisfactorio.
El Profesor introduce la configuración correcta.	El sistema muestra un mensaje para notificar que la configuración se estableció con éxito.	Satisfactorio.

**Caso de Prueba para el caso de uso: Registrar asistencia**

Entrada	Respuesta del Sistema	Resultados
El Estudiante accede al sistema en un horario en que le corresponde trabajar.	El sistema muestra el bloque titulado "Control de Asistencia" con el botón "Registrar Entrada".	Satisfactorio.
El Estudiante presiona el botón "Registrar Entrada" en un horario mayor que la hora de entrada de la sesión.	El botón desaparece y aparece en su lugar el botón de "Registrar Salida" y se muestra el mensaje "Ha registrado su entrada tarde".	Satisfactorio.
El Estudiante accede al sistema en un horario en que le corresponde trabajar y ya ha registrado su entrada previamente.	El sistema muestra el bloque titulado "Control de Asistencia" con la opción "Registrar Salida".	Satisfactorio
El Estudiante presiona el botón "Registrar Salida, en un horario menor que la hora de salida de la sesión.	El botón desaparece y se muestra el mensaje "Ha registrado su salida anticipadamente".	Satisfactorio
El Estudiante accede al sistema en un horario en que no le corresponde trabajar.	El sistema no muestra el bloque titulado "Control de Asistencia"	Satisfactorio.
El Estudiante presiona el botón "Registrar Entrada" a la hora correcta.	El botón desaparece y aparece en su lugar el botón de "Registrar Salida" y se muestra el mensaje "Ha registrado su entrada	Satisfactorio.

	correctamente”.	
El Estudiante presiona el botón “Registrar Salida” a la hora indicada.	El botón desaparece y se muestra el mensaje “Ha registrado su salida correctamente”.	Satisfactorio.

**Caso de Prueba para el caso de uso: Registrar jornada de producción**

Entrada	Respuesta del Sistema	Resultados
El Profesor accede al sistema en un horario menor que la hora de salida de la última sesión.	El sistema no muestra el bloque titulado “Registrar Jornada”.	Satisfactorio.
El Profesor accede al sistema cuando ya se ha registrado previamente esa jornada de producción.	El sistema no muestra el bloque titulado “Registrar Jornada”.	Satisfactorio.
El Profesor accede al sistema en un horario mayor que la hora de salida de la última sesión más 10 minutos de prórroga.	El sistema muestra el bloque titulado “Registrar Jornada”.	Satisfactorio.
El Profesor presiona el botón “Registrar Jornada”.	El botón desaparece y se muestra el mensaje “Se han registrado los datos de la jornada correctamente”.	Satisfactorio.

**Caso de Prueba para el caso de uso: Reportar incidencia de estudiante**

Entrada	Respuesta del Sistema	Resultados
Campos de configuración vacíos.	El sistema emite un mensaje de error para que el Profesor no deje ningún campo vacío.	Satisfactorio.
El Profesor al introducir los datos de una incidencia tipo falta, especifica una fecha diferente de la actual.	El sistema muestra un mensaje de error haciéndole saber al Profesor que las faltas solo se registran en el día actual.	Satisfactorio.



El Profesor, al introducir los datos de una incidencia tipo justificación, especifica una fecha anterior al día actual.	El sistema muestra un mensaje de error haciéndole saber al Profesor que las justificaciones solo se registran en fechas mayores o iguales que el día actual.	Satisfactorio.
El Profesor introduce la configuración correcta.	El sistema muestra un mensaje para notificar que la configuración se estableció con éxito.	Satisfactorio.

Se llevaron a cabo pruebas de seguridad con el objetivo de asegurar que el sistema solo fuera accedido por los actores definidos, se logró comprobar que el Jefe de Departamento es el único que puede modificar el horario de las sesiones de producción y el Profesor el único que puede gestionar el horario de los estudiantes, registrar las jornadas de producción y reportar las incidencias. A su vez tanto el Consultor como el Estudiante sólo pudieron acceder a las funcionalidades correspondientes a los casos de usos inicializados por ellos en el Diagrama de CU del Sistema.

Se instaló el módulo en distintas máquinas para probar el adecuado funcionamiento y almacenamiento de los datos correspondientes a los usuarios y la asistencia, así como el correcto cumplimiento de los requisitos no funcionales. El tratamiento de errores fue enfocado principalmente a los errores que pudieran ocurrir durante la interacción del usuario con el sistema, producto al desconocimiento o falta de experiencia en el área de la informática.

En el presente capítulo se han descrito actividades y artefactos correspondientes a los flujos de trabajo de Diseño y de Implementación de la metodología RUP, obteniéndose como resultados de los mismos el Modelo de Clases Persistentes, el Modelo de Despliegue y el Modelo de Implementación. Estos resultados son importantes para la implementación del sistema ya que rigen todos los procesos de desarrollo de software a la hora de implementar el módulo. Se hicieron cada una de las relaciones entre clases que interactúan con el sistema así como el diagrama de entidad-relación para mostrar las dependencias y referencias entre las tablas de la base de datos que se utilizaron a la hora de desarrollar el sistema. En el despliegue se llevaron a cabo los distintos tipos de comunicaciones que existen entre los dispositivos físicos de la aplicación y en el de componentes se controlan cada uno de los ficheros, ejecutables, páginas y archivos que fueron utilizados, así como la relación entre cada uno de estos para la ejecución del módulo.

## CONCLUSIONES

- En el diagnóstico del problema se detectó la necesidad de gestionar el control de asistencia vinculado a la producción de los estudiantes del Departamento de Gestión y Representación de Contenidos Web.
- A partir del análisis de aplicaciones existentes para el control de parámetros de horarios e incidencias laborales para entornos de escritorio y web, se determinó la creación de un nuevo sistema que respondiera a las necesidades y retos propuestos por el nuevo sistema de formación de la Universidad.
- Se implementó un módulo para Drupal que permitió el control de los elementos de asistencia, incidencias laborales y fue capaz de emitir evaluaciones parciales.
- La propuesta se desarrolló a través de un módulo para el CMS Drupal, que se integra al sitio del Departamento de GRCW.
- La implementación del módulo SCA, con el uso de herramientas libres, tiene un valor agregado por lo que representa para la soberanía tecnológica a que aspira Cuba en la rama de las Tecnologías de la Información y las Comunicaciones.

## RECOMENDACIONES

- Incorporar una interfaz de administración al módulo que permita configurar desde la aplicación criterios como: la cantidad de sesiones de trabajo, los días laborales, los rango de IP permitidos para acceder al sistema y la fórmula de evaluación en base a los parámetros de asistencia.
- Emplear información como nombres y apellidos y foto del usuario, proporcionada por el servidor LDAP de la UCI, para incorporarla a los reportes.
- Modificar la funcionalidad “Registrar Jornada”, para que esta se realice de forma automática, una vez culminada la última sesión de trabajo.
- Realizar un manual de usuario y de administración.

## BIBLIOGRAFÍA

1. **Engineering Support** . Computer Design Company - Engineering Support. *eTime Manual*. [Online] 09 26, 2002. [Cited: diciembre 05, 2009.] [http://www.engineeringsupport.eu/timesheets/e-time\\_timesheet\\_manual.pdf](http://www.engineeringsupport.eu/timesheets/e-time_timesheet_manual.pdf).
2. **Martínez, Yosvany Umpierre**. *Trabajo de diploma para optar por el título de Ingeniero Informático. Sistema de Control de Asistencia de trabajadores*. Ciudad Habana : Universidad de las Ciencias Informáticas, 2004.
3. **Design-soft**. Design-soft. *Sicon*. [Online] 2005. [Cited: diciembre 05, 2009.] <http://www.design-soft.com.pe/Sicon.htm>.
4. **Comunidad de Drupal**. Desoft. *Fastos*. [Online] 2008. [Cited: diciembre 07, 2009.] <http://www.desoft.cu/Productos1/FastosPagus/tabid/442/Default.aspx?PageContentID=23>.
5. **Rodríguez, Raúl Martínez**. "GREHU: Un sistema integral para gestionar los recursos humanos". [Online] 1998. [Cited: diciembre 08, 2009.] <http://www.gestec.disaic.cu/memorias/98/Trabajos/RaulMRguez.doc>.
6. **Universidad de las Ciencias Informáticas**. Sitio Oficial de la Universidad de las Ciencias Informáticas. [Online] 2010. [Cited: Diciembre 10, 2010.] <http://www.uci.cu/?q=node/55>.
7. **Cosialls, Delfí**. Servicio de Información.1-46 p. *Hechos, Datos e Información, en Gestión Clínica y Gerencial de Hospitales*. Madrid. : Ediciones Harcourt, S.A., 2000.
8. **Arellano, M. y Gamboa, T**. Revista Espacios digital. *Tecnologías y Sistemas de Información al servicio de la salud*. [Online] 2004. [Cited: enero 30, 2010.] <http://www.revistaespacios.com/a04v25n02/04250241.html>.
9. **Barros, Oscar**. *Tecnologías de la información y su uso en gestión. Una visión moderna de los sistemas de información*. Chile : Editorial Mc Graw-Hill, 1998.
10. **Asín, Daniel Cohen y Enrique**. *Sistemas de Información para los negocios. Un enfoque para toma de decisiones*. México : McGraw-Hill / Interamericana Editores, S.A., 2000.
11. **Mendiyl, Jesús José Espinosa**. SEMARNAT. Secretaría de Medio Ambiente y Recursos Naturales. *LINEAMIENTOS PARA EL REGISTRO Y CONTROL DE ASISTENCIA DEL PERSONAL OPERATIVO DE LA SEMARNAT*. [Online] noviembre 15, 2005. [Cited: enero 30, 2010.] <http://www.semarnat.gob.mx/leyesyformas/normateca/disposicioneseliminadas/Lineamientos%20Para%2>

- 0el%20Registro%20y%20Control%20de%20Asistencia%20del%20Personal%20Operativo%20de%20la%20Semarnat.pdf.
12. **DESIGN SOFT.** DESIGN SOFT. *Control de asistencia.* [Online] 2005. [Cited: enero 31, 2010.] [http://www.design-soft.com.pe/Control de asistencia.htm](http://www.design-soft.com.pe/Control%20de%20asistencia.htm).
  13. **Villegas, Sabino Ayala.** El prisma. *Proceso de registro y control de personal.* [Online] 2004. [http://www.elprisma.com/apuntes/administracion\\_de\\_empresas/controldepersonal/default.asp](http://www.elprisma.com/apuntes/administracion_de_empresas/controldepersonal/default.asp).
  14. **SPN. Red de Software.** SPN. Red de Software. *Descargar E-Time Español.* [Online] 2002. [Cited: diciembre 20, 2009.] [http://es.softpicks.net/software/Utilitarios/Herramientas-de-sistema/E-Time\\_es-166595.htm](http://es.softpicks.net/software/Utilitarios/Herramientas-de-sistema/E-Time_es-166595.htm).
  15. **Sistemas Paez.** Sistemas Paez. *Doce.* [Online] 09 2008. [Cited: enero 31, 2010.] <http://www.sistemaspaez.com/doce2/index.php>.
  16. **SecureTech A&M.** SecureTech A&M. *SmartClk.* [Online] 2010. [Cited: enero 31, 2010.] [http://www.securetech.com.uy/documentos/securetech/SmartClk 20V316.pdf](http://www.securetech.com.uy/documentos/securetech/SmartClk%20V316.pdf).
  17. **ArchivosPC.** ArchivosPC. *Exactus Pro.* [Online] 2006. [Cited: febrero 1, 2010.] <http://exactus-pro-2006.archivospc.com/>.
  18. **Exactus Pro.** Exactus Pro. [Online] 2006. [Cited: febrero 1, 2010.] <http://www.exactuspro.com/>.
  19. **Universidad de Zaragoza.** Universidad de Zaragoza. *Aplicación Web del Control Horario.* [Online] Julio 2001. [Cited: febrero 1, 2010.] [http://wzar.unizar.es/gerencia/pdf/control\\_horario.pdf](http://wzar.unizar.es/gerencia/pdf/control_horario.pdf).
  20. **fundacite-merida.** Fundación para el Desarrollo de la Ciencia y la Tecnología del estado Mérida (FUNDACITE). *Sistema de Control de Asistencia.* [Online] septiembre 12, 2007. [Cited: febrero 1, 2010.] [http://sistemas.fsl.fundacite-merida.gob.ve/softwaremap/trove\\_list.php?form\\_cat=237&discrim=237](http://sistemas.fsl.fundacite-merida.gob.ve/softwaremap/trove_list.php?form_cat=237&discrim=237).
  21. **MININT UCI.** Sistema de Control de Acceso a los laboratorios del modulo del MININT en la UCI. *Sistema de Control de Acceso a los laboratorios del modulo del MININT en la UCI.* Universidad de las Ciencias Informáticas (UCI) : s.n., 2010.
  22. **Assets.** Assets. *Módulo de Recursos Humanos.* [Online] 2004. [Cited: febrero 01, 2010.] <http://assets.co.cu/humanos.asp>.
  23. **ASSETS NS.** Archivo de ayuda. *ASSETS NS Sistema de Gestión Integral.*
  24. **Moreno, Nohemy Cardentey.** *Módulo de control de acceso de UCI-Lab.* Universidad de las Ciencias Informáticas, febrero 01, 2010.
  25. **Victoria.** Definición de HTML. *Definición ABC.* [Online] Enero 12, 2009. [Cited: Febrero 03, 2010.]

<http://www.definicionabc.com/tecnologia/html.php>.

26. **Maestros del Web.** ¿Qué es CSS? *Maestros del Web*. [Online] 2003. [Cited: Febrero 03, 2010.] <http://www.maestrosdelweb.com/editorial/introcscs/>.
27. Versiones de CSS. *blog: Aprendiendo CSS*. [Online] Diciembre 18, 2006. [Cited: Febrero 10, 2010.] Disponible en: <http://www.loretahur.net/AprendiendoCSS/versiones-de-css/>.
28. **Nieto Pérez, Iván.** Curso de JavaScript. [Online] 2008. [Cited: Febrero 03, 2010.] <http://www.elcodigo.net/tutoriales/javascript/javascript1.html>.
29. **The PostgreSQL Global Development Group.** Sitio oficial de PostgreSQL. *PostgreSQL 8.4.4 Documentation*. [Online] 2010. [Cited: Febrero 04, 2010.] <http://www.postgresql.org/docs/8.4/static/intro-what-is.html>.
30. **Marillera, Álvaro.** e-GHOST. *Instalación, configuración y manejo de MySQL*. [Online] 2010. [Cited: Febrero 04, 2010.] <http://www.e-ghost.deusto.es/docs/TutorialMySQL.html>.
31. **Robertson, J.** Sistemas de Gestión de Contenidos. “*What is a content management system?*”. [Online] 2004. [Cited: febrero 05, 2010.] [http://www.steptwo.com.au/papers/kmc\\_what/](http://www.steptwo.com.au/papers/kmc_what/).
32. **López Avellaneda, Daniel.** Introducción a los CMS. *La página web del centro con SPIP*. [Online] Marzo 8, 2008. [Cited: febrero 05, 2010.] <http://www.cepazahar.org/recursos/file.php/7/apuntes/1-Introduccion.pdf>.
33. **Westgate, J. VanDyk y M.** *Pro Drupal Development*. 2007.
34. **Drupal Comunity.** Drupal. *About Drupal by Community of Drupal*. [Online] 04 29, 2010. [Cited: febrero 05, 2010.] <http://drupal.org/about>.
35. **Reyero, Jose A.** Sobre Drupal. *Drupal.org*. [Online] Marzo 16, 2006. [Cited: febrero 06, 2010.] <http://www.drupal.org.es/drupal>.
36. **Educación Salesiana.** Educación Salesiana. *Glosario Salesiano*. [Online] 2008. [Cited: febrero 06, 2010.] <http://www.edusal.cl/moodle/mod/glossary/view.php>.
37. **Molpereces, Alberto.** WillyDev. *Procesos de desarrollo: RUP, XP, FDD*. [Online] 2003. [Cited: febrero 08, 2010.] <http://www.willydev.net/descargas/Articulos/General/cualxfddrup.PDF>.
38. **Mendoza Sanchez, María A.** Metodologías De Desarrollo De Software. *www.informatizate.net*. [Online] Junio 7, 2004. [Cited: Febrero 08, 2010.] [www.informatizate.net/articulos/pdfs/metodologias\\_de\\_desarrollo\\_de\\_software\\_07062004.pdf](http://www.informatizate.net/articulos/pdfs/metodologias_de_desarrollo_de_software_07062004.pdf).
39. **Wells, D.** Extreme Programming: A gentle introduction. *Sitio oficial de XP Extreme Programming*.

- [Online] Septiembre 28, 2009. [Cited: febrero 12, 2010.] <http://www.extremeprogramming.org/>.
40. **Jacobson, I, y otros.** El Proceso Unificado de Desarrollo de Software. *Catálogo en línea de la Biblioteca de la Universidad de las Ciencias Informáticas*. [Online] 2000. [Cited: Febrero 12, 2010.] <http://bibliodoc.uci.cu/pdf/reg00060.pdf>.
41. **José Joskowicz.** Instituto de Ingeniería Eléctrica. *Reglas y Prácticas en eXtreme Programming*. [Online] 2008. [Cited: febrero 12, 2010.] <http://iie.fing.edu.uy/~josej/docs/XP%20-%20Jose%20Joskowicz.pdf>.
42. **Brian C. Christensen** . Planner an open source project scheduling tool. *What is Planner?* [Online] 2003. [Cited: febrero 15, 2010.] <http://www.simpleprojectmanagement.com/planner/what.html>.
43. **KDE Web Dev Team.** Quanta Plus. *Welcome to the Home of Quanta Plus* . [Online] 2005. [Cited: febrero 25, 2010.] <http://quanta.kdewebdev.org/index.php>.
44. **infosertec.** Infosertec Web 2.0. *Sun Microsystems presenta NetBeans IDE 6.8*. [Online] diciembre 17, 2009. [Cited: febrero 26, 2010.] <http://www.infosertec.com.ar/blog/?p=12228>.
45. **Web Development Evolved.** Firebug. *Web Development Evolved*. [Online] 05 2010. [Cited: febrero 28, 2010.] <http://getfirebug.com>.
46. **Martin Maldonado, Daniel.** Apache, el servidor Web más reconocido. *aplicacionesempresariales.com*. [Online] Abril 20, 2008. [Cited: marzo 03, 2010.] <http://www.aplicacionesempresariales.com/apache-el-servidor-web-mas-reconocido.html>.
47. **Félix, Alvaro del Castillo San.** Entorno de acs. *El servidor de web Apache: Introducción práctica*. [Online] 2000. [Cited: marzo 04, 2010.] <http://acsblog.es/articulos/trunk/LinuxActual/Apache/html/index.html>.
48. **UML tool, business process modeler and database designer for software development team.** Boost Productivity Whith Innovative and Intuitive Tecnologies. *Visual Paradigm for UML*. [Online] 2010. [Cited: febrero 4, 2010.] <http://www.visual-paradigm.com/>.
49. **Jacobson, Ivar.** *The Roadmap to the Unified Software Development Process*. s.l. : Cambridge University Press-SIGS Books, 2000.
50. **Drupal Comunity.** Drupal. *User: access and management settings*. [Online] mayo 04, 2010. [Cited: marzo 15, 2010.] <http://drupal.org/handbook/modules/user>.
51. **Drupal Comunity.** Drupal. *About nodes*. [Online] mayo 03, 2010. [Cited: marzo 18, 2010.] <http://drupal.org/handbook/modules/node>.

52. **Drupal Comunity.** Drupal. *Working with blocks (content in regions)*. [Online] abril 22, 2010. [Cited: marzo 18, 2010.] <http://drupal.org/handbook/modules/block>.
53. **Drupal Comunity.** Drupal. *LDAP*. [Online] marzo 29, 2009. [Cited: abril 05, 2010.] [http://drupal.org/handbook/modules/ldap\\_integration](http://drupal.org/handbook/modules/ldap_integration).
54. **Clever Age.** Drupal. *Modulo Date*. [Online] 2006. [Cited: abril 18, 2010.] [http://drupal.org/project/publication\\_date](http://drupal.org/project/publication_date).
55. **Joao Ventura.** Drupal. *Printer, e-mail and PDF versions*. [Online] Noviembre 2, 2004. [Cited: abril 20, 2010.] <http://drupal.org/project/print>.
56. **Marlon Lara Pérez, Leover Armando González Rodríguez.** *Sistema de Gestión de la Vigilancia Tecnológica en el Ministerio de la Informática y las Comunicaciones*. CI UDAD DE LA HABANA, CUBA : s.n., 2009.
57. **Developers and coders.** Drupal. *Drupal programming from an object-oriented perspective*. [Online] 12 25, 2009. [Cited: mayo 05, 2010.] <http://drupal.org/node/547518>.