

**Universidad de las Ciencias Informáticas
Facultad 15**



**Título: Sistema para el control de pases extras
de los estudiantes de la Facultad 15.**

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

Autor: Dairely De la Cruz San Juan
Tutor: Ing. Henry Raúl González Brito

Julio 2010

Declaración de Autoría

DECLARACION DE AUTORIA

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

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

Dairely de la Cruz San Juan

Ing. Henry Raúl González Brito

Firma del Autor

Firma del Tutor

AGRADECIMIENTOS

A la revolución por brindarnos la posibilidad de ser alguien en la vida.

A la Universidad de las Ciencias Informáticas por forjarnos como profesionales y personas de bien.

A unas de las personas que más quiero y que desde que empecé la Universidad siempre han estado a mi lado en los buenos y malos momentos, me apoyaron en estos 5 y 6 años, con el apoyo de ellos he logrado estar aquí en estos momentos, para esas dos personas maravillosas que quiero que con su apoyo este trabajo ha salido a delante a mi madre y Oscar.

A mi abuela Miriam por creer y tener fe en mí.

A mi abuelo Bernardo lo quiero mucho

A mis tíos Kenia, papito, Fina.

A mis primos José Alberto, Yulissa Yordanis.

A Milton por cuidar de mi mama y apoyarme.

A mi gran y querido hermano Yunaikel.

A mi sobrino Sandy.

A Iris por ayudarme, apoyarme y tener paciencia conmigo.

A mi tutor y mi oponente por ser dos profesionales excelentes y apoyarme en el trascurso del trabajo, Henry y Haydee

A mi tribunal especialmente al profesor Rafael.

A mis amistades de la uci que han estado conmigo estos 5 años por mencionar algunos, los 4 fantásticos Alexis, Ricardo, Carlos, Lázaro, las amistades del apartamento, las del grupo viejo y nuevo, a mis amigos especialmente a mi hermanito Adrielito, mis amigas y hermanas Daineris, Lismary, Mabel, Milena y todas y todos aquellos que de algún modo me han apoyado y me han dado fuerza para seguir dando todo de mi, cada uno de ustedes saben que los quiero mucho.

Muchas Gracias

DEDICATORIA

A Josefa San Juan Vidal por esta en los bueno y malos momentos de mi vida, por estar siempre presente cuando la necesitaba, por dar todo de ella para yo ser la persona que tanto ella ha querido, a ti madre por ser tan maravillosa.

Dairely

RESUMEN

La aplicación que da título a este Trabajo de Diploma consiste en la creación de un sistema automatizado para el procesamiento de manera eficaz y organizada de los elementos involucrados en la reservación de los pases extras de los estudiantes de la Facultad 15 de la Universidad de las Ciencias Informáticas, brindando además la posibilidad de elaborar los reportes correspondientes.

No se ha realizado en la Universidad de las Ciencias Informáticas un Sistema que resuelva los problemas que existen a la hora de gestionar los pases extras de los estudiantes.

El objetivo es la realización del análisis, diseño e implementación de un prototipo funcional de un sistema informático que permita realizar solicitudes de pases y así lograr una organización del proceso, un acceso más rápido a la información por parte de los usuarios y del decanato de la Facultad y conocer las estadísticas relacionadas para utilizar los recursos de una forma óptima y realizar la solicitud del pase de una forma más eficiente. Todo esto permitirá un mayor control de esta información a partir de reportes y restricciones asociadas al negocio, que facilitaría a la administración una mayor gestión de todos los elementos asociados con la salida de los estudiantes, alimentación, asistencia a clases, etc.

Palabras claves:

Análisis, estudiantes, Facultad, pases extras.

DECLARACION DE AUTORIA	II
AGRADECIMIENTOS	III
DEDICATORIA	IV
RESUMEN.....	V
INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	4
1.1. Introducción	4
1.2. Sistemas automatizados existentes vinculados al campo de acción.....	4
1.3. Herramientas y entornos de desarrollo utilizados.....	11
1.4. Conclusión.....	17
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA	18
2.1. Introducción	18
2.2. Problema y situación problemática	18
2.3. Objeto de informatización	18
2.4. Información que se maneja.....	18
2.5. Propuesta del sistema	18
2.6. Modelo del negocio.....	19
2.7. Especificación de los requisitos de software.....	20
2.8. Diagrama de caso de uso del negocio	23
2.9. Descripción de los casos de uso del negocio.....	23
2.10. Diagrama de actividades	26
2.11. Diagrama de clase del modelo de objeto	27
2.12. Definición de los actores del sistema.....	27
2.13. Definición de los casos de uso del sistema.....	28
2.14. Diagrama del caso de uso del sistema	32
2.15. Conclusiones	32
CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA.....	33
3.1. Introducción	33
3.2. Modelo del análisis	33
3.3. Diagramas de clases de análisis por cada caso de uso.....	33
3.4. Diseño	36
3.5. Diagrama de clases del Diseño	36
3.6. Diseño de la Base de Datos	42
3.7. Diagrama entidad relación.....	45
3.8. Diagrama modelo de datos.....	46
3.9. Diagrama Modelo Lógico	47
3.10. Diagrama Modelo Físico	48
3.11. Descripciones de las tablas de la base de datos	49
3.12. Conclusiones	51
CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA	52
4.1. Introducción	52
4.2. Implementación	52
4.3. Diagrama de despliegue.....	52
4.4. Diagrama de componentes.....	53
4.5. Modelo de Prueba	53

Índice

4.6. Conclusiones.....	55
CONCLUSIONES.....	56
RECOMENDACIONES.....	57
TRABAJOS CITADOS.....	58
BIBLIOGRAFÍA.....	60
ANEXOS.....	62
GLOSARIO DE TÉRMINOS.....	65

INTRODUCCIÓN

A partir de la idea expresada por nuestro comandante en jefe Fidel Castro Ruz de crear una universidad única en su tipo, nace la Universidad de las Ciencias Informática (UCI), la cual tiene como misión, la formación de jóvenes profesionales, comprometidos con su país y altamente calificados en la rama de la informática.

En la UCI se desarrollan varios proyectos en el ámbito de la informática y las comunicaciones que pretenden informatizar al máximo todos los procesos, no solo relacionados con la actividad educacional dentro de la universidad, sino en el entorno de la sociedad, la cual será el punto de partida para la informatización del país.

Desde la creación de la universidad en el 2003-2004 se les ha dado la posibilidad a los estudiantes de salir de pase los fines de semana hacia todos los municipios de Ciudad de la Habana. Este proceso era muy difícil de gestionar, pero gracias al sistema de reservación de pase, la Dirección de Transporte alcanzó un nivel de planificación y gestión adecuado.

Para poder salir de pase, el estudiante debe recoger un comprobante que especifica el nombre, apellidos, número de carnet de identidad, número de solapín, ruta de ida, ruta de regreso, municipio de destino. Esta reservación solo es realizada de sábado para domingo. El proceso para que el estudiante pueda solicitar un pase extra para algún día entre semana todavía no se encuentra automatizado, lo que provoca que exista una situación problemática expresada a continuación:

Para que un estudiante pueda solicitar un pase para un día entre semana, tiene que pedirle al profesor guía una autorización, luego el profesor guía le manda una notificación al vicedecano con el nombre del estudiante y los motivos por el cual solicita el pase. Este proceso es engorroso ya que le ocupa espacio y organización al profesor y al vicedecano en su correo. El vicedecano cuando tenga la notificación, verifica si le puede realizar el pase al estudiante teniendo en cuenta las actividades que se realizan en la escuela, al ser aceptado el pase, el estudiante va al vicedecanato para recogerlo. Una vez elaborado el boletín de pase tiene que validarlo con la firma de la decana y el cuño de la Facultad.

Cuando un estudiante solicite un pase, el vicedecano tiene que archivar todos los datos correspondiente al estudiante, como el nombre, los apellidos, el número de solapín, grupo, fecha de entrada, fecha de salida, los motivos por los cuales está solicitando el pase y a la provincia o municipio al que se dirige.

Para dar un reporte a cualquier directivo de la Facultad acerca del estudiante, el vicedecano tiene que buscar todos los estudiantes que están de pase junto con sus datos, esta situación le es incómoda y le

influye en su trabajo, por lo que es un proceso lento a la hora de atender a cada profesor que le pida un reporte del estudiante que no se encuentra en la universidad.

Se necesita desarrollar una nueva aplicación que sea capaz de agilizar el proceso y que se obtenga un buen funcionamiento. Para lograrlo se ha propuesto la realización del análisis, el diseño y la implementación de un prototipo funcional de un sistema para la reservación de pase extras de los estudiantes de la Facultad 15.

Campo de acción

Sistemas informáticos para el control de las reservaciones de pases extras de la Facultad 15.

Considerando lo anterior este trabajo tiene como **objetivo general**:

Desarrollar el análisis y diseño del sistema para el control de pases extras de los estudiantes de la Facultad 15.

Derivando de este objetivo tenemos los siguientes **objetivos específicos**:

- Realizar un estudio de sistemas informáticos relacionados con la gestión de pases extras y de capital humano.
- Desarrollar los artefactos del análisis.
- Desarrollar los artefactos del diseño.
- Desarrollar los artefactos de la implementación del prototipo funcional.

Con el propósito de dar respuesta a los objetivos antes planteados se trazaron las siguientes **tareas a desarrollar**:

- Realización de una entrevista a la vicedecano de formación de la Facultad 15 para un levantamiento de requisitos y ver como se quiere que funcione el sistema para la reservación de pases extras de los estudiantes de la Facultad 15.
- Selección y revisión bibliográfica de temas relacionados con el campo de acción.
- Implementación de un prototipo funcional que siga las necesidades establecidas para las salidas de los estudiantes días entre semana.
- Validación de la propuesta.

La **hipótesis** si se concreta el desarrollo de un sistema para el control de pases extras de los estudiantes de la Facultad 15 se optimizarán los procesos actuales para el control de pases extras de los estudiantes.

Al concluir con la investigación se espera obtener un modelo informático y un prototipo funcional del proceso de otorgamiento de pases extras en la Facultad 15. Este aporte contribuirá a la informatización de este proceso el próximo curso.

El trabajo está estructurado por 4 capítulos donde a continuación se hace una explicación de lo que trata cada uno.

Capítulo 1. Fundamentación teórica

En este capítulo se abordará del estado del arte donde se analizarán los sistemas existentes en el ámbito mundial, nacional y en la universidad. Además se describirán las herramientas utilizadas para el desarrollo de la aplicación.

Capítulo 2. Características del sistema.

En este capítulo se obtendrá un estudio del modelo del negocio, donde se le dará a comprender el problema presentado, se especificarán los requisitos del software, y se modelaran las características básicas del sistema.

Capítulo 3. Análisis y diseño del sistema.

Se definirá en este capítulo los diagramas de clases del análisis y diseño que se aplica y el modelo de la base de datos, con las tablas y sus descripciones.

Capítulo 4. Implementación y prueba.

En el capítulo 4 se presentarán los principales diagramas de implementación, diagramas de despliegue y de componentes, la descripción de los casos de prueba de integración que se llevan a cabo sobre el sistema.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1. Introducción

En el presente capítulo se hará un estudio referente a las aplicaciones existentes relacionadas con el sistema que se desarrollará. Además se describen y se valoran las propuestas de algunas herramientas y tecnologías para el desarrollo de la aplicación.

1.2. Sistemas automatizados existentes vinculados al campo de acción.

Uno de los más grande propósitos de Cuba y del mundo es la automatización de empresas, las cuales llevan a cabo toda la gestión y control de las personas que de una forma u otra trabajan o son internos de las mismas. Los recursos humanos se ocupan de seleccionar, contratar, formar, emplear y retener a los colaboradores de la organización. Estas tareas las puede desempeñar una persona o departamento en concreto junto a los directivos de la organización.

1.2.1. Sistemas de recursos humanos en el ámbito internacional

1.2.1.1. Tarjetas de Proximidad - Inner Plus.

Inner Plus es una herramienta moderna, creada para optimizar procesos internos en empresas de todos los tamaños. Este software es un colector de datos que permite una eficiente marcación de asistencia y control de acceso de sus empleados.

Inner Plus fue desarrollado para las empresas modernas que buscan automatización con la mejor relación costo-beneficio. El equipo funciona a través de la lectura de datos de tarjetas con código de barras, tarjetas magnéticas, tarjetas inteligentes, proximidad o hasta por el teclado. Las informaciones son recogidas, procesadas y enviadas a un microcomputador.

La multifuncionalidad de Inner Plus permite el accionamiento de sirenas, cerraduras eléctricas y barreras, entre otros dispositivos. Unas de las empresas que distribuye y utiliza este producto es Topdata.

Aplicaciones

- Control de asistencia de los empleados
- Control de producción
- Control de acceso

- Control del comedor
- Monitor de cristal líquido de 2 líneas y 16 columnas.
- Teclado numérico y de funciones.
- Control simultáneo del accionamiento de sirenas y de aperturas de puertas.
- Memoria no-volátil que mantiene los datos colectados y el funcionamiento del reloj interno por hasta un año en caso que haya falta de energía (TOPDATA Sistemas de Automatización, 2007)

1.2.1.2. Control de Asistencia y Accesos Biométrico - T1000 Bio.

El T1000 Bio es una garantía de eficiencia en el control de acceso a ambientes restringidos, dispensa el uso de tarjetas y señas identificando los usuarios por medio de su huella digital, característica única e intransferible de cada persona, fue la opción elegida por una de las mayores empresas de alimentación en Grecia para control de asistencia de sus empleados.

- Evita el uso de tarjetas.
- No necesita de conexión al microcomputador para operar.
- Kit completo incluyendo colector de datos, cable de comunicación y software para control de asistencia.
- Tiene accionamiento para control de puertas y barreras.

Especificaciones Técnicas:

- Reloj de control.
- Sensor óptico para verificación de la huella digital.
- Acepta hasta 100 usuarios.
- Lista de control de acceso.
- Puede controlar el accionamiento de sirenas y la apertura de puertas.
- Permite visualización de los registros en el propio monitor.
- Puede ser configurado para aceptar marcaciones apenas en las zonas de horarios definidas.
- Pueden ser registrados hasta 16 horarios para alarma (señalización de entrada y de salida de turnos de los trabajadores del centro).
- Cambio automático del horario en el inicio y fin del horario de verano.
- Configuración y colecta de datos.

- Dispensa el uso de cables, almacenamiento de hasta 4.000 marcaciones.
- Configuración del reloj.
- Emisión de diversos reportes.
- Asistencia.
- Totalización de las horas trabajadas, extras, faltas. (TOPDATA Sistemas de Automatización, 2007)

1.2.1.3. Sistema I-Soft-ERP

El Sistema Integrado INTELLISOFT (i-Soft), es una solución informática orientada a la gestión del negocio de empresas de diversos tipos: privadas, gubernamentales y no gubernamentales, de diversos tamaños: grandes, medianas y micro empresas.

I-Soft es una herramienta que está orientado fundamentalmente a producir información analítica y de gestión con valor agregado en el proceso de negocios de la empresa. Se encuentra en continuo crecimiento para satisfacer los requerimientos funcionales del rápido cambio del entorno y la tecnología que impone a las empresas, incorporando para ello procesos con los patrones y soluciones de calidad más reconocidos. I-Soft, permite registrar, revisar y controlar el flujo de información de toda la cadena de valor de la empresa en forma coordinada, agilizando la operatividad y consolidación de la información en una base de datos empresarial centralizada. I-Soft, está desarrollado con tecnología que permite la vigencia en el tiempo, está integrado por módulos complementarios e independientes que se implementan de acuerdo a las necesidades del cliente para cubrir el alcance de su negocio y presupuesto. Este sistema es utilizado en países como Panamá, Brasil, ya que es un sistema exclusivo para empresas grandes y medias.

Posee los siguientes módulos:

Contabilidad general

- Registro de comprobantes.
- Estados financieros.

Compras

- Emisión de órdenes de compra.
- Niveles de aprobación.
- Compras de bienes y servidores.

Cuentas por pagar

- Registro de obligaciones.
- Gestión de obligaciones por pagar.

Ventas y facturación

- Cotizaciones.
- Ingresos de pedidos.

Cuentas por cobrar

- Registros de deudores.
- Registro y provisión de documentos por cobrar.

Inventarios

- Catálogo de artículos.
- Manejo de múltiples almacenes.

Tesorería

- Control de movimientos bancarios.
- Emisión automática de cheques.
- Conciliación bancaria. (Intellisoft SA, 2007)

1.2.2. Sistemas de recursos humanos en el ámbito nacional y en la Universidad de las Ciencias Informáticas

1.2.2.1. Sistema de Recursos Humanos (Fastos).

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. Este sistema realizado por Desoft es utilizado en Cuba teniendo algunos tipos de áreas tales como:

Registro de los empleados:

Se guardan los datos de los empleados, así como informaciones referente a los reporte de vacaciones, certificados médicos, licencias, resolución.

- Control de la plantilla:

Permite establecer la estructura organizativa de las plazas de la entidad.

- Control de asistencia:

Lo cual incorpora el control de claves de asistencias, turnos de trabajos, horarios, tarjeta de asistencia e incidencias de cada empleado. Permite acoplar relojes para actualizar la información de la tarjeta de asistencia de forma automática.

- Informes y modelos:

Permite obtener un total de 56 informes, por ejemplo cierre del periodo, análisis de fondo de tiempo, estadísticos.

- Control de la capacitación respecto a:

Estudios Realizados, Cursos, Eventos, Experiencia docente, Publicaciones, Conocimientos, Idiomas extranjeros, Plan de desarrollo, e informes

- 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. (Desoft S.A., 2008)

1.2.2.2. Gestión Integral Aduanal (GINA)

Dentro de GINA está el módulo de recursos humanos que se encuentra en la evaluación del desempeño de los trabajadores de la Aduana General de la República de Cuba (AGR). Es un sistema desarrollado en la Universidad de las Ciencias Informáticas para gestionar la evaluación del desempeño de los trabajadores de la AGR. Dentro de sus principales funcionalidades están:

- Listar Evaluados del Evaluador Mensualmente:

Permite que cada evaluador pueda ver la lista de sus trabajadores subordinados para seleccionar cual será el que evaluará. Permite filtrar por condición en que se encuentre la evaluación en el momento así como por las categorías evaluativas vigentes establecidas por la Dirección de Recursos Humanos. (DRH)

- Evaluar Trabajadores Mensualmente:

Permite que cada evaluador pueda calificar a cada uno de sus subordinados teniendo en cuenta los criterios evaluativos establecidos, según el desempeño realizado por los mismos y los indicadores generales y específicos establecidos. Permite además, que pueda agregar una nota explicativa del conjunto de criterios emitidos en la evaluación. La evaluación se realizará en el período indicado por la AGR.

- Cálculo del Nivel de Estimulación:

Calcula la evaluación teniendo en cuenta los pesos asignados a cada indicador específico y general de forma ponderada lo cual da como resultado el nivel de estimulación en divisa que cobrará el trabajador.

- Evaluar Automáticamente:

Evalúa automáticamente las evaluaciones incompletas o no realizadas cuando se cumpla el plazo determinado por la DRH de la AGR. Además se realizará el cálculo del nivel de estimulación y se aplicará una sanción sobre el nivel de estímulo para el evaluador incumplidor.

- Ver la Evaluación Mensual:

El trabajador podrá revisar su evaluación y agregar una nota explicativa sobre su criterio de la evaluación realizada a él por su evaluador. El trabajador tendrá un período que indicará la AGR para realizar la revisión de su evaluación.

- Modificar evaluación:

La evaluación estará disponible para su modificación mientras el trabajador no acceda a ver la evaluación. Una vez confirmada solo será posible agregar notas a la evaluación por parte del evaluador y/o del trabajador, las cuales quedaran registradas.

- Evaluar Trabajadores anualmente:

El evaluador realizará las evaluaciones de los trabajadores anualmente lo cual incluye que decisiones a tomar sobre el trabajador, si debe ser ratificado para continuar en su puesto, removido o promovido, además debe valorar si tiene cualidades de dirigente.

- Generar Reportes:

Realizará un conjunto de reportes que brindará información de apoyo para las evaluaciones y para el resto del proceso evaluativo. (Vargas Allegue E. D., 2009)

1.2.2.3. Akademos

Akademos es un sistema automatizado para la gestión académica de la Universidad de las Ciencias Informáticas. Se desarrolló como una aplicación Web haciendo uso de la tecnología Microsoft.NET y los servicios web logrando la creación de sistemas que permitan informatizar las distintas áreas de esta Universidad. Este sistema se hizo con el propósito de llevar un control de las notas de todos los estudiantes de la universidad calculando su índice académico desde el primer año hasta las asignaturas que va cursando durante el curso.

Esta aplicación consta de módulos como:

- Módulo de plan de estudio:

El plan de estudios es el módulo principal del sistema. Este módulo se define como una secuencia de niveles, los cuales los estudiantes van venciendo acorde van finalizando las asignaturas. Las asignaturas a su vez se ubican acorde a un orden de precedencia en los distintos momentos del plan de estudios. Este módulo permite además definir perfiles y niveles para agrupar las asignaturas.

- Módulo de matrícula:

El objetivo fundamental del módulo de matrícula es el control de los datos de los estudiantes así como la gestión de los movimientos a que estos son sometidos en su paso por la universidad. Se define como movimiento cualquier acción que se realiza sobre un estudiante que provoca un cambio de su estado dentro de la universidad, y generan un conjunto de informaciones que debe quedar registrada en su expediente.

- Módulo de expediente:

En un centro de estudios, en su trayectoria por el mismo el estudiante deja un conjunto de informaciones relacionada con su desempeño académico y otros aspectos sobre su comportamiento, lo cual conforma el expediente del estudiante. El conjunto de informaciones que van dejando las generaciones de estudiantes debe ser almacenado con el objetivo de sacar conclusiones históricas que contribuyan al mejoramiento de la actividad del centro.

- Módulo de registro del profesor:

Este módulo es el encargado de permitir el control del progreso de un período académico. Está muy relacionado con el de plan de estudio, ya que, de él toma la información que necesita referente a las evaluaciones y formas de calificación de las asignaturas. Una vez, que el sistema cuente con esta información, es posible que puedan elaborarse una gran cantidad de reportes para determinar el desempeño de un estudiante, un grupo o una escuela completa.

- Módulo de Planificación:

Este módulo es el encargado de facilitar la planificación de un periodo docente. Específicamente da la posibilidad de planificar tanto la carga docente de los profesores como la gestión de los horarios. Este módulo permitirá a los jefes de departamento, decanos docentes de Facultades entre otros, mantener un control de la plantilla de profesores y realizar un balance de la carga docente de estos. Permite la validación de los horarios atendiendo a varios criterios, entre ellos el solapamiento de profesores y locales. También, es un mecanismo eficaz de publicación de los mismos.

- Módulo de reportes:

Generar reportes es quizás la tarea fundamental de todo sistema de gestión de datos. Akademos proporciona este módulo de reportes que cuenta con diferentes herramientas, las cuales permiten el diseño y publicación de nuevos reportes. Con esta funcionalidad el usuario puede diseñar reportes que relacionen varios aspectos de los estudiantes, convirtiéndose en una poderosa herramienta para el estudio de la comunidad estudiantil. (Marín, 2005)

1.3. Herramientas y entornos de desarrollo utilizados

Para el análisis, diseño e implementación de la propuesta de solución que se presenta en el trabajo, se define utilizar algunas herramientas y lenguajes. Se utilizó Apache como servidor de aplicaciones Web, PostgreSQL como Sistema Gestor de Base de Datos y como aplicación gráfica para la base de datos se utilizó pgAdmin3. Para el desarrollo web se utilizó PHP como lenguaje de programación, Visual Paradigm para el modelado de UML, Symfony como Framework para la capa de negocio y el desarrollo de aplicaciones web.

1.3.1. Apache 2

Apache es una tecnología de código abierto, es un servidor altamente configurable de diseño modular. Apache posee integración con lenguajes como Perl, PHP y otros lenguajes de scripting. También trabaja con Java y páginas JSP (Java Servlet Pages). Apache permite la creación de ficheros de logs, de este modo se puede tener un mayor control sobre lo que sucede en el servidor.

Es el servidor más usado en sistemas Linux, se usa para páginas web solicitadas por equipos cliente. Normalmente los clientes solicitan y muestran las páginas mediante el uso de navegadores web. Los servidores web Apache a menudo se usan en combinación con el motor de bases de datos MySQL, el lenguaje de scripting PHP, y otros lenguajes de scripting populares como Python y Perl. (Kair, 2003)

Se escogió apache como servidor web por ser uno de los más robustos actualmente, además de que es uno de los proyectos colaborativos con más fuerza. Apache es completamente software libre y Open Source, siendo una de sus características más notables que trabaja de forma modular, o sea, se le pueden añadir módulos para el trabajo con distintos lenguajes de programación así como conexiones a bases de datos.

1.3.2. PHP 5.2.9

PHP es un lenguaje interpretado de propósito general, ampliamente usado y que está diseñado para desarrollo web y puede ser incrustado dentro de código HTML. Generalmente se ejecuta en un servidor web, tomando el código en PHP como su entrada y creando páginas web como salida. Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas.

PHP también tiene la capacidad de ser ejecutado en la mayoría de los sistemas operativos, tales como UNIX (y de ese tipo, como Linux o Mac OS X) y Windows. Permite la conexión a diferentes tipos de servidores de bases de datos tales como MySQL, Postgres, Oracle, Microsoft SQL Server. (Gallego Vázquez, 2003)

Características:

- Es un lenguaje multiplataforma.
- Completamente orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una Base de Datos.
- El código fuente escrito en PHP es invisible al navegador y al cliente ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador. Esto hace que la programación en PHP sea segura y confiable.
- Capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL.
- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos (llamados extensiones).
- Posee una amplia documentación en su página oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.
- Permite aplicar técnicas de programación orientada a objetos.
- Tiene manejo de excepciones (desde PHP5).

PHP no obliga a quien lo usa a seguir una determinada metodología. El programador puede aplicar en su trabajo cualquier técnica de desarrollo que le permita escribir código ordenado, estructurado y manejable. Un ejemplo de esto son los desarrollos que en PHP se han hecho del patrón de diseño Modelo Vista Controlador (MVC), que permiten separar el tratamiento y acceso a los datos, la lógica de control y la interfaz de usuario en tres componentes independientes.

1.3.3. Eclipse 3.5

Eclipse es un entorno de desarrollo integrado de código abierto multiplataforma. Esta plataforma, típicamente ha sido usada para desarrollar entornos de desarrollo integrados, como el IDE de Java llamado Java Development Toolkit (JDT) y el compilador (ECJ) que se entrega como parte de Eclipse (y que son usados también para desarrollar el mismo Eclipse). El entorno de desarrollo integrado (IDE) de Eclipse emplea módulos (en inglés *plug-in*) para proporcionar toda su funcionalidad al frente de la plataforma de cliente, a diferencia de otros entornos monolíticos donde las funcionalidades están todas incluidas, las necesite el usuario o no. Este mecanismo de módulos es una plataforma ligera para componentes de software, permitiéndole a Eclipse extenderse usando otros lenguajes de programación como son C/C++ y Python. La arquitectura plug-in permite escribir cualquier extensión deseada en el ambiente, como sería Gestión de la configuración. Se provee soporte para Java y CVS en el SDK de Eclipse. Y no tiene por qué ser usado únicamente para soportar otros lenguajes de programación.

En cuanto a las aplicaciones clientes, eclipse provee al programador con Frameworks para el desarrollo de aplicaciones gráficas, definición y manipulación de modelos de software, aplicaciones web.

El Eclipse incluye las herramientas de desarrollo de Java, ofreciendo un IDE con un compilador de Java interno y un modelo completo de los archivos fuente de Java. Esto permite técnicas avanzadas de refactorización y análisis de código.

1.3.4. PostgreSQL 8.4

PostgreSQL es un sistema de administración de base de datos (DBMS por sus siglas en inglés) desarrollado en la Universidad de California. PostgreSQL inició muchos conceptos que solo estuvieron disponibles en bases de datos comerciales mucho más tarde. Este DBMS soporta estándares SQL e incorpora otras funcionalidades como:

- Consultas complejas.
- Llaves foráneas.
- Vistas
- Procedimientos almacenados.
- Integridad transaccional.

PostgreSQL puede ser fácilmente extendido por el usuario en muchas vías, por ejemplo adicionando nuevos tipos de datos, funciones, lenguajes indexados y lenguajes procedurales. (Group, 2010)

Se escogió PostgreSQL por ser uno de los más fuertes objeto-relacionales sistemas de base de datos. Con más de 15 años como proyecto colaborativo ha ganado una fuerte reputación por su integridad de los datos. Puede ser ejecutado en casi todos los grandes sistemas operativos, incluyendo Linux, UNIX y Windows. Tiene gran integración con el estándar SQL 2008, provee además gran cantidad de tipo de datos como INTEGER, NUMERIC, BOOLEAN, CHAR, VARCHAR, DATE, INTERVAL, y TIMESTAMP. Además soporta almacenamiento de grandes objetos binarios, incluyendo imágenes, sonidos, o videos. Tiene interfaces nativas para los lenguajes C/C++, Java, .Net, Perl, Python, Ruby, Tcl, ODBC, y otros.

1.3.5. Visual Paradigm para UML 3.4

Visual Paradigm para UML (VPUML) es una herramienta de diseño que soporta todos los diagramas UML y los diagramas de entidad-relación. VPUML provee muchas facilidades para el modelado de casos de uso incluyendo funciones completas de diagramas de casos de usos utilizando UML, editor de flujo de eventos, y generación de diagrama de actividades. VPUML produce además documentación en PDF, HTML y formatos de Microsoft Word. (Visual Paradigm, 2010)

Características principales:

- Soporte de UML versión 2.1.
- Diagramas de Procesos de Negocio - Proceso, Decisión, Actor de negocio, Documento.
- Modelado colaborativo con CVS y Subversion (control de versiones).
- Interoperabilidad con modelos UML2 (meta modelos UML 2.x para plataforma Eclipse) a través de XMI.
- Ingeniería de ida y vuelta.
- Ingeniería inversa - Código a modelo, código a diagrama.
- Ingeniería inversa Java, C++, Esquemas XML, XML, NET exe/dll, CORBA IDL.
- Generación de código - Modelo a código, diagrama a código.
- Editor de Detalles de Casos de Uso - Entorno todo-en-uno para la especificación de los detalles de los casos de uso, incluyendo la especificación del modelo general y de las descripciones de los casos de uso.
- Diagramas EJB - Visualización de sistemas EJB.

- Generación de código y despliegue de EJB - Generación de beans para el desarrollo y despliegue de aplicaciones.
- Diagramas de flujo de datos.
- Soporte ORM - Generación de objetos Java desde la base de datos.
- Generación de bases de datos - Transformación de diagramas de Entidad-Relación en tablas de base de datos.
- Ingeniería inversa de bases de datos - Desde Sistemas Gestores de Bases de Datos (DBMS) existentes a diagramas de Entidad-Relación.
- Generador de informes.
- Distribución automática de diagramas - Reorganización de las figuras y conectores de los diagramas UML.
- Importación y exportación de ficheros XMI.
- Integración con Visio – Dibujo de diagramas UML con plantillas de Microsoft Visio.
- Editor de figuras.

Se escogió Visual Paradigm porque provee una suite dedicada al desarrollo de software y servicios, donde se puede transformar fácilmente requerimientos del sistema en soluciones de software. Brinda soporte a la última generación de BPMN (Modelado de procesos de negocio), así como para los modelados de bases de datos ERD (diagrama entidad-relación) y ORM (mapeo de objetos relacionales). Visual Paradigm provee además las herramientas para ejecutar ingeniería inversa con los ERD y su base de datos correspondiente , así como generar el código fuente en C++, Java, IDL, PHP , Python ,XML y esquemas XML, base de datos (con JDBC), y archivos .NET .exe y .dll.

1.3.6. pgAdmin3

pgAdmin3 es una interfaz gráfica para bases de datos en PostgreSQL. Esta herramienta es software libre, mantenida por una comunidad internacional de desarrolladores. Esta herramienta está disponible en múltiples plataformas como FreeBSD, Linux y Windows NT/2000/XP/Vista/7.

pgAdmin3 ofrece una variedad de funcionalidades tales como:

- Crear y borrar esquemas, bases de datos, tablas, columnas, etc.
- Ejecución de consultas SQL.
- Exportar los resultados de consultas SQL a ficheros.

- Copias de seguridad de bases de datos o tablas.
- Configurar usuarios, grupos y privilegios.
- Visualizar, editar e insertar datos en las tablas. (Matthew N, 2005)

1.3.7. Framework de desarrollo: Symfony

El Framework de desarrollo Symfony está diseñado para optimizar, el desarrollo de las aplicaciones Web, separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación Web. Este Framework proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación Web compleja. Además automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación. Symfony está desarrollado completamente en PHP. Ha sido probado en numerosos proyectos reales y en sitios Web de comercio electrónico de primer nivel. Compatible con la mayoría de gestores de base de datos como MySQL, PostgreSQL, Oracle y SQL Server de Microsoft. Se adaptó perfectamente a las necesidades reales presentadas para la propuesta de solución, software libre, que cuenta con una gran comunidad en varios idiomas, incluyendo español y que permite agilizar el trabajo.

Se puede ejecutar tanto en plataformas *nix (Unix, Linux) como en plataformas Windows. Symfony está basado en un patrón clásico del diseño Web, conocido como arquitectura Modelo-Vista-Controlador (MVC), toma lo mejor de esta arquitectura y la implementa de forma que el desarrollo de aplicaciones sea rápido y sencillo. (Potencier F, 2008)

1.3.8. Modelo-Vista-Controlador (MVC)

El patrón Modelo-Vista-Controlador (MVC) para diseño web, separa el modelado del dominio, la presentación y las acciones basadas en datos ingresados por el usuario en tres clases diferentes

Modelo: Administra el comportamiento y los datos del dominio de aplicación, responde a requerimientos de información sobre su estado (usualmente formulados desde la vista) y responde a instrucciones de cambiar el estado (habitualmente desde el controlador).

Vista: Maneja la visualización de la información.

Controlador: Controla el flujo entre la vista y el modelo (los datos).

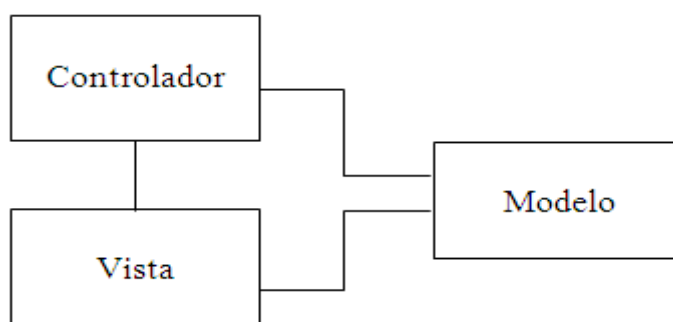


Figura 1 Esquema Modelo-Vista-Controlador

Tanto la vista como el controlador dependen del modelo, el cual no depende de las otras clases. Esta separación permite construir y probar el modelo, independientemente de la representación visual

Entre las ventajas del estilo Modelo-Vista-Controlador están las siguientes:

Soporte de múltiples vistas: Dado que la vista se halla separada del modelo y no hay dependencia directa del modelo con respecto a la vista, la interfaz de usuario puede mostrar múltiples vistas de los mismos datos simultáneamente. Por ejemplo, múltiples páginas de una aplicación Web pueden utilizar el mismo modelo de objetos mostrado de maneras diferentes.

Adaptación al cambio: Los requerimientos de interfaz de usuario tienden a cambiar con mayor rapidez que las reglas de negocios. Los usuarios pueden preferir distintas opciones de representación, o requerir soporte para nuevos dispositivos como teléfonos celulares. Dado que el modelo no depende de las vistas, agregar nuevas opciones de presentación generalmente no afecta al modelo. (T., 2007)

1.4. Conclusión

En este capítulo se da a conocer las herramientas propuestas para el proceso de desarrollo del sistema para la gestión de los pases extras de la Facultad 15, dichas herramientas fueran escogidas de acorde a los principios de usabilidad, potencia y que fueran de código abierto para cumplir con los lineamientos de la Universidad. De acuerdo a los sistemas estudiados anteriormente, se concluye que ninguno de ellos cumple con las necesidades del sistema requerido para el control de pases extras de la Facultad 15. Dichos sistemas realizan la gestión de los recursos humanos, en su mayoría enfocados a negocios específicos y no se adecuan a los requerimientos que se definen para el negocio del control de pases extras de la Facultad 15. Es necesario un sistema diseñado específicamente que cumpla todo el ciclo de vida del proceso de gestión de pases extras para la Facultad 15.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

2.1. Introducción

En el presente capítulo se dará la descripción de todos los procesos que se realizan manualmente para la gestión de pases extras de los estudiantes, también se mostrará de manera detallada los procesos vinculados con el negocio. Estos procesos se podrán visualizar con una mejor claridad en los diagramas que están vinculados en este capítulo.

2.2. Problema y situación problemática

La Universidad de las Ciencias Informáticas la Facultad 15, tiene como objetivo, para que su trabajo mejore, el útil funcionamiento y la documentación de un sistema que controle todos los pases extras de los estudiantes de la Facultad. Los datos que se recogen para gestionar la solicitud de pase no están incluidos en una base de datos, a la que se pueda acceder y que sirva de ayuda a la hora de realizar algún pase para algún estudiante. Se debe tener en cuenta cuando un profesor necesite consultar algún reporte del estudiante que no se encuentra en la universidad, para poder justificarle, algún trabajo convocado por la FEU, la universidad o en su proyecto. El encargado de este proceso tiene que buscar primeramente a todos los estudiantes que han solicitado pase, esta solicitud es hecha manualmente por lo tanto la gestión de la misma viene siendo la situación problemática que se presenta en la Facultad.

2.3. Objeto de informatización

Se desea informatizar el proceso para la gestión y control de los pases extras de los estudiantes de la Facultad 15 de la Universidad de las Ciencias Informáticas.

2.4. Información que se maneja.

La gestión para la reservación de pase contiene implícito todos los datos personales y ubicación del estudiante, uno de los objetivos del sistema es que la manipulación de la misma se haga de una manera segura y confiable, que no atenten contra la autenticidad y seguridad.

2.5. Propuesta del sistema

Para dar solución al problema existente se ha decidido desarrollar un sistema para la gestión y control de los pases extras de los estudiantes, para facilitar la gestión de la búsqueda de los datos de los estudiantes a la hora de justificar algún tipo de actividad referente a la docencia y una planificación de los recursos de la universidad. Por lo que se propone realizar una aplicación web en PHP, en la cual el usuario que autoriza a dar los pases es el que tiene el máximo control de la aplicación.

2.6. Modelo del negocio

Es una labor de los analistas de procesos de negocio, quienes tienen la misión de entender cómo funciona el negocio; modelando el mismo a través de diagramas de actividades donde se refleja la secuencia de pasos que se llevan a cabo, las personas beneficiadas con las acciones realizadas y las que realizan las actividades, que son los que, en ocasiones crean, modifican o acceden a contenedores de información llamados entidades, este flujo de trabajo es uno de los más importantes en el proceso unificado de desarrollo. (UCI, 2009)

2.6.1. Actores de negocio

El término actor significa el rol que algo o alguien juega cuando interactúa con el Negocio para beneficiarse de sus resultados. De acuerdo con esta idea un actor del Negocio representa un tipo particular de usuario del Negocio más que un usuario físico, ya que varios usuarios físicos pueden realizar el mismo papel en relación al negocio, o sea, ser instancias de un mismo actor. (UCI, 2009)

Actores del negocio	Justificación
Usuario	Todo interesado en consultar reporte o solicitar un pase.
Supervisor	Todo el interesado en consultar un reporte del estudiante que esté de pase

Tabla 1 Actores del negocio

2.6.2. Trabajadores del negocio.

Un trabajador del negocio es una abstracción de una persona (o grupo de personas), un ordenador o un sistema de información; que actúa en el negocio realizando una o varias actividades, interactuando con otros trabajadores del negocio y manipulando entidades del negocio (UCI, 2009).

Trabajadores del negocio	Justificación
Supervisor	Es el encargado de autorizar que un estudiante pueda salir de pase

Tabla 2 Trabajadores del negocio

2.7. Especificación de los requisitos de software.

2.7.1. Requerimientos Funcionales

Son capacidades o condiciones que el sistema debe cumplir. En la realización de los casos de uso del negocio, se obtienen las actividades que serán objeto de automatización. Estas actividades no son exactamente los requisitos funcionales, pero sí son el punto de partida para identificar qué debe hacer el sistema. Los requisitos funcionales se mantienen invariables sin importar con que propiedades o cualidades se relacionen.

El sistema permite:

RF1-Autenticarse.

- Solicitar al usuario datos del dominio UCI.
- Validar datos.
- Mostrar al usuario las opciones a las que tiene acceso.
- Notificar con un mensaje.

RF2-Gestionar solicitud de pase.

- Permitir realizar una solicitud.
 - Permitir efectuar la solicitud de pase.
 - Mostrar todas las opciones.
 - Confirmar la solicitud de pase.
- Permitir cancelar solicitud.
 - Mostrar los datos para efectuar la cancelación de pase.

- Efectuar la cancelación.
- Mensaje de confirmación.
- Permitir modificar solicitud
 - Permitir efectuar modificación.
 - Mostrar los campos que puede modificar.
 - Confirmar modificación.

RF3-Revisar solicitud.

- Permitir aceptar la solicitud de pase.
 - Mostrar los estudiantes que realizaron una solicitud de pase.
 - Mostrar datos del estudiante.
 - Permitir un mensaje de confirmación.
- Permitir denegar solicitud de pase.
 - Mostrar los estudiantes que han solicitado un pase.
 - Mostrar los datos del estudiante.
 - Permitir denegar la reservación.
 - Permitir mensaje de confirmación del por qué fue denegada la solicitud.

RF4-Modificar permisos.

- Permite buscar una persona para modificar su permiso para la gestión de solicitud de pase.
- Permite modificar los permisos de un usuario según su rol.

RF5-Consultar reporte.

- Mostrar el reporte que solicite el usuario según su rol. (UC11, 2009)

2.7.2. Requerimientos no funcionales.

Los requisitos no funcionales son propiedades o cualidades que el producto debe tener, son fundamentales en el éxito del producto. Normalmente están vinculados a requisitos funcionales, es decir una vez se conozca lo que el sistema debe hacer podemos determinar cómo ha de comportarse, qué cualidades debe tener o cuán rápido o grande debe ser. Forman una parte significativa de la especificación. Son importantes para que clientes y usuarios puedan valorar las características no

funcionales del producto, pues si se conoce que el mismo cumple con toda la funcionalidad requerida, las propiedades no funcionales. (UCI2, 2009)

a) Apariencia o Interfaz Externa:

El diseño contará con una interfaz sencilla, sin muchas imágenes para no demorar las respuestas al usuario.

b) Usabilidad:

El sistema podrá ser utilizado por personas que tengan un conocimiento básico en el manejo de las computadoras.

c) Soporte:

El sistema es de fácil instalación y configuración.

d) Implementación:

Usar PHP como lenguaje de programación y utilizar PostgreSQL como Sistema Gestor de Bases de Datos.

e) Seguridad:

Garantizar que la información sea vista únicamente por el personal autorizado.

f) Confiabilidad:

El sistema debe estar disponible de forma que el usuario pueda trabajar a cualquier hora del día. Mantenimiento: el sistema debe estar bien documentado de forma que el tiempo de mantenimiento sea mínimo.

g) Restricciones:

En el desarrollo de la aplicación se utiliza Visual Paradigm como lenguaje de modelado para lograr una mejor documentación del sistema. Se emplea como lenguaje de programación PHP y como gestor de bases de datos PostgreSQL.

h) Políticos-culturales

El producto estará basado fundamentalmente en las características peculiares de la universidad, por lo que no será factible su empleo en otros tipos de entidades donde el comportamiento del negocio no es peculiar.

2.8. Diagrama de caso de uso del negocio

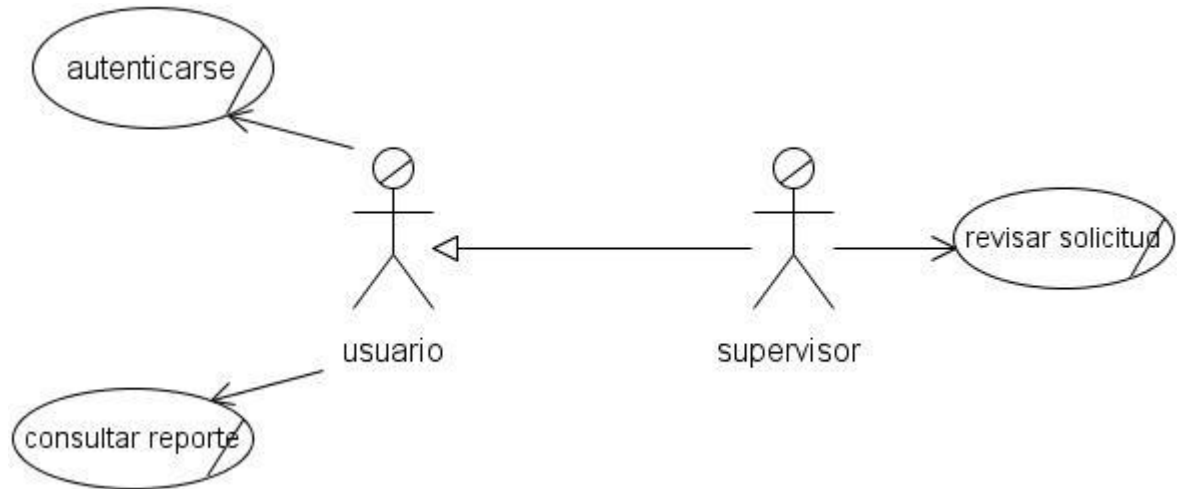


Diagrama 1 Casos de Uso del Negocio

2.9. Descripción de los casos de uso del negocio

Caso de uso Autenticarse.

CU-1	Autenticarse
Actor	Usuario
Trabajador	Supervisor
Propósito	Autenticarse para realizar los procesos previstos de acuerdo a su rol.
Resumen	El caso de uso se inicializa cuando un usuario entra al sistema para registrarse
Curso normal de eventos	
Acción del actor	Respuesta del proceso del negocio
1. El usuario se presenta ante el Sistema de Reservación y solicita ingresar al Sistema.	1.1. El sistema le pide autenticación.
2. El usuario ingresa los datos de autenticación. (usuario y contraseña).	1.2. El Sistema verifica al usuario y permite el acceso.
Curso alternativo de Eventos	
Acciones del Actor	Respuesta del Proceso de Negocio
	2.1. El Sistema comprueba que no es un usuario e informa.

Tabla 3 Caso de uso del negocio: Autenticarse

Caso de Uso Revisar solicitud.

CU-2	Revisar solicitud
Actor	Supervisor
Trabajador	
Propósito	Brindar la posibilidad al supervisor revisar la solicitud de pase teniendo la opción de rechazarla o aceptarla.
Resumen	El Caso de Uso se inicia cuando el supervisor entra al sistema para aceptar o cancelar una solicitud de pase hecha por el estudiante.
Curso Normal de Eventos	
Acciones del Actor	Respuesta del Proceso del Sistema
1. El supervisor entra al sistema.	1.1. El sistema muestra las opciones de aceptar o cancelar la solicitud.
Escenario 1: Aceptar reservación2	
1. El supervisor selecciona la opción para aceptar la reservación.	1.1. El sistema muestra la opción de aceptar
2. El supervisor confirma que desea aceptar la reservación.	1.2. En caso de que el supervisor sea un profesor acepta la solicitud y el sistema adiciona la solicitud 1.3. Luego que el profesor la acepte pasa a ser vista por el vicedecano, este si acepta la solicitud, imprime el boletín de pase y concluye el caso de uso.
Escenario 2: Denegar reservación2.	
1. El supervisor selecciona la opción de denegar la reservación.	1.1. El sistema muestra las reservaciones hechas.
2. El supervisor selecciona la reservación a cancelar.	1.2. El sistema muestra un mensaje de advertencia para la acción a realizar.
3. El supervisor confirma si quiere o no denegar la reservación.	1.3. Si el supervisor acepta cancelar la reservación se concluye el caso de uso.
Curso Alternativo de Eventos	
Acciones del Actor	Respuesta del Proceso del Sistema
	1.1. Si el supervisor cancela la acción, se culmina el Caso de Uso sin ejecutar ninguna acción.

Tabla 4 Caso de uso del negocio: Revisar solicitud

Caso de Uso Consultar reporte

CU-3	Consultar reporte
Actor	Usuario
Trabajador	
Propósito	Brindar la posibilidad tener un reporte sobre la solicitud de pase.
Resumen	El caso de uso se inicia cuando el usuario le solicita al sistema ver el reporte según el rol que desempeña.
Curso Normal de Eventos	
Acciones del Actor	Respuesta del Proceso de Negocio
1. El usuario solicita al sistema de reservación consultar los reportes de acuerdo al rol.	1.1. El sistema de reservación confirma que sea un usuario del sistema. 1.2. El sistema de reservación le da los reportes de acuerdo al rol que ejerce la persona.

Tabla 5 Caso de uso del negocio: Consultar reporte

2.10. Diagrama de actividades

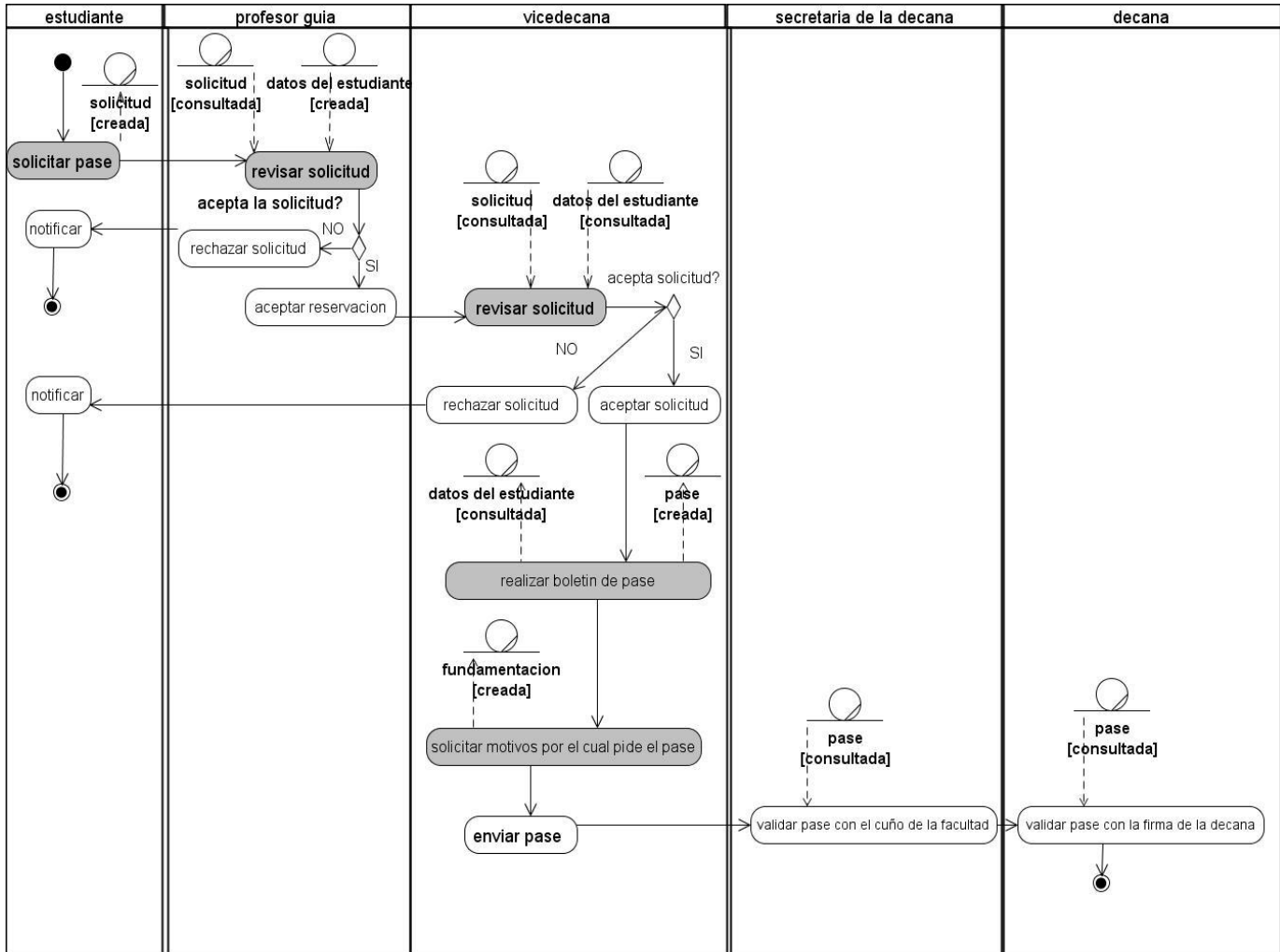


Diagrama 2 Diagrama de Actividades

2.11. Diagrama de clase del modelo de objeto

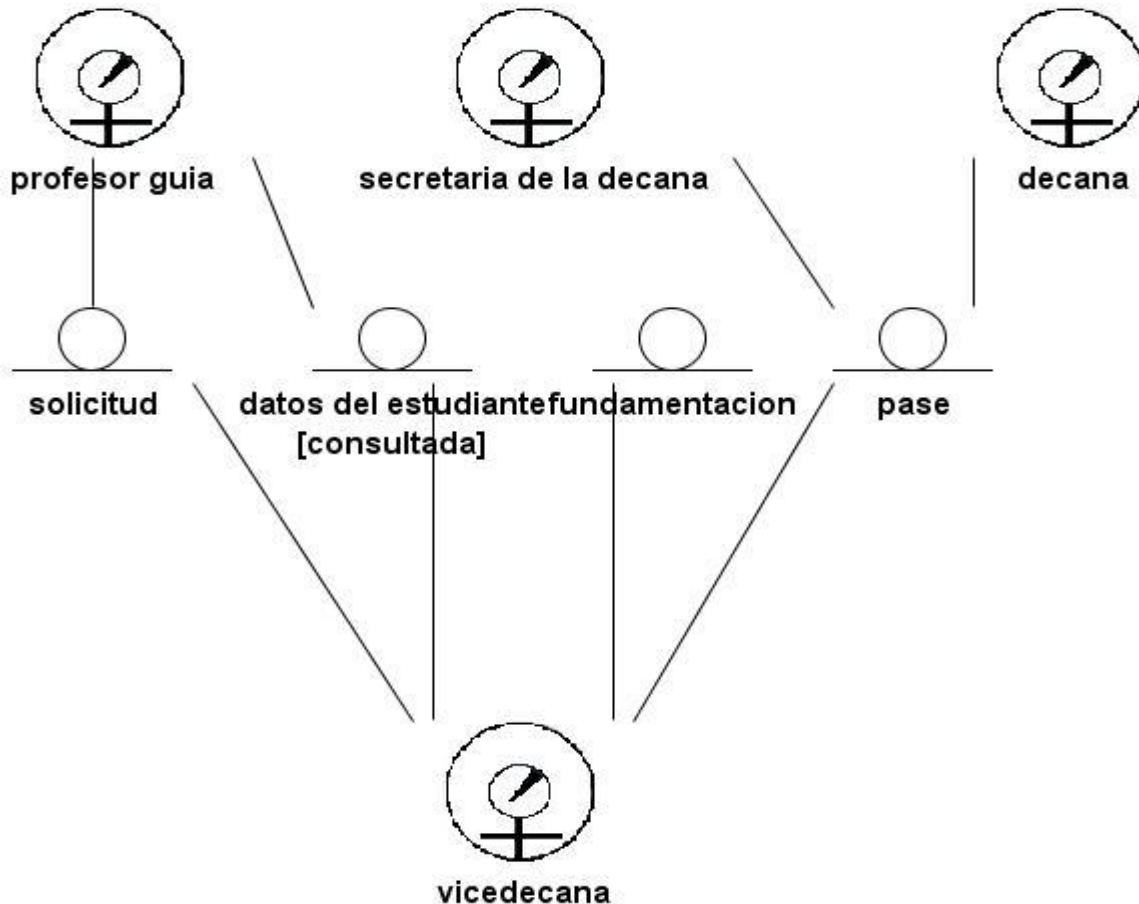


Diagrama 3 Diagrama de clases modelo de objetos

2.12. Definición de los actores del sistema.

Un actor es una idealización de una persona externa, de un proceso, que interactúa con el sistema. Los actores son objetos que residen fuera del sistema, en tanto que los casos de uso son objetos que residen dentro del sistema. Un actor participa en uno o más casos de uso. Un actor puede ser caracterizado por un conjunto de atributos que caracterizan su estado. Los actores pueden ser definidos en jerarquías de generalización. Un actor puede ser una persona, otro sistema informático, o un proceso. (UCI, 2009)

Actores	Justificación
Usuario	Se encarga de autenticarse y consultar los reportes referentes a una solicitud de pase.
Supervisor	Es el encargado de revisar dicha solicitud una vez hecha.
Estudiante	Es el encargado de gestionar la solicitud de pase
Vicedecano	Es el encargado de modificar el permiso de la persona según su rol.

Tabla 6 Actores del sistema

2.13. Definición de los casos de uso del sistema

Conjunto de secuencia de acciones que un sistema ejecuta y que produce un resultado observable para un actor. Un caso de uso es una unidad coherente de funcionalidad, externamente visible, proporcionada por una unidad del sistema y expresada por secuencias de mensajes intercambiados por la unidad del sistema y uno o más actores. (UCI2, 2009)

CU-1	Autenticarse
Actor	Usuario
Trabajador	
Propósito	Autenticación del usuario para ingresar al Sistema
Resumen	El caso de uso se inicia cuando el usuario ingresa a la página de inicio para ingresar al Sistema de Reservación.
Referencia	RF-1
Curso normal de eventos	
Acciones del Actor	Respuesta del Proceso del Sistema
1. El usuario se presenta ante el Sistema de Reservaciones y solicita ingresar.	1.1. El sistema pide autenticación.
2. El usuario ingresa los datos de autenticación. (usuario y contraseña)	1.2. El sistema verifica al usuario y permite el acceso.
Curso alternativo de eventos	
Acciones del Actor	Respuesta del Proceso del Sistema
	1.3. El sistema comprueba que no es un usuario e informa.

Tabla 7 Caso de uso del sistema: Autenticarse

CU-2	Gestionar solicitud de pase.
Actor	Estudiante
Trabajador	
Propósito	Gestionar una solicitud de pase
Resumen	El caso de uso se inicia cuando el estudiante solicita realizar una reservación de pase así como cancelarla o modificarla.
Referencia	RF-2
Curso normal de eventos	
Acciones del actor	Respuesta del Proceso del Sistema
1. El estudiante selecciona la opción gestionar reservación.	1.1. El sistema muestra la opción de realizar una reservación, cancelarla o modificarla
Escenario 1: Permitir realizar una solicitud.	
1. El estudiante selecciona la opción para realizar una solicitud de pase.	1.1. El sistema muestra los campos para poder realizar una solicitud de pase.
2. El estudiante introduce los datos que hacen falta para realizar una solicitud de pase.	1.2. El sistema guarda los datos de la solicitud y concluye el caso de uso.
Escenario 2: Permitir cancelar solicitud	
1. El estudiante selecciona la opción para cancelar la solicitud.	1.1. El sistema muestra la opción de buscar la solicitud hecha por el estudiante.
2. El estudiante selecciona la solicitud de pase que va a cancelar	1.2. El sistema muestra el mensaje de cancelar la solicitud.
3. El estudiante confirma si quiere cancelarla o no.	1.3. Si el estudiante acepta el sistema cancela y finaliza el caso de uso.
Escenario 3: Permitir modificar solicitud	
1. El estudiante selecciona la opción de modificar la solicitud de pase.	1.1. El sistema le muestra los campos que pueden modificar.
2. El estudiante selecciona el campo a modificar.	1.2. El sistema muestra un mensaje notificando si desea modificar la fecha de entrada, la fecha de salida y la fundamentación de la solicitud de pase.
3. El estudiante confirma que desea modificarla.	1.3. El sistema modifica y guarda la modificación y concluye el caso de uso.
Curso alternativo de Eventos	
Acciones del Actor	Respuesta del Proceso del Sistema
	1.4. Si el estudiante cancela la reservación se culmina el caso de uso sin ejecutar ninguna otra acción.

Tabla 8 Caso de uso del sistema: Gestionar solicitud de pase

CU-3	Revisar solicitud
Actor	Supervisor
Trabajador	
Propósito	Brindar la posibilidad al supervisor revisar la solicitud de pase teniendo la opción de rechazarla o aceptarla.
Resumen	El Caso de Uso se inicia cuando el supervisor entra al sistema para aceptar o cancelar una solicitud de pase hecha por el estudiante.
Referencia	RF-3
Curso Normal de Eventos	
Acciones del Actor	Respuesta del Proceso del Sistema
2. El supervisor entra al sistema.	1.2. El sistema muestra las opciones de aceptar o cancelar la solicitud.
Escenario 1: Aceptar reservación2	
3. El supervisor selecciona la opción para aceptar la reservación.	1.4. El sistema muestra la opción de aceptar
4. El supervisor confirma que desea aceptar la reservación.	1.5. En caso de que el supervisor sea un profesor acepta la solicitud y el sistema adiciona la solicitud 1.6. Luego que el profesor la acepte pasa a ser vista por el vicedecano, este si acepta la solicitud, imprime el boletín de pase y concluye el caso de uso.
Escenario 2: Denegar reservación2.	
4. El supervisor selecciona la opción de denegar la reservación.	1.4. El sistema muestra las reservaciones hechas.
5. El supervisor selecciona la reservación a cancelar.	1.5. El sistema muestra un mensaje de advertencia para la acción a realizar.
6. El supervisor confirma si quiere o no denegar la reservación.	1.6. Si el supervisor acepta cancelar la reservación se concluye el caso de uso.
Curso Alternativo de Eventos	
Acciones del Actor	Respuesta del Proceso del Sistema
	1.2. Si el supervisor cancela la acción, se culmina el caso de uso sin ejecutar ninguna acción.

Tabla 9 Caso de uso del sistema: Revisar solicitud

CU-4	Modificar permisos
Actor	Vicedecano
Trabajador	
Propósito	Brindar la posibilidad de modificar los permisos a determinado rol, para que este pueda, gestionar solicitud de pase
Resumen	El caso de uso se inicia cuando el vicedecano solicite al sistema la información sobre determinado persona.
Referencia	RF4
Curso Normal de Eventos	
Acciones del Actor	Respuesta del Proceso del Sistema
1. El vicedecano entra al sistema	1.1. El sistema muestra la opción de modificar permisos
2. El vicedecano busca a la persona según criterios establecidos.	1.2. El sistema muestra la persona que coincide con los criterios de búsqueda 1.3. El sistema muestra las opciones de modificar los atributos de la persona encontrada en la búsqueda.
3. El vicedecano modifica el campo permisos de la persona	1.4. El sistema guarda las operaciones realizadas sobre la persona y se finaliza el caso de uso

Tabla 10 Caso de uso del sistema: Modificar permisos

CU-5	Consultar reporte
Actor	Usuario
Trabajador	
Propósito	Brindar la posibilidad tener un reporte sobre la solicitud de pase.
Resumen	El caso de uso se inicia cuando el usuario le solicita al sistema ver el reporte según el rol que desempeña.
Curso Normal de Eventos	
Acciones del Actor	Respuesta del Proceso de Negocio
1. El usuario solicita al sistema de reservación consultar los reportes sobre la solicitud de pase, teniendo en cuenta el rol que desempeña.	1.1. El sistema de reservación confirma que sea un usuario del sistema. 1.2. El sistema de reservación le da los reportes de acuerdo al rol que ejerce la persona. 1.3. El sistema muestra los reportes, el usuario revisa y finaliza el caso de uso.

Tabla 11 Caso de uso del sistema Consultar reportes

2.14. Diagrama del caso de uso del sistema

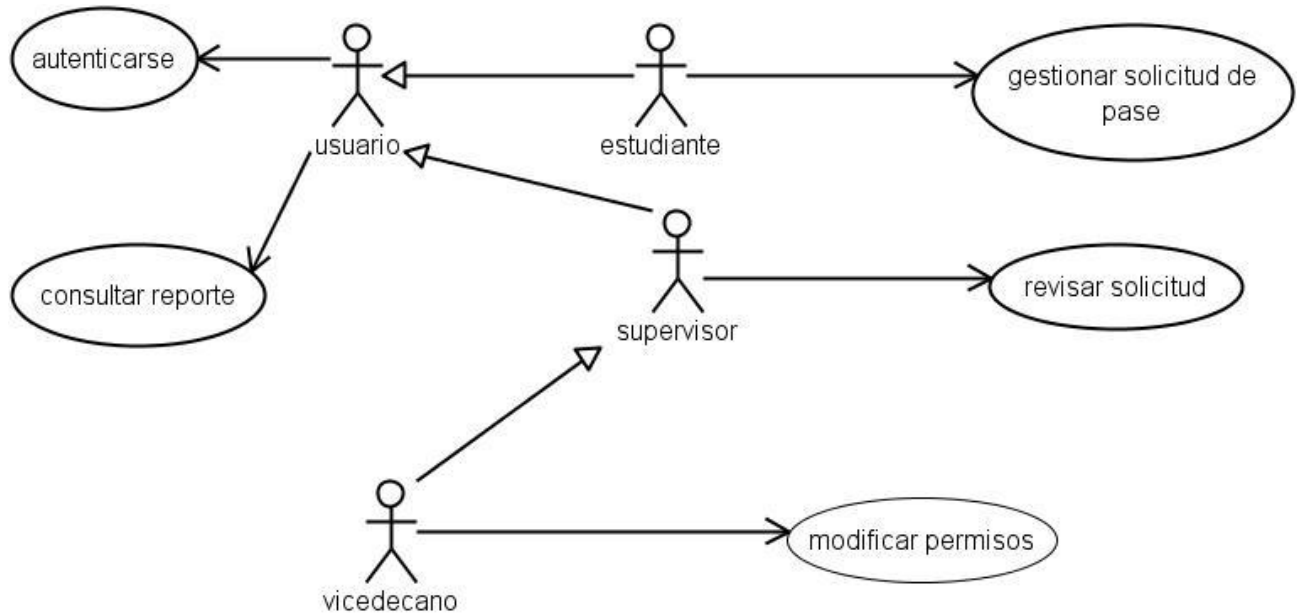


Diagrama 4 Diagrama de caso de uso del sistema

2.15. Conclusiones

En este capítulo se ha desarrollado la propuesta de la solución obteniéndose un modelado del diagrama de CUS, la descripción detallada de cada uno de los mismos en un lenguaje sencillo, se realizó una descripción referente al negocio y se especificaron los requisitos, tratando de que se cumplan todos los requerimientos y las funciones que han sido consideradas necesarias en este capítulo.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

3.1. Introducción

En el presente capítulo se exponen los diagramas de clases tanto de análisis como de diseño que muestran las relaciones entre las clases involucradas en cada caso de uso.

3.2. Modelo del análisis

3.2.1. Definición del modelo de análisis. Modelo de clases de análisis

Un modelo de análisis estructura los requisitos de un modo que facilita su comprensión, su preparación, su modificación y en general, su mantenimiento. Puede considerarse como una primera aproximación al modelo de diseño.

Las clases del análisis siempre encajan en uno de tres estereotipos básicos: de interfaz; de control o de entidad. Las clases de interfaz se utilizan para modelar la interacción entre el sistema y sus actores, las clases entidad se utilizan para modelar información que posee una vida larga y que es a menudo persistente y las clases de control representan coordinación, secuencia, transacciones y control de otros objetos. (UCI3, 2009)

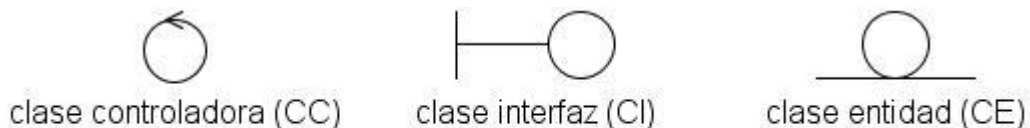


Figura 2 Clases del análisis

3.3. Diagramas de clases de análisis por cada caso de uso.

Un Diagrama de clases del análisis es un artefacto en el que se representan los conceptos en un dominio del problema.

• **CU Autenticarse**

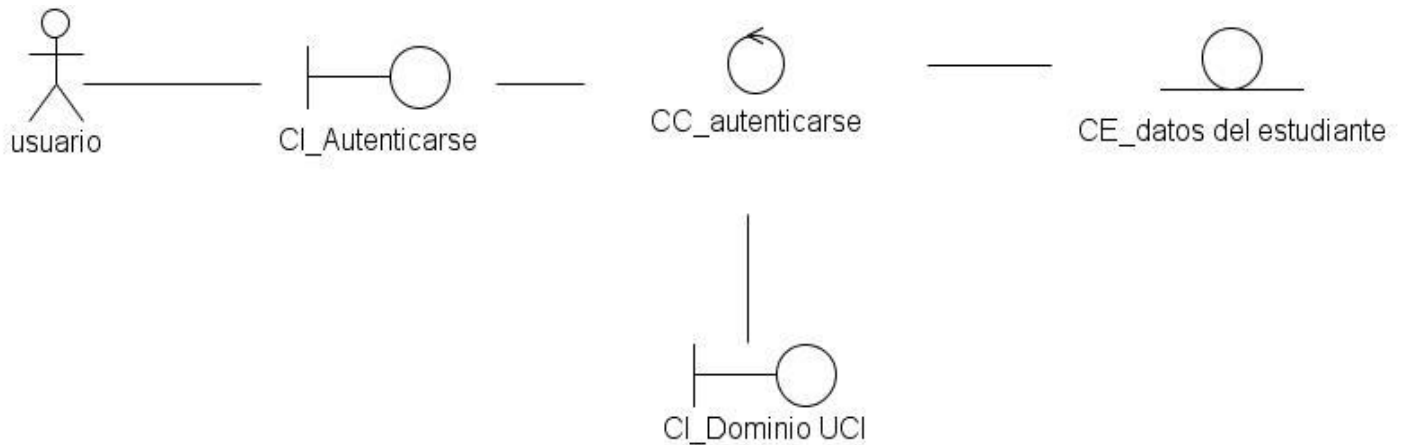


Diagrama 5 Diagrama de clases del análisis: Caso de uso Autenticarse

• **CU Gestionar solicitud de pase.**

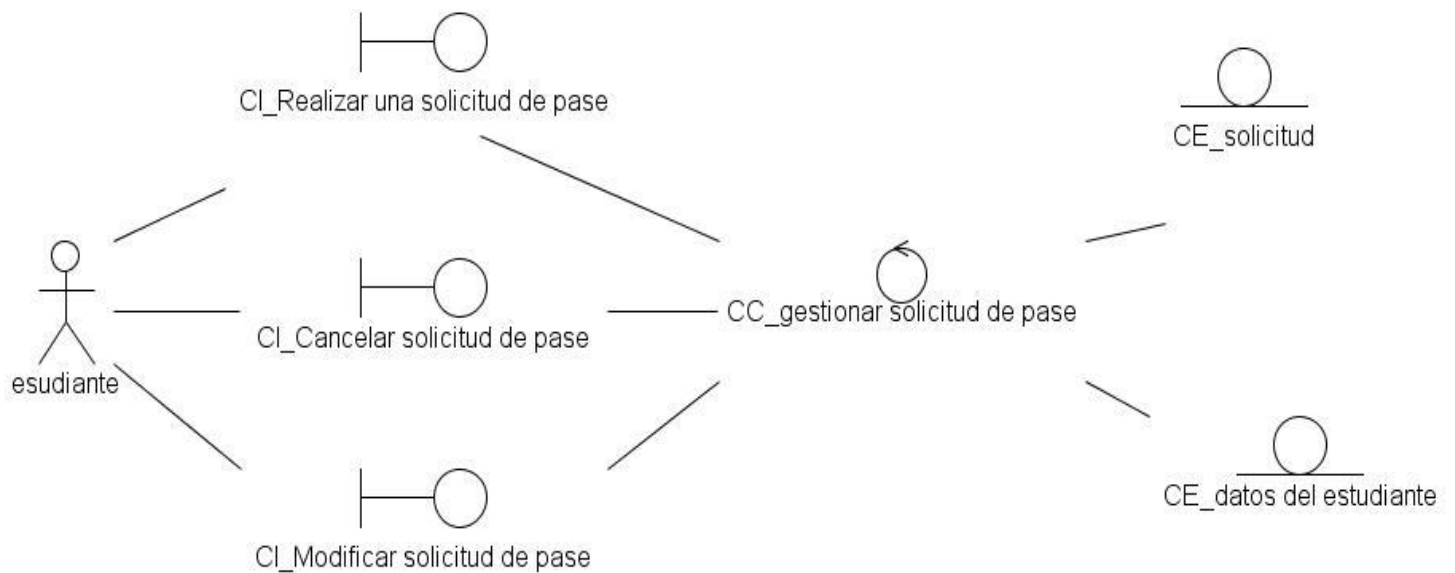


Diagrama 6 Diagrama de clases del análisis: Caso de uso Gestionar solicitud de pase

• **CU Revisar solicitud**

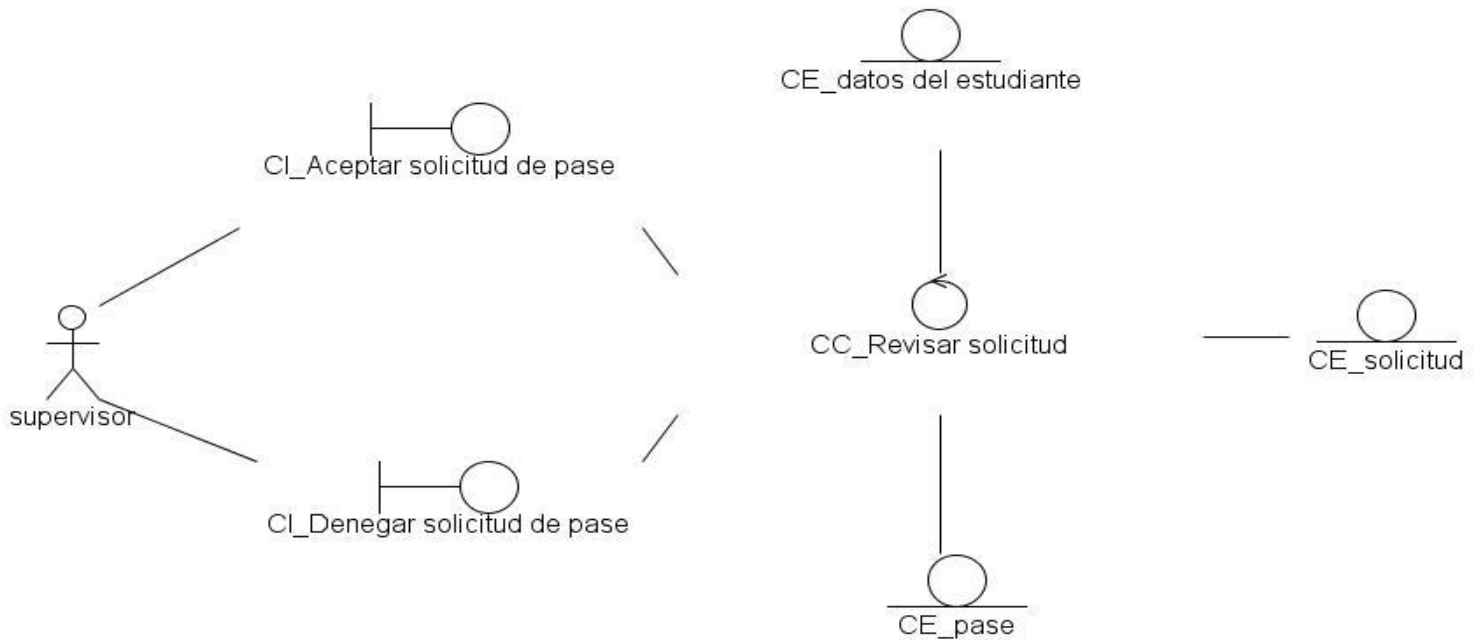


Diagrama 7 Diagrama de clases del análisis: Caso de uso revisar solicitud.

• **CU Modificar permisos**

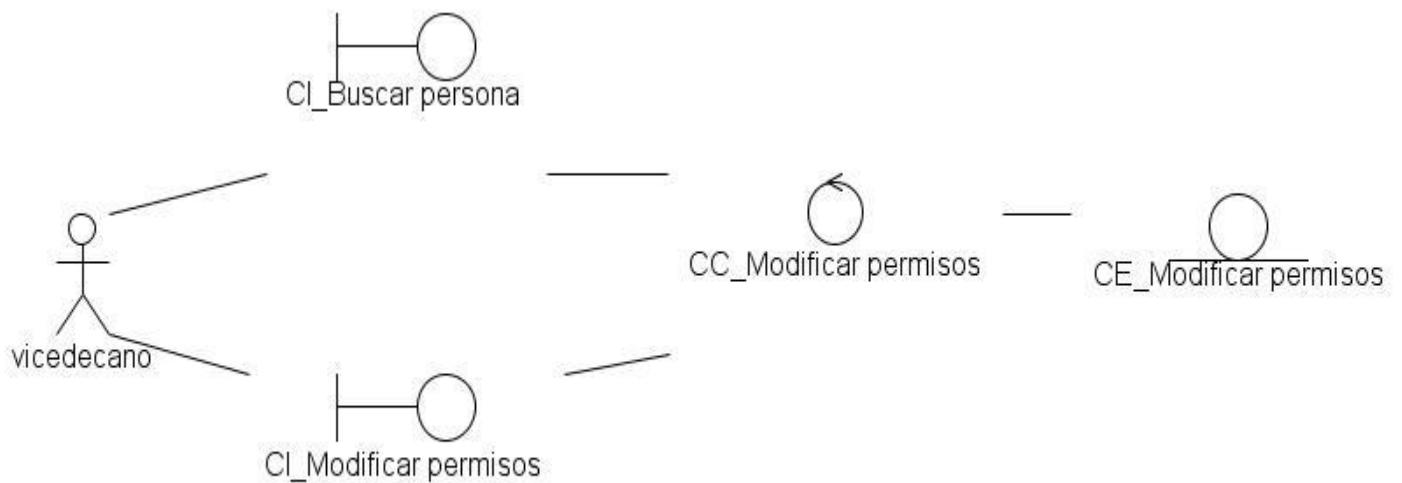


Diagrama 8 Diagrama de clases del análisis: Caso de uso Modificar permisos.

- **CU Consultar reporte.**

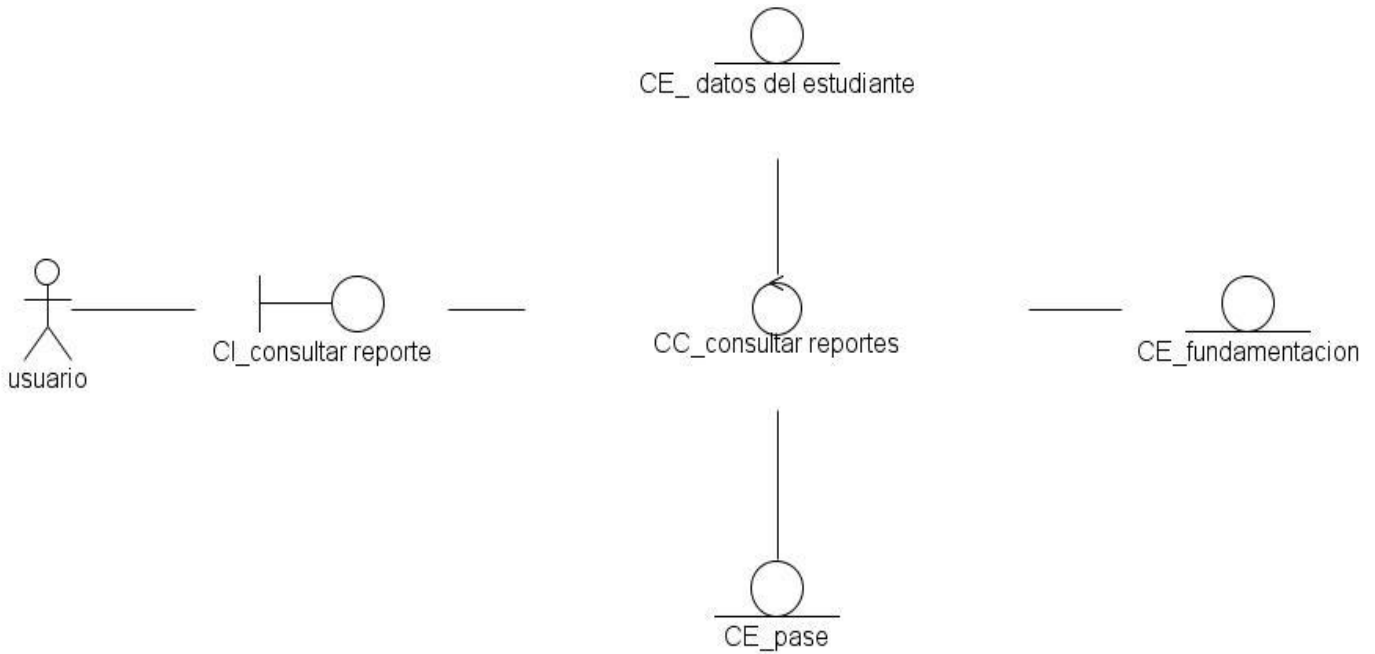


Diagrama 9 Diagrama de clases del análisis: Caso de uso Consultar reportes.

3.4. Diseño

Mediante el diseño se realiza un refinamiento del análisis, teniendo en cuenta los requisitos no funcionales, que no es más que ver cómo cumple el sistema sus objetivos y considerando además el entorno de implementación.

3.5. Diagrama de clases del Diseño

El diagrama de clases para aplicaciones web son más importantes la modelación de la lógica y estado del negocio que los detalles de presentación.

Para obtener un nivel correcto de abstracción y detalle que permita tener un resultado final, es mejor modelar las páginas, los enlaces entre estas, así como los contenidos dinámicos, una vez que estén en el

navegador del cliente, estos son los artefactos que se necesita modelar para la implementación del producto final.

- **Diagrama de Clases del Diseño: CU Autenticar.**

Describe todo el proceso de autenticación que se lleva a cabo para acceder a la aplicación por parte del usuario.

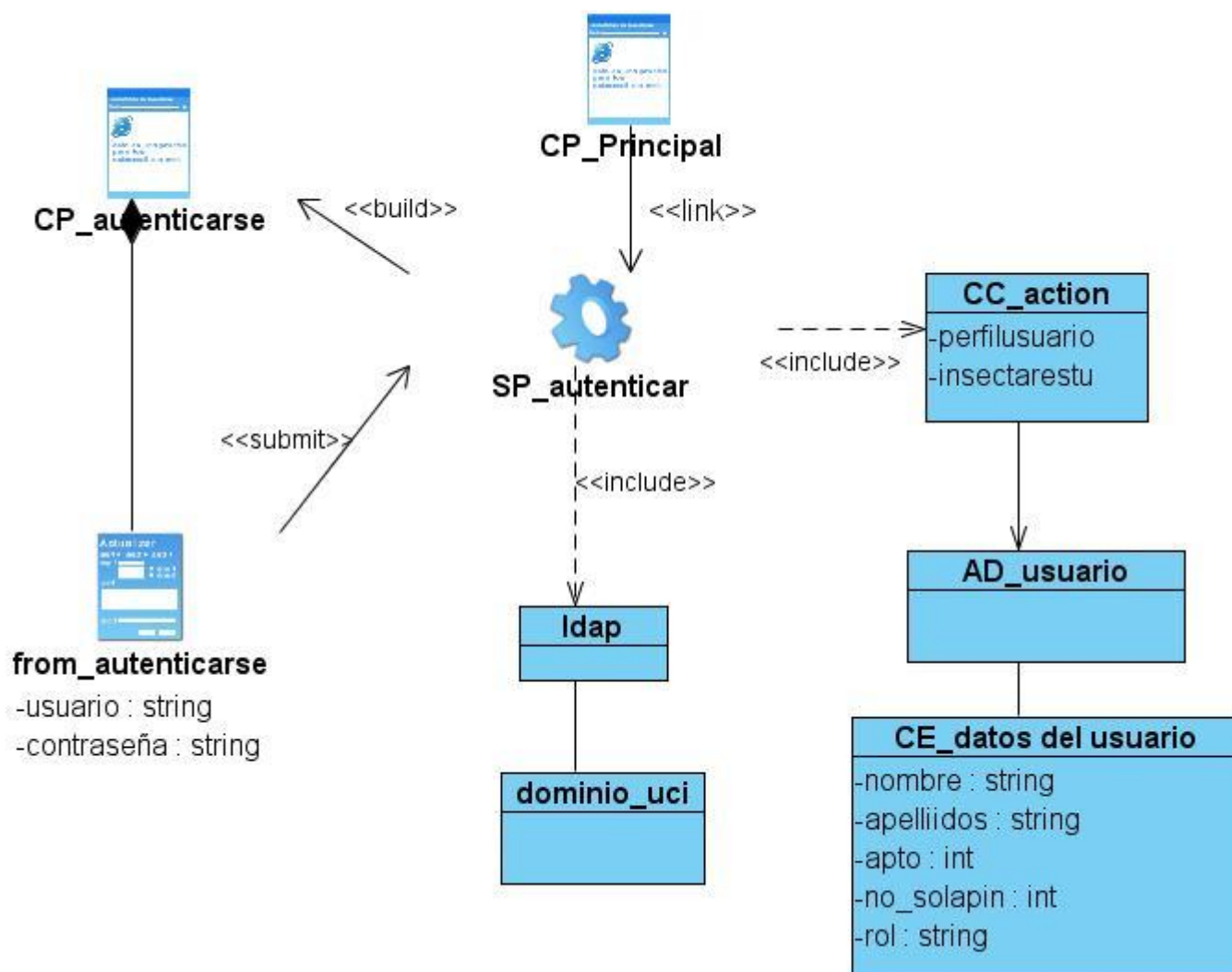


Diagrama 10 Diagrama de Clases del Diseño: Caso de Uso Autenticar.

- Diagrama de Clases del Diseño: CU Gestionar solicitud de pase.

Describe los procesos de realizar, cancelar o modificar una solicitud de pase por parte de un estudiante.

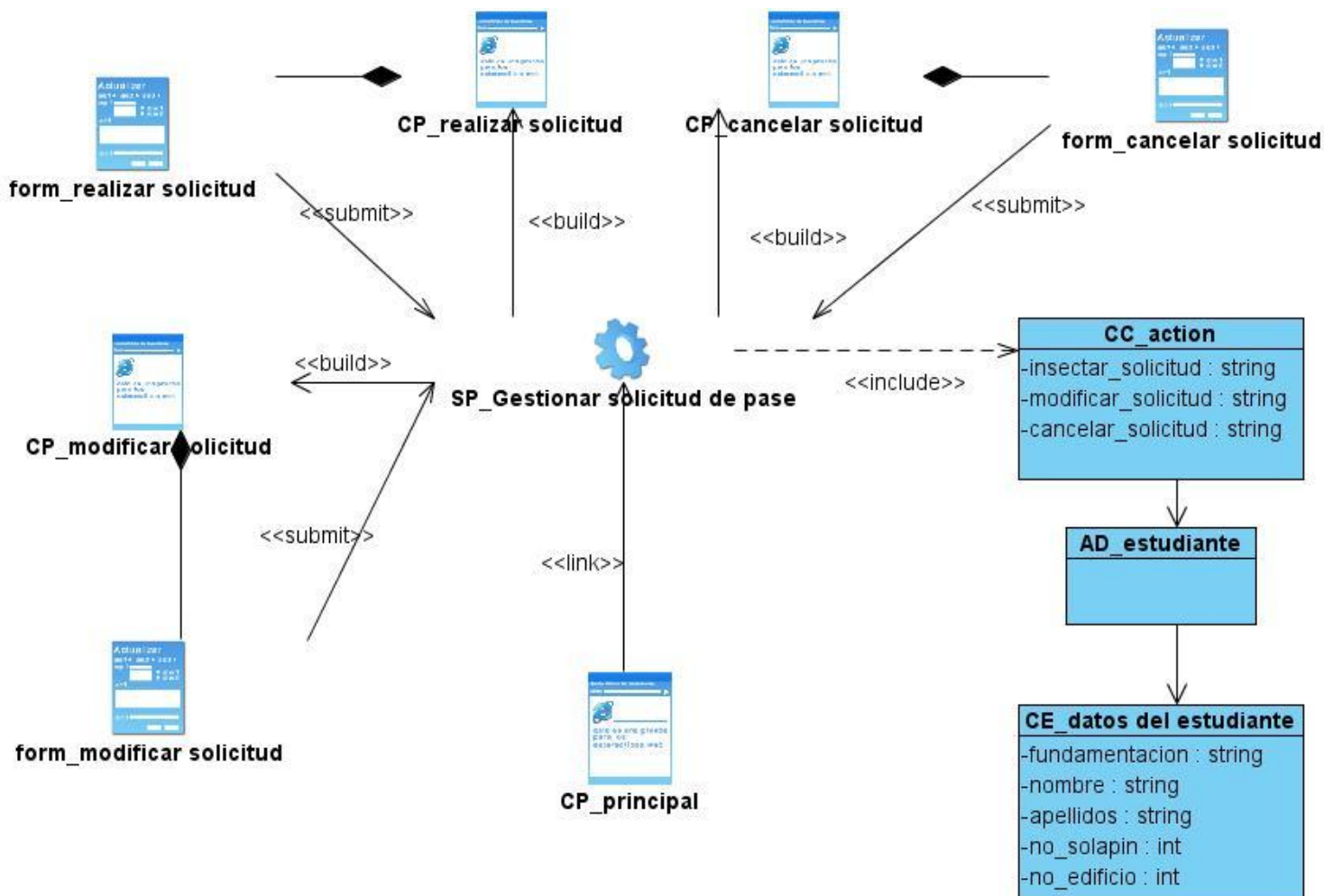


Diagrama 11 Diagrama de Clases del Diseño: Caso de Uso Gestionar solicitud de pase.

- **Diagrama de Clases del Diseño: CU Revisar solicitud**

Describe el proceso de gestión de la solicitud de pase por parte del supervisor.

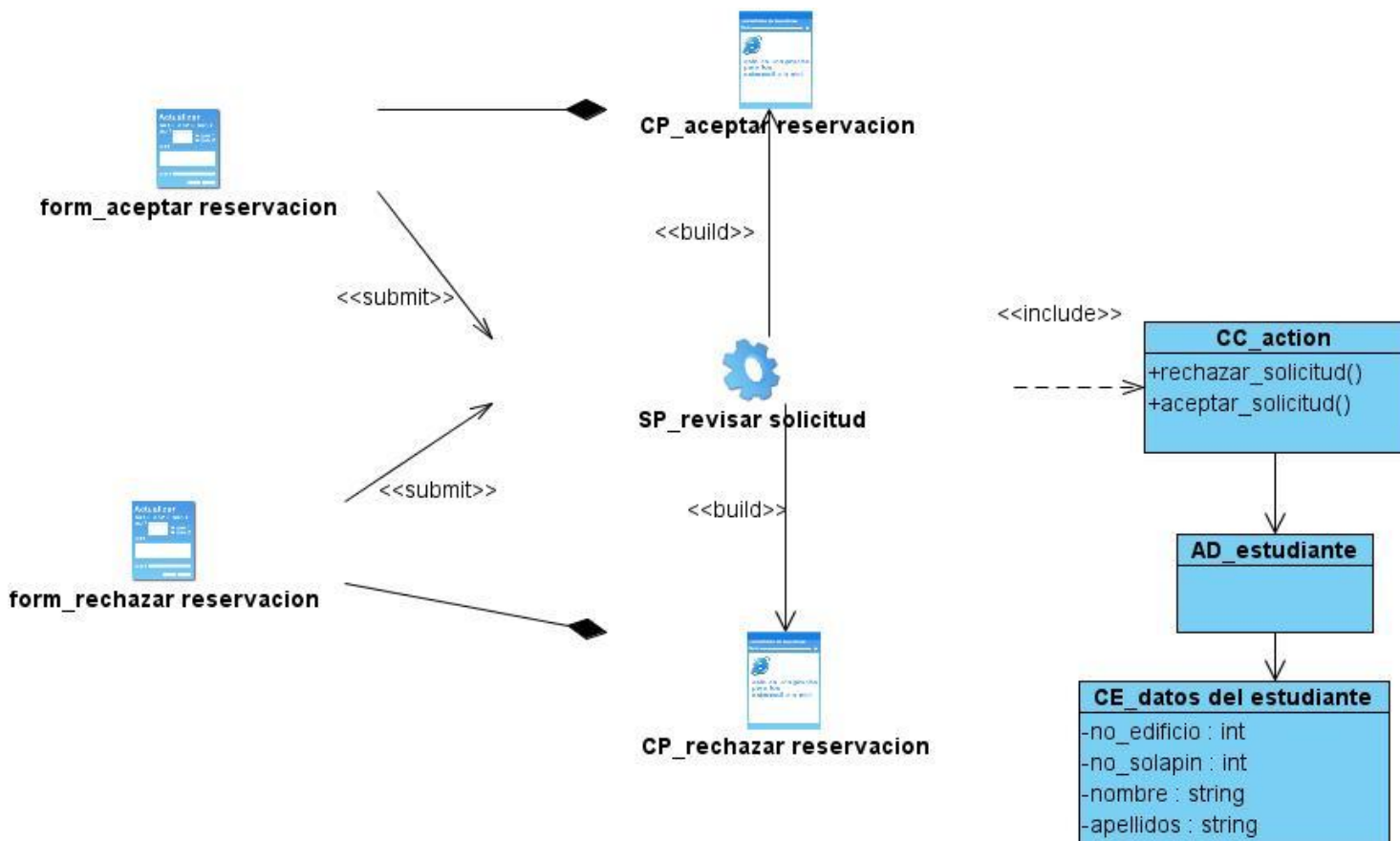


Diagrama 12 Diagrama de Clases del Diseño: Caso de Uso Revisar solicitud

- **Diagrama de Clases del Diseño: CU Modificar permisos**

Describe el proceso donde el supervisor busca a la persona para poder modificarle los permisos para gestionar una solicitud.

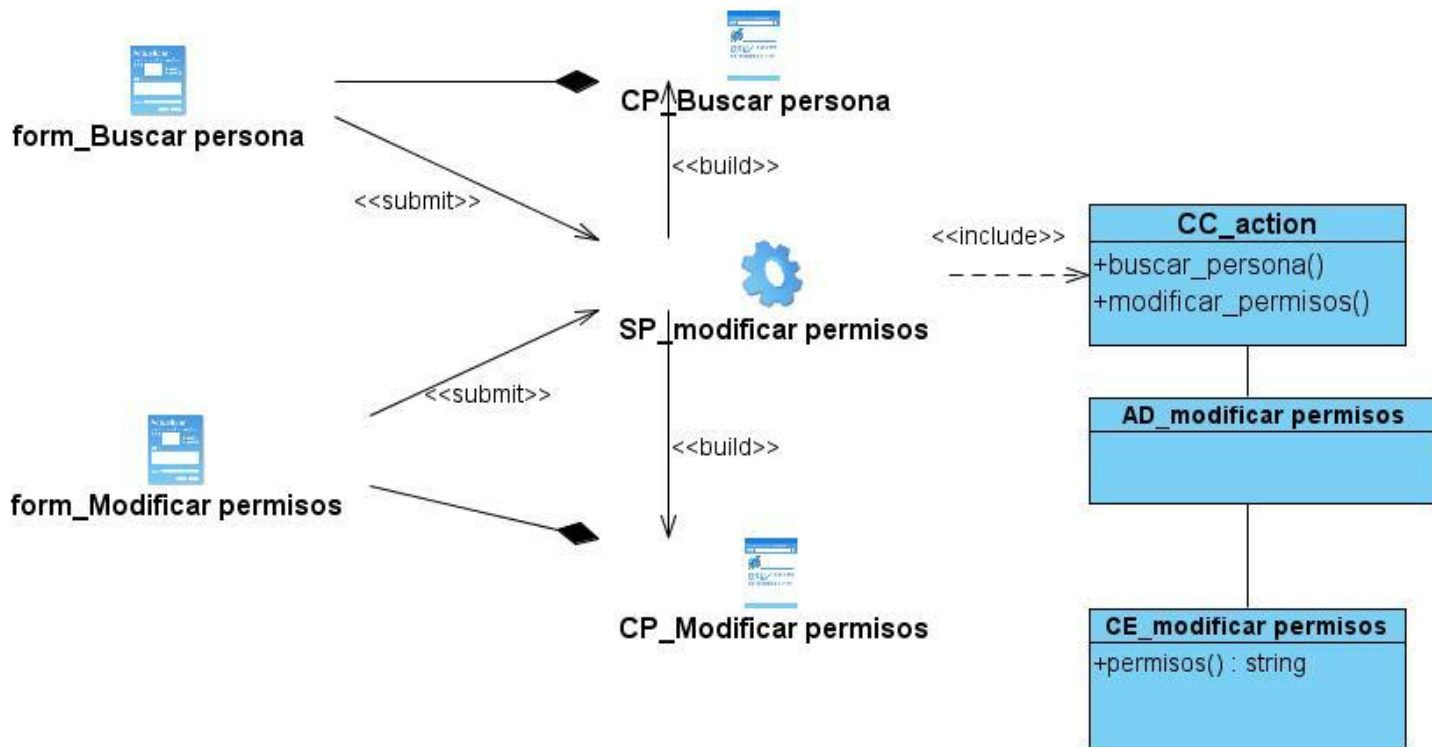


Diagrama 13 Diagrama de Clases del Diseño: Caso de Uso Modificar permisos

- **Diagrama de Clases del Diseño: CU Consultar reporte**

Describe el proceso donde el usuario pueda consultar los reportes previos de una solicitud de pase.

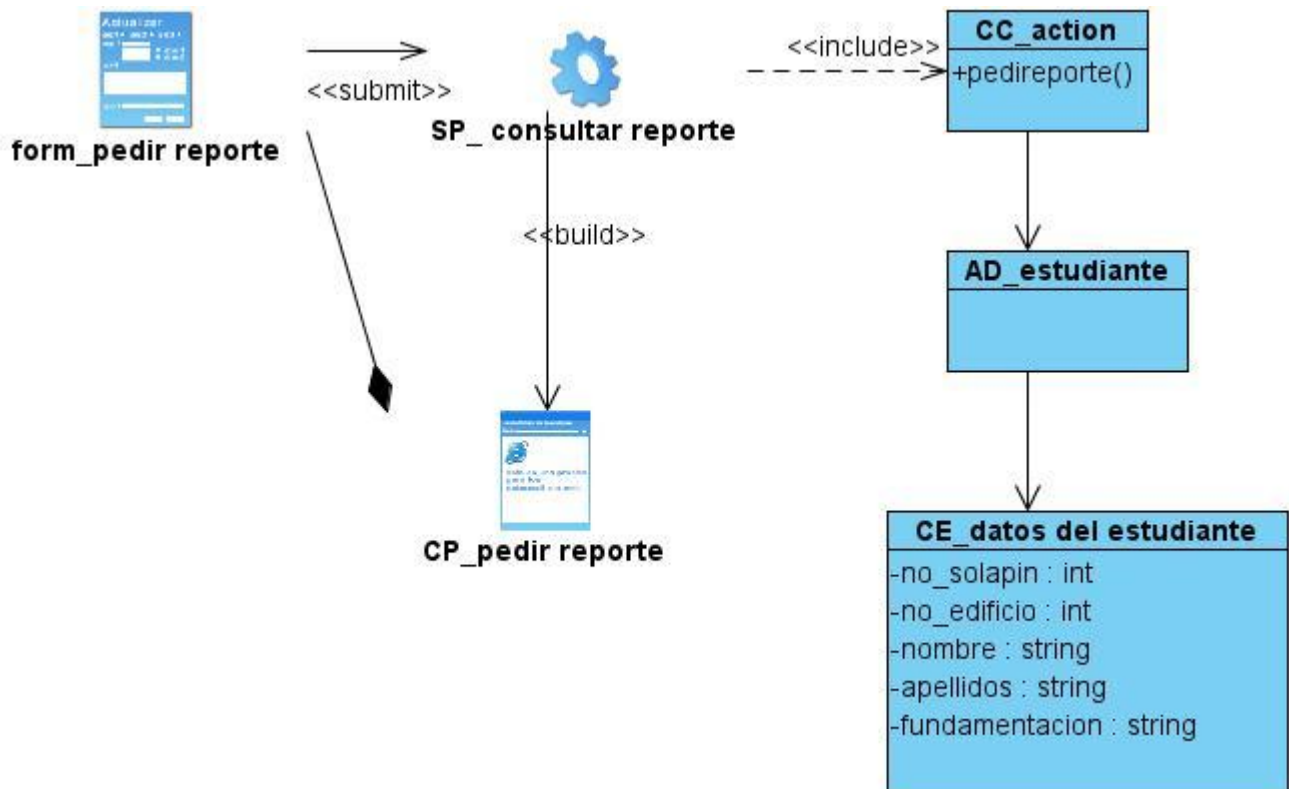


Diagrama 14 Diagrama de Clases del Diseño: Caso de Uso Consultar reporte

3.5.1. Definiciones del Diseño que se aplican al Sistema de Reservación de Pases orientado a la arquitectura de Symfony.

La arquitectura moldea la forma que debe tener el sistema, además los Casos de Uso deben de centralizarse en la arquitectura, así como debe permitir el desarrollo de nuevos casos de uso.

Se creó una aplicación llamada “pases”, dentro de esta se crearon tres módulos inicio, administrar y reservar. Cada módulo consta de tres carpetas, estas son: actions, config y templates. Dentro de “actions” se encuentra el fichero `action.class.php`, donde se definen las acciones del módulo. Dentro de la aplicación “pases” también se presentan cinco carpetas y dentro de “config” se encuentra el

databases.yml, este archivo es utilizado para la conexión a la base de datos. La carpeta templates tiene el archivo layout.php que contiene el diseño visual de todo el subsistema.

En la dirección “pases/lib/model” se encuentran las clases definidas para el modelo de la aplicación.

En la carpeta web, se encuentran tres carpetas “css”, “images” y “js”, además del fichero “pases_dev.php” que representa el controlador frontal de la aplicación. La carpeta “css” contiene las hojas de estilo propias de la aplicación “pases”, dentro de “images” también se encuentran las imágenes, lo mismo para “js” pero con intención de contener los ficheros java script.

El framework Symfony cumple con una gran gama de patrones de diseño que lo hacen muy robusto tanto en su código fuente como en las aplicaciones que se implementan con él.

3.5.2. Patrones GRASP (General Responsibility Assignment Software Patterns) implementados:

Front-Controller: Todas las peticiones deben ser manejadas por un único controlador frontal, que es el punto de entrada a la aplicación. Este es el caso de front-controller definido por Symfony para cada entorno. Quien se encarga de recibir y delegar al controlador del módulo todas las peticiones así como de mostrar las respuestas de dichas peticiones.

Bajo Acoplamiento: La clase Actions hereda solamente de sfActions para lograr un bajo acoplamiento de clases. (Potencier F, 2008)

3.5.3. Patrones GOF implementados:

Decorator: Este patrón añade funcionalidad a una clase dinámicamente. El archivo layout.php, que también se denomina plantilla global almacena el código HTML que es común a todas las páginas de la aplicación, para no tener que repetirlo en cada página. El contenido de la plantilla se integra en el layout, o si se mira desde el otro punto de vista, el layout decora la plantilla. (Potencier F, 2008)

3.6. Diseño de la Base de Datos

Una base de datos es un conjunto de datos lógicamente coherente, con un significado inherente. Se diseña, se construye y se prueba con datos para propósitos específicos. Está dirigida a un grupo de

usuarios y tiene ciertas aplicaciones preconcebidas. Posee una fuente de la cual se derivan los datos, cierto grado de interacción con los hechos del mundo real, un público activamente interesado en su contenido; su tamaño es variado y debe de ser posible buscar, obtener y actualizar los datos siempre que sea necesario.

3.6.1. Diagrama Entidad Relación de la BD

El modelo Entidad Relación está basado en una percepción del mundo real que consta de un conjunto de objetos básicos llamados *entidades* con sus *atributos* y de las *interrelaciones* que existen entre estos objetos. Se desarrolló para facilitar el diseño de Bases de Datos permitiendo la especificación de un esquema del universo de discurso que representa la estructura completa de una Base de Datos.

Normalización de la Base de datos

Regla	Descripción
Primera forma normal	Incluye la eliminación de todos los grupos repetidos
Segunda forma normal	Asegura que todas las columnas que no son llave sean completamente dependientes de la llave primaria (PK).
Tercera forma normal	Elimina cualquier dependencia transitiva. Una dependencia transitiva es aquella en la cual las columnas que no son llave son dependientes de otras columnas que tampoco son llave.

Tabla 12 Reglas de normalización

Primera Forma Normal

La regla de la Primera Forma Normal establece que las columnas repetidas deben eliminarse y colocarse en tablas separadas.

Poner la base de datos en la Primera Forma Normal resuelve el problema de los encabezados de columna múltiples. Muy a menudo, los diseñadores de bases de datos inexpertos harán algo similar a la tabla no normalizada. Una y otra vez, crearán columnas que representen los mismos datos. La normalización ayuda a clarificar la base de datos y a organizarla en partes más pequeñas y más fáciles de entender. En

lugar de tener que entender una tabla gigantesca y monolítica que tiene diferentes aspectos, sólo tenemos que analizar los objetos pequeños y más tangibles, así como las relaciones que guardan con otros objetos.

Segunda Forma Normal

La regla de la Segunda Forma Normal establece que todas las dependencias parciales se deben eliminar y separar dentro de sus propias tablas. Una dependencia parcial es un término que describe a aquellos datos que no dependen de la llave primaria de la tabla para identificarlos.

Una vez alcanzado el nivel de la Segunda Forma Normal, se controlan la mayoría de los problemas de lógica. Podemos insertar un registro sin un exceso de datos en la mayoría de las tablas.

Tercera Forma Normal

Una tabla está normalizada en esta forma si todas las columnas que no son llave son funcionalmente dependientes por completo de la llave primaria y no hay dependencias transitivas. Como se explica anteriormente que una dependencia transitiva es aquella en la cual existen columnas que no son llave que dependen de otras columnas que tampoco son llave.

Cuando las tablas están en la Tercera Forma Normal se previenen errores de lógica cuando se insertan o borran registros. Cada columna en una tabla está identificada de manera única por la llave primaria, y no debe haber datos repetidos. Esto provee un esquema limpio y elegante, que es fácil de trabajar y expandir. (Mato Garcia, 1999)

3.7. Diagrama entidad relación.

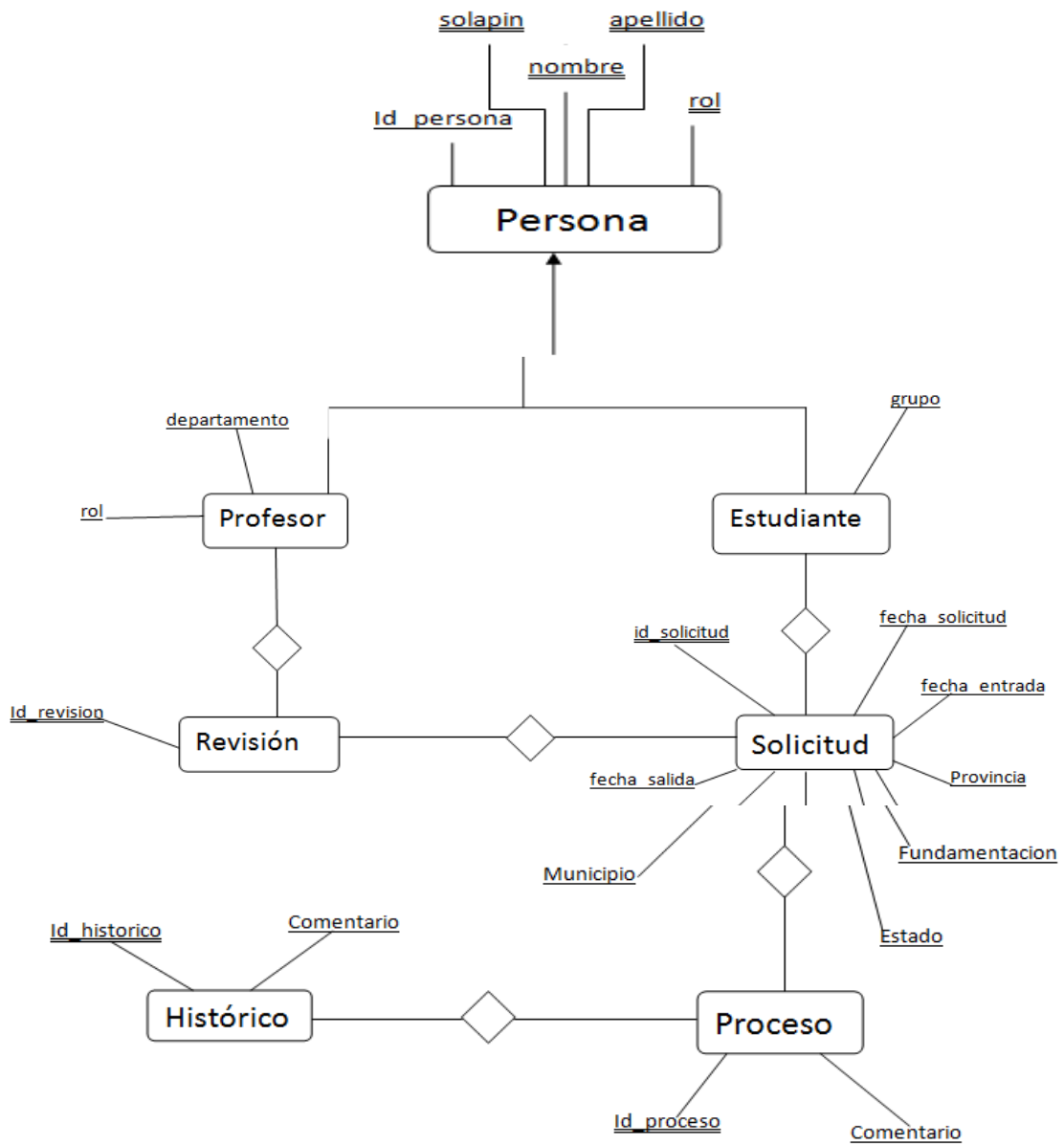


Diagrama 15 Diagrama entidad relación

3.8. Diagrama modelo de datos.

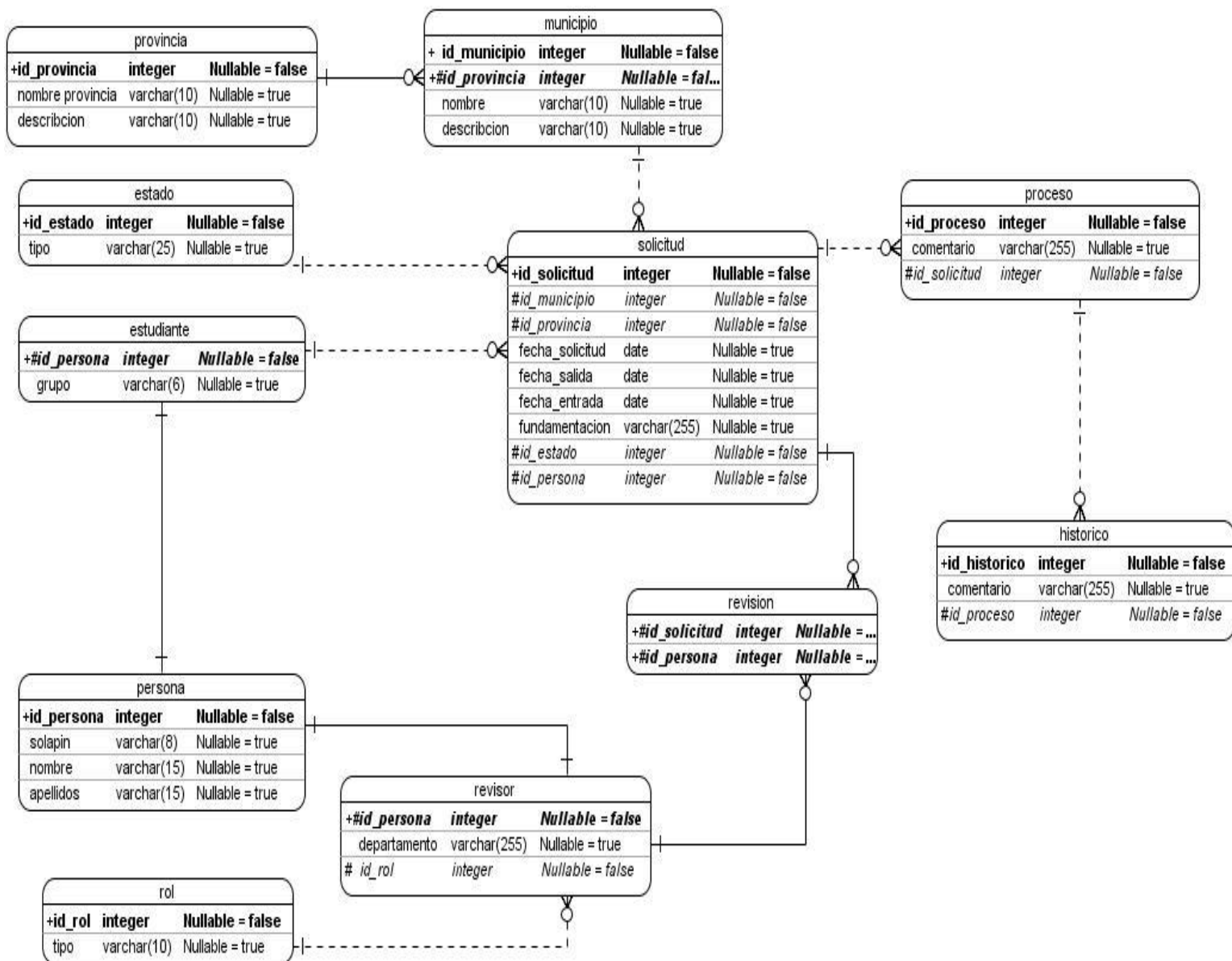


Diagrama 16 Modelo de datos.

3.9. Diagrama Modelo Lógico

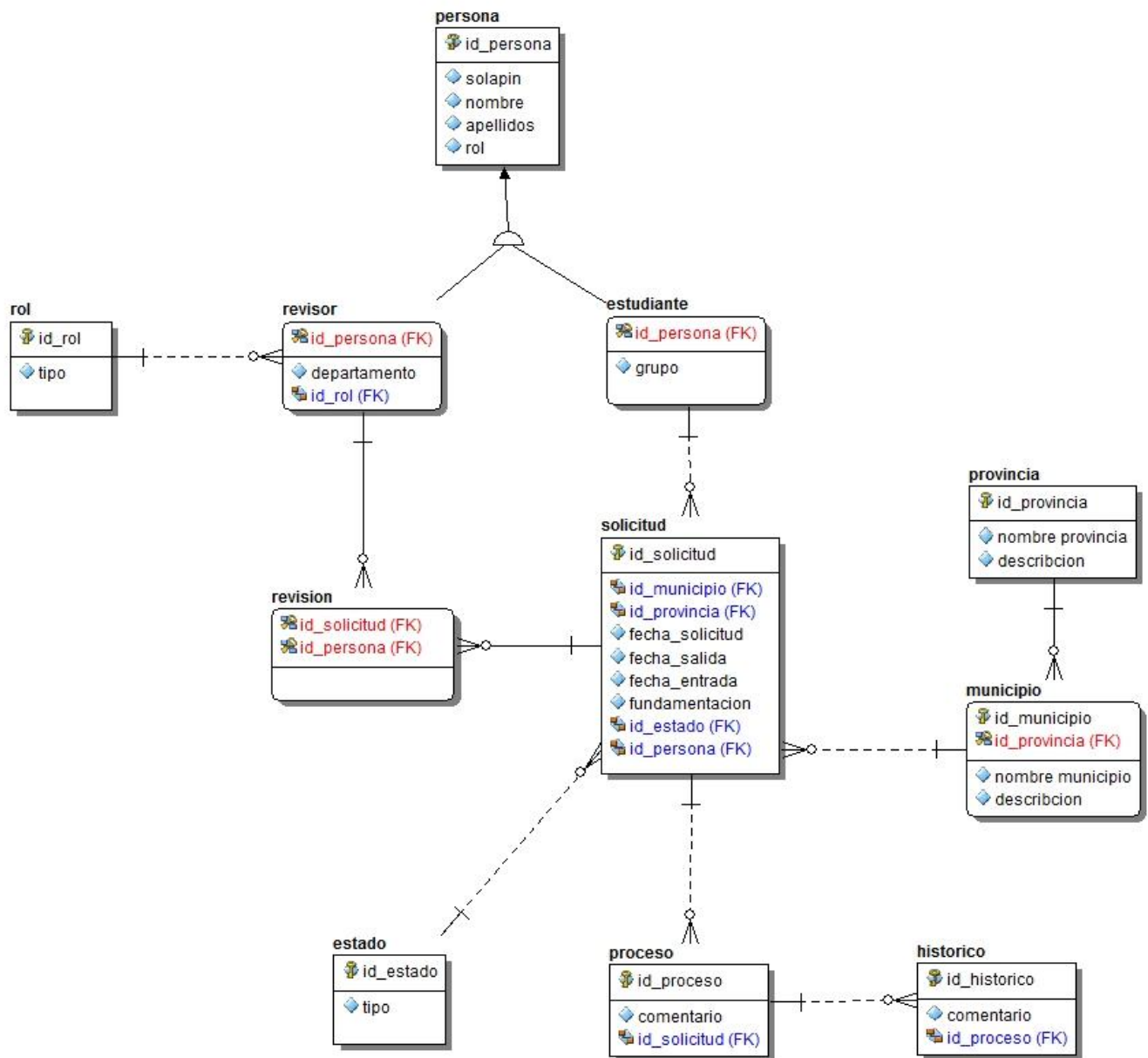


Diagrama 17 Modelo Lógico

3.10. Diagrama Modelo Físico

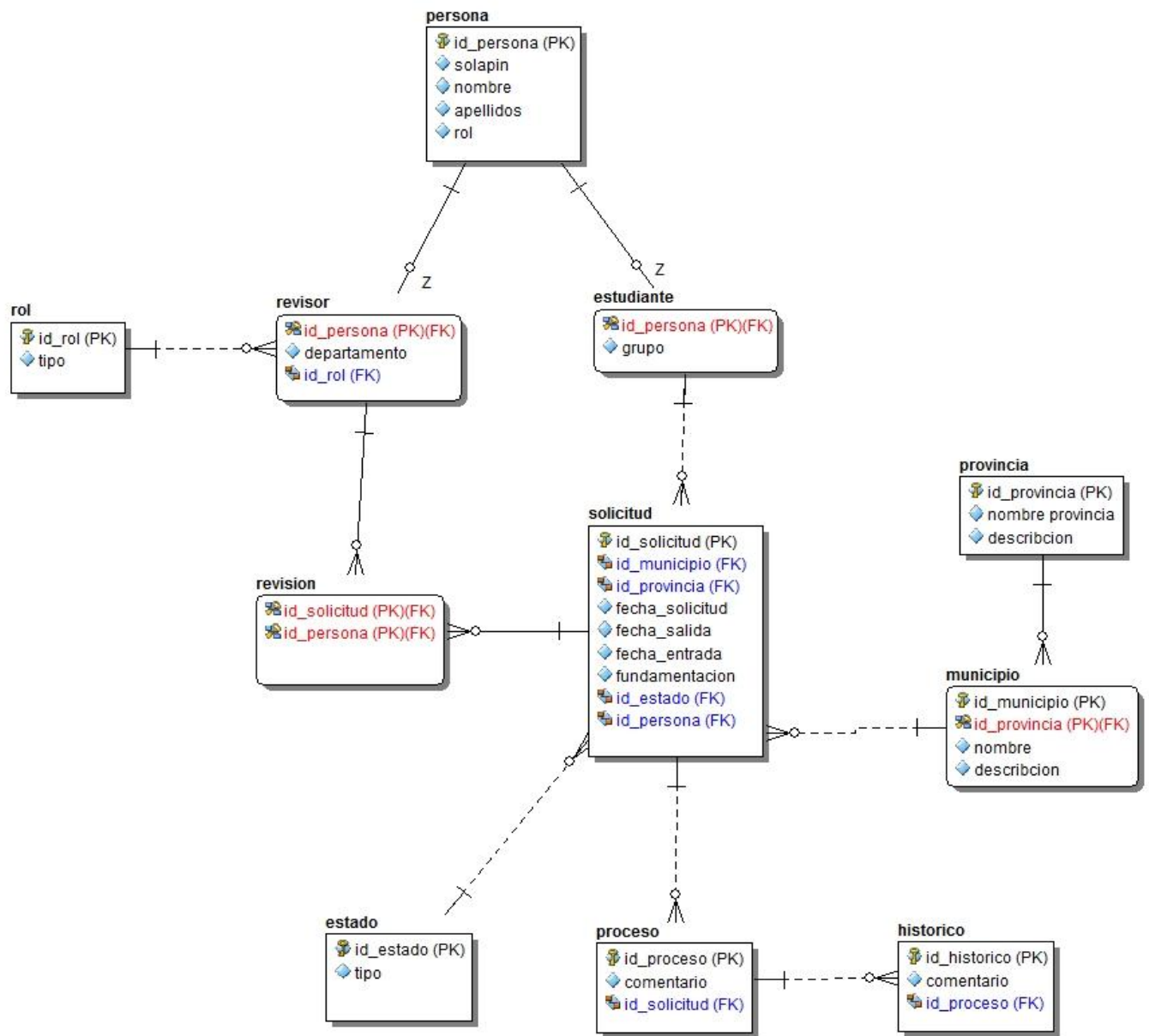


Diagrama 18 Modelo Físico

3.11. Descripciones de las tablas de la base de datos

Nombre: persona		
Descripción:		
Se guarda todos los datos de la persona, sean estudiantes o profesores		
Atributo	Tipo	Descripción
id_persona	Integer	Guarda el id correspondiente al id de la persona
solapín	Varchar	Guarda el id del expediente de la persona
nombre	Varchar	Guarda el nombre de la persona
apellidos	Varchar	Guarda los apellidos de la persona

Tabla 13 Persona

Nombre: revisor		
Descripción:		
Guarda la información referente a una persona de tipo revisor		
Atributo	Tipo	Descripción
departamento	Varchar	Guarda la información referente a el departamento de el profesor

Tabla 14 Revisor

Nombre: rol		
Descripción:		
Guarda la información referente al rol que tienen las personas de tipo profesor		
Atributo	Tipo	Descripción
Tipo	Varchar	Guarda el tipo de permiso.

Tabla 15 Rol

Nombre: estudiante		
Descripción:		
Guarda la información referente a una persona de tipo profesor		
Atributo	Tipo	Descripción
grupo	Varchar	Guarda el grupo de la persona de tipo estudiante.

Tabla 16 Estudiante

Nombre: estado		
Descripción:		
Guarda la información referente al estado de la solicitud de pase.		
Atributo	Tipo	Descripción
Id_estado	Integer	Guarda el id del estado.
Tipo	Varchar	Guarda el tipo de estado

Tabla 17 Estado

Nombre: revisión		
Descripción:		
Guarda la información referente a las personas de tipo profesor que tienen permisos de revisión de solicitud de pases.		
Atributo	Tipo	Descripción
Id _revisor	Integer	Guarda el identificador del revisor

Tabla 18 Revisión

Nombre: solicitud		
Descripción:		
Guarda la información referente a una solicitud de pase		
Atributo	Tipo	Descripción
Id _solicitud	Integer	Guarda el id de la solicitud
Fecha _solicitud	Date	Guarda la fecha de la petición de la solicitud
Fecha _salida	Date	Guarda la fecha de la salida de el pase
Fecha _entrada	Date	Guarda la fecha de la entrada de el pase
fundamentación	Varchar	Guarda la fundamentación de solicitud de el pase

Tabla 19 Solicitud

Nombre: proceso		
Descripción:		
Guarda la información sobre el proceso de una solicitud		
Atributo	Tipo	Descripción
Id _proceso	Integer	Guarda el id de el proceso
Comentario	Varchar	Guarda el comentario sobre la solicitud procesada

Tabla 20 Proceso

Nombre: histórico		
Descripción:		
Guarda el histórico de solicitudes de pase		
Atributo	Tipo	Descripción
Id _historico	Integer	Guarda el id del histórico
Comentario	Varchar	Guarda el comentario del histórico de la solicitud.

Tabla 21 Histórico

Nombre: provincia		
Descripción		
Guarda a la provincia a la que se dirige		

Atributo	Tipo	Descripción
Id_provincia	Integer	Guarda el id de la provincia
nombre	Varchar	Guarda el nombre de la provincia
Descripción	Varchar	Guarda la descripción de la provincia a la que se dirige

Tabla 22 Provincia

Nombre: municipio		
Descripción		
Guarda el municipio al que se dirige.		
Atributo	Tipo	Descripción
Id_municipio	Integer	Guarda el id del municipio
nombre	Varchar	Guarda el nombre del municipio
descripción	Varchar	Guarda la descripción del municipio.

Tabla 23 Municipio

3.12. Conclusiones

En este capítulo se definió la parte de análisis y diseño del sistema, dando a conocer sus diagramas, así como sus respectivas descripciones. En el diseño de la base de datos se da a conocer el modelo lógico y físico de la base de datos, seguidamente las tablas que lo integran junto con los atributos y su definición.

CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA

4.1. Introducción

En este capítulo se dará a conocer los principales diagramas de implementación como son los diagramas de despliegue y componentes, dando a conocer una breve descripción de los casos de prueba de integración que se llevan a cabo sobre el sistema.

4.2. Implementación

En la implementación se empieza con el resultado del diseño e implementando el sistema en términos de componentes y cómo estos se organizan de acuerdo a los nodos específicos en el modelo de despliegue. Se está fuertemente determinado por el lenguaje de programación. Los diagramas de despliegue y componentes conforman lo que se conoce como un modelo de implementación al describir los componentes y construir su organización y dependencia entre nodos físicos en los que funcionará la aplicación.

4.3. Diagrama de despliegue

Un diagrama de despliegue muestra las relaciones físicas entre los componentes *hardware* y *software* en el sistema final, es decir, la configuración de los elementos de procesamiento en tiempo de ejecución y los componentes *software* (procesos y objetos que se ejecutan en ellos).

- **Diagrama de despliegue**

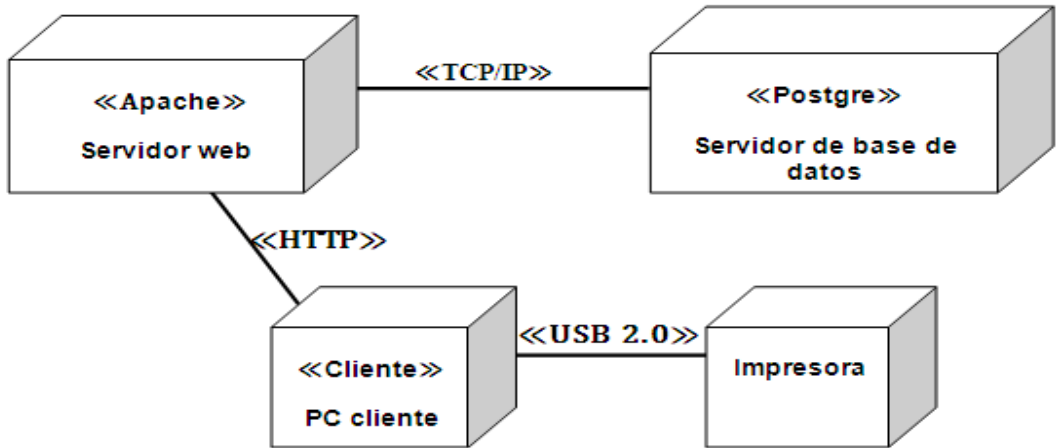


Diagrama 19 Diagrama de Despliegue

4.4. Diagrama de componentes.

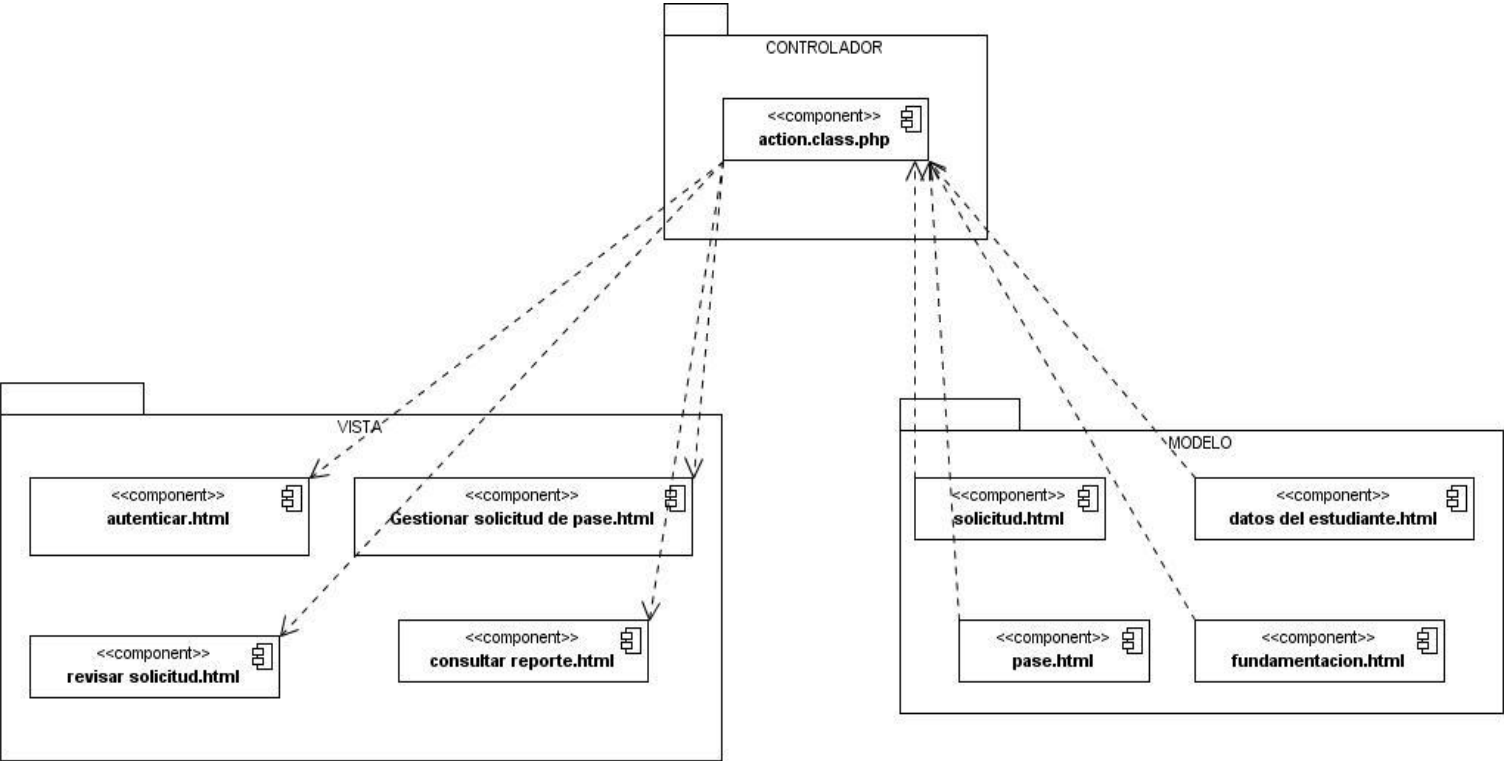


Diagrama 20 Diagrama de Componente

4.5. Modelo de Prueba

El proceso del modelo de prueba es un proceso con la intención de descubrir errores. Es una actividad en la cual un sistema o componente es ejecutado bajo condiciones o requerimientos específicos, los resultados son observados y registrados, realizando una evaluación de algún aspecto de interés.

Se pueden aplicar dos métodos fundamentales que son:

La prueba de caja negra se refiere a las que se llevan a cabo sobre la interfaz del software. O sea, los casos de prueba pretenden demostrar que las funciones del software son operativas, que la entrada se acepta de forma adecuada y que se produce un resultado correcto, así como que la integridad de la información externa se mantiene.

La prueba de la caja blanca se refiere a los caminos lógicos del software proponiendo casos de prueba específicos de condiciones y/o bucles. Se puede examinar el estado del programa en varios puntos para determinar si el estado real coinciden con el esperado o mencionado.

El método aplicado fue el de caja negra en el cual se verifican las especificaciones funcionales y no consideran la estructura interna del programa.

No validan funciones ocultas (por ejemplo funciones implementadas pero no descritas en las especificaciones funcionales del diseño) por tanto los errores asociados a ellas no serán encontrados. Los casos de prueba pretenden demostrar que las funciones del software son operativas. Que las entradas se aceptan de la forma adecuada y que se produzca el resultado correcto.

4.4.1. Descripción de los casos de prueba

Caso de uso Autenticarse

Caso de uso: Autenticarse.
Caso de prueba: Autenticar un estudiante introduciendo correctamente todos los datos.
Entrada: El estudiante introduce correctamente los datos necesarios para autenticarse. Usuario: mperez Contraseña: perez82/05/09m@yo
Resultado: El sistema muestra la página principal con los accesos correspondientes al estudiante autenticado.
Condiciones: El estudiante autenticado debe estar registrado.

Caso de uso: Autenticar estudiante.
Caso de prueba: Autenticar un estudiante introduciendo incorrectamente todos los datos.
Entrada: El estudiante introduce incorrectamente los datos necesarios para autenticarse. Usuario: mper23ez Contraseña: perez82/05/09m@yo
Resultado: El sistema muestra un mensaje para que sepa que tiene algún dato erróneo.
Condiciones:

Caso de uso Gestionar solicitud de pase.

Escenario modificar solicitud de pase

Caso de uso: gestionar solicitud de pase
Caso de prueba modificar solicitud introduciendo datos correctamente
Entrada: El estudiante introduce correctamente los datos para modificar la solicitud de pase.
Resultado:

El sistema actualiza los datos del usuario
Condiciones: El estudiante debe de estar registrado y debe de estar autenticado

Caso de uso: gestionar solicitud de pase
Caso de prueba modificar solicitud de pase introduciendo datos erróneos.
Entrada: El estudiante introduce incorrectamente algún dato para la modificación de la solicitud.
Resultado: El sistema muestra un mensaje para que comprenda que tiene algún dato erróneo.
Condiciones:

Escenario realizar solicitud de pase

Caso de uso: gestionar solicitud de pase
Caso de prueba realizar solicitud de pase introduciendo datos correctamente
Entrada: El estudiante introduce correctamente los datos para realizar la solicitud de pase.
Resultado: El sistema registra los datos del estudiante a la base de datos.
Condiciones: Los datos del estudiante deben de ser válidos para la solicitud de pase.

Caso de uso: gestionar solicitud de pase
Caso de prueba realizar solicitud de pase introduciendo datos erróneos.
Entrada: El estudiante introduce incorrectamente algún dato para realizar una solicitud.
Resultado: El sistema muestra un mensaje para que comprenda que tiene algún dato erróneo.
Condiciones:

4.6. Conclusiones.

En este capítulo se desarrolló las tareas relacionadas con la implementación y el modelo de prueba, dando a conocer unos de los diagrama que ponen de manifiesto el entorno del trabajo en términos de Hardware y Software que es el diagrama de despliegue, donde se mostró la estructura física del sistema representándolo con los diagramas de despliegue. En el modelo de prueba se realizó el de caja negra comprobando las validaciones y funcionalidades que debe realizar la aplicación, donde detectar algunos errores y corregirlos.

CONCLUSIONES

El desarrollo de este trabajo de diploma se orientó al análisis y diseño de un Sistema de Control de Pases para la Facultad 15. Dicho sistema proporcionará una vía de solución para el desarrollo del proceso de reservación de pases extras de la Facultad 15.

Al concluir el trabajo de diploma se dio cumplimiento al objetivo general del trabajo: Desarrollar el análisis y diseño del sistema para el control de pases extras de los estudiantes de la Facultad 15.

Se realizó un estudio de los sistemas informáticos relacionados con la gestión de pases extras y de capital humano. Se obtuvieron los artefactos del análisis y diseño y se obtuvieron además los prototipos funcionales de la implementación.

RECOMENDACIONES

Continuar con el estudio del proceso para el sistema de reservación de pases extras de los estudiantes.

Continuar con el estudio de la aplicación.

Estudiar las posibilidades de integrar el sistema como un módulo a otras entidades que necesiten un servicio para la gestión de las solicitudes de pases extras de los estudiantes.

Incrementar otras funcionalidades.

Hacer un estudio de diferentes herramientas y metodologías para perfeccionar la aplicación.

TRABAJOS CITADOS

Desoft S.A. 2008. Desoft S.A. *Desoft S.A.* [En línea] Desoft S.A., 2008. [Citado el: 10 de Marzo de 2010.] <http://www.desoft.cu/Productos1/FastosPagus/tabid/442/Default.aspx>.

Gallego Vázquez, José Antonio. 2003. *Desarrollo Web con PHP y MySQL*. Madrid : Anaya, 2003. 84-415-1525-5.

Group, The PostgreSQL Global Development. 2010. *PostgreSQL 8.4.2 Documentation*. California : University of California., 2010.

Intellisoft SA. 2007. Intellisoft. *Intellisoft*. [En línea] Intellisoft SA, 2007. [Citado el: 10 de Marzo de 2010.] <http://intellisoftsa.com/cms/index.php>.

Kair, M. J. 2003. *La Biblia de el Servidor Apache2*. Madrid, España : Anaya Multimedia, 2003.

Marín, Y.D.C. 2005. *Sistema de Gestión Académica: Módulo de Gestión de Profesores*. Ciudad de la Habana : Instituto Superior Politécnico “José Antonio Echeverría”, 2005. Disponible en [[http://biblioteca.reduc.edu.cu/biblioteca.virtual/cgi/CD-ROM/otros/UCIENCIA%202007%20\(E\)/ponencias/tcis/PDF/PROPUESTA%20DE%20TAREAS-2489342976/PROPUE~1.pdf](http://biblioteca.reduc.edu.cu/biblioteca.virtual/cgi/CD-ROM/otros/UCIENCIA%202007%20(E)/ponencias/tcis/PDF/PROPUESTA%20DE%20TAREAS-2489342976/PROPUE~1.pdf)].

Mato Garcia, R.M. 1999. *Diseño de Base de Datos*. Cuba : Pueblo y Educación, 1999. 959-13-1273-3.

Matthew N, Stones R. 2005. *Beginning databases with PostgreSQL: from novice to professional. 2nd Edition*. United States of America : Apress, 2005. 1-59059-478-9.

Potencier F, Zaninotto F. 2008. *Symfony. La guía definitiva*. 2008.

T., Ozdemir. 2007. *Actionscript.org. Actionscript.org*. [En línea] 28 de Octubre de 2007. [Citado el: 10 de Marzo de 2010.] <http://www.actionscript.org/resources/articles/679/1/MVC-Design-Pattern/Page1.html>.

TOPDATA Sistemas de Automatización. 2007. COMTEL. *COMTEL*. [En línea] COMTEL EL SALVADOR - DIVISIÓN DE OFICENTER S.A. DE C.V., 2007. [Citado el: 10 de Marzo de 2010.] <http://www.comtel.com.sv/paginas/innerplus.html>.

—. 2007. COMTEL. *COMTEL*. [En línea] COMTEL EL SALVADOR - DIVISIÓN DE OFICENTER S.A. DE C.V., 2007. [Citado el: 10 de Marzo de 2010.] <http://www.comtel.com.sv/paginas/t1000bio.html>.

UCI1. 2009. E.V.A. *Conferencia 5 Fase de Inicio. Modelo del Negocio.* [En línea] UCI, 6 de Octubre de 2009. [Citado el: 10 de Marzo de 2010.] <http://eva.uci.cu/mod/resource/view.php?id=21010>.

UCI2. 2009. E.V.A. *Conferencia 6 Fase de Inicio. Disciplina de Requisitos.* [En línea] UCI, 3 de Noviembre de 2009. [Citado el: 10 de Marzo de 2010.] <http://eva.uci.cu/mod/resource/view.php?id=22095>.

UCI3. 2009. E.V.A. *Conferencia 7 Fase de Elaboración. Flujo de trabajo de Análisis y Diseño.* [En línea] UCI, 30 de Noviembre de 2009. [Citado el: 10 de Marzo de 2010.] <http://eva.uci.cu/mod/resource/view.php?id=22668>.

Vargas Allegue E. D., Fernández Martínez Y. 2009. *CERES. Definición de un sistema Integral de Gestión de Recursos Humanos.* Habana, Cuba : Universidad de las Ciencias Informáticas, 2009.

Visual Paradigm. 2010. Visual Paradigm International. *Visual Paradigm International.* [En línea] 29 de Marzo de 2010. [Citado el: 10 de Mayo de 2010.] <http://www.visual-paradigm.com/product/vpuml/>.

BIBLIOGRAFÍA

1. **Visual Paradigm.** Visual Paradigm International. *Visual Paradigm International*. [En línea] 29 de Marzo de 2010. [Citado el: 10 de Mayo de 2010.] <http://www.visual-paradigm.com/product/vpum/>.
2. **Potencier F, Zaninotto F.** *Symfony. La guía definitiva*. 2008.
3. **Pérez Marrero, Y y Valdés Martí, A.** *Sistema de Reservación para la Salida Semestral de Profesores*. Ciudad Habana : Universidad de las Ciencias Informáticas, 2007.
4. **Marín, Y.D.C.** *Sistema de Gestión Académica: Módulo de Gestión de Profesores*. Ciudad de la Habana : Instituto Superior Politécnico “José Antonio Echeverría”, 2005. Disponible en [[http://biblioteca.reduc.edu.cu/biblioteca.virtual/cgi/CD-ROM/otros/UCIENCIA%202007%20\(E\)/ponencias/tcis/PDF/PROPUESTA%20DE%20TAREAS-2489342976/PROPUE~1.pdf](http://biblioteca.reduc.edu.cu/biblioteca.virtual/cgi/CD-ROM/otros/UCIENCIA%202007%20(E)/ponencias/tcis/PDF/PROPUESTA%20DE%20TAREAS-2489342976/PROPUE~1.pdf)]
5. **Group, The PostgreSQL Global Development.** *PostgreSQL 8.4.2 Documentation*. California : University of California., 2010.
6. **Kair, M. J.** *La Biblia de el Servidor Apache2*. Madrid, España : Anaya Multimedia, 2003.
7. **Intellisoft SA.** Intellisoft. *Intellisoft*. [En línea] Intellisoft SA, 2007. [Citado el: 10 de Marzo de 2010.] <http://intellisoftsa.com/cms/index.php>.
8. **UCI3.** E.V.A. *Conferencia 7 Fase de Elaboración. Flujo de trabajo de Análisis y Diseño*. [En línea] UCI, 30 de Noviembre de 2009. [Citado el: 10 de Marzo de 2010.] <http://eva.uci.cu/mod/resource/view.php?id=22668>.
9. **UCI2.** E.V.A. *Conferencia 6 Fase de Inicio. Disciplina de Requisitos*. [En línea] UCI, 3 de Noviembre de 2009. [Citado el: 10 de Marzo de 2010.] <http://eva.uci.cu/mod/resource/view.php?id=22095>.
10. **UCI1.** E.V.A. *Conferencia 5 Fase de Inicio. Modelo del Negocio*. [En línea] UCI, 6 de Octubre de 2009. [Citado el: 10 de Marzo de 2010.] <http://eva.uci.cu/mod/resource/view.php?id=21010>.
11. **Mato Garcia, R.M.** *Diseño de Base de Datos*. Cuba : Pueblo y Educación, 1999. 959-13-1273-3.
12. **Desoft S.A.** Desoft S.A. *Desoft S.A.* [En línea] Desoft S.A., 2008. [Citado el: 10 de Marzo de 2010.] <http://www.desoft.cu/Productos1/FastosPagus/tabid/442/Default.aspx>.
13. **Gallego Vázquez, José Antonio.** *Desarrollo Web con PHP y MySQL*. Madrid : Anaya, 2003. 84-415-1525-5.

14. **TOPDATA Sistemas de Automatización.** COMTEL. *COMTEL*. [En línea] COMTEL EL SALVADOR - DIVISIÓN DE OFICENTER S.A. DE C.V., 2007. [Citado el: 10 de Marzo de 2010.] <http://www.comtel.com.sv/paginas/t1000bio.html>.
15. —. COMTEL. *COMTEL*. [En línea] COMTEL EL SALVADOR - DIVISIÓN DE OFICENTER S.A. DE C.V., 2007. [Citado el: 10 de Marzo de 2010.] <http://www.comtel.com.sv/paginas/innerplus.html>.
16. **Vargas Allegue E. D., Fernández Martínez Y.** *CERES. Definición de un sistema Integral de Gestión de Recursos Humanos.* Habana, Cuba : Universidad de las Ciencias Informáticas, 2009.
17. **Matthew N, Stones R.** *Beginning databases with PostgreSQL: from novice to professional. 2nd Edition.* United States of America : Apress, 2005. 1-59059-478-9.
18. **T., Ozdemir.** *Actionscript.org. Actionscript.org.* [En línea] 28 de Octubre de 2007. [Citado el: 10 de Marzo de 2010.] <http://www.actionscript.org/resources/articles/679/1/MVC-Design-Pattern/Page1.html>.
19. **Lamar Febles, A y García García, Y.** *Desarrollo de un sistema de gestión y control de Información del Servicio de Transporte en la UCI.* Ciudad Habana : Universidad de las Ciencias Informáticas, 2007.
20. **Araujo Pérez, D y Y., González Nuzzo.** *Sistema para el control de Recursos Humanos. Proceso de reclutamiento en los niveles de dirección.* Ciudad Habana : Universidad de las Ciencias Informáticas., 2007.
21. **Santos Rivas, A y H., Leyva Avila.** *Sistema Informatizado de Gestión de Contratos Profesor Invitado.* Ciudad Habana : Universidad de las Ciencias Infomáticas, 2007.
22. **Montoya Rodríguez, Elda.** *Sistema de Gestión de Información de los Recursos Humanos (SGIRH) de la Facultad 2.* Ciudad Habana : Universidad de las Ciencias Informáticas., 2009.
23. **Artículos Informativos USA.** *Articulos Informativos. Sistemas de Control de Acceso.* [En línea] [Citado el: 7 de Febrero de 2010.] <http://www.articulosinformativos.com>.
24. **SmartCard Systems S.A.** *SmartCard Systems S.A. SmartCard Systems S.A.* [En línea] SmartCard Systems S.A., 2010. [Citado el: 7 de Febrero de 2010.] <http://www.scssa.com.ar/>.

ANEXOS

Prototipos funcionales.

Página principal, de la aplicación donde el usuario pueda tener una explicación del objetivo del sistema.



Página donde el usuario en este caso el estudiante pueda realizar, modificar y cancelar la solicitud de pase.

The screenshot shows a web browser window displaying the 'SISTEMA DE RESERVACIÓN DE PASES FACULTAD 15' application. The browser's address bar shows 'Sf 1.4.5' and various icons for configuration, view, logs, and file size (7680.0 KB) and load time (136 ms).

The application header features the title 'SISTEMA DE RESERVACIÓN DE PASES' and 'FACULTAD 15' on a blue background with a wavy pattern. Below the header is a navigation bar with the following menu items: INICIO, RESERVAR, MODIFICAR, CANCELAR, and SALIR.

The main content area is divided into three sections:

- VÍNCULOS:** A list of links with star icons: INTRANET, INTER-NOS, CORREO, FEU, and FACULTAD 15.
- BIENVENIDOS:** A welcome message: 'Esta aplicación tiene como objetivo proporcionarle a los profesores y a los estudiantes gestionar todo el trabajo referente a las solicitudes de pases extras, llevando el control detallado de todos los datos necesarios para emitir un pase a un estudiante, y permitiéndole a el profesor y a la vicedecana o decana de la facultad aceptar o rechazar la solicitud de pase.'
- USUARIO:** A user profile section featuring a photo of Oscar Luis Garcell Martínez and the following details:
 - Nombres: Oscar Luis
 - Apellidos: Garcell Martínez
 - Correo: olgarcell@estudiantes.uci.cu
 - Solapín: 58522
 - Rol: Estudiante

At the bottom of the page, there is a copyright notice: 'Copyright © 2010 --- Todos los derechos reservados.'

Página donde el usuario reserva una solicitud de pase, agregando la fecha de salida y la de entrada, municipio, provincia a la que se dirige, y la fundamentación que no es más que el motivo por el cual el estudiante está solicitando el pase.

The screenshot shows a web application interface with a blue header and a light blue background. The header contains navigation links: INICIO, RESERVAR, MODIFICAR, CANCELAR, and SALIR. On the left, there is a 'VÍNCULOS' section with a gear icon and a list of links: INTRANET, INTER-NOS, CORREO, FEU, and FACULTAD 15. The main content area is divided into three sections. The central section is titled 'RESERVAR' and contains a form with the following fields: 'Fecha Salida:' (text input), 'Fecha Entrada:' (text input), 'Provincia:' (dropdown menu with 'P del Río' selected), 'Municipio:' (dropdown menu with 'P. de Río' selected), and 'Fundamentación:' (text area). Below the form are three buttons: 'Nuevo', 'Aceptar', and 'Cancelar'. To the right of the form is a 'USUARIO' section with a gear icon, a profile picture of a young man, and the following text: 'Nombres: Oscar Luis', 'Apellidos: Garcell Martínez', 'Correo: olgarcell@estudiantes.uci.cu', 'Solapín: 58522', and 'Rol: Estudiante'. At the bottom of the page, there is a footer with the text: 'Copyright © 2010 ---. Todos los derechos reservados.'

GLOSARIO DE TÉRMINOS

1. **CVS:** El Concurrent Versions System (CVS), también conocido como Concurrent Versioning System, es una aplicación informática que implementa un sistema de control de versiones: mantiene el registro de todo el trabajo y los cambios en los ficheros (código fuente principalmente) que forman un proyecto (de programa) y permite que distintos desarrolladores (potencialmente situados a gran distancia) colaboren. CVS se ha hecho popular en el mundo del software libre.
2. **ECJ:** Es el sistema evolutivo de computo de la investigación escrito en java. Es un marco que apoya una variedad de técnicas evolutivas del computo, por ejemplo algoritmos genéticos, programación genética, estrategias de la evolución, y evolución diferenciada
3. **HTML:** Se trata de la sigla de HyperText Markup Language, es decir, Lenguaje de Marcas de Hipertexto. EL HTML permite describir la estructura y el contenido en forma de texto.
4. **Java:** Java es un lenguaje de programación orientado a objetos desarrollado por Sun Microsystems a principios de los años 90
5. **Jsp:** Java Server Pages (JSP) es una tecnología Java que permite generar contenido dinámico para web, en forma de documentos HTML, XML o de otro tipo.
6. **Logs:** trazas de las cosas que pasan en una aplicación. Es un registro oficial de eventos durante un rango de tiempo en particular
7. **Microsoft SQL Server:** Microsoft SQL Server es un sistema para la gestión de bases de datos producido por Microsoft basado en el modelo relacional
8. **MySQL:** MySQL es un sistema de gestión de base de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones
9. **Oracle:** Herramienta cliente servidor para la gestión de base datos
10. **Perl:** Perl es un lenguaje de propósito general originalmente desarrollado para la manipulación de texto.
11. **Postgres:** Sistema de gestión de base de datos relacional orientada a objeto.
12. **Python :** Python es un lenguaje de programación interpretado
13. **Script:** En informática, un script es un conjunto de instrucciones. Permiten la automatización de tareas, creando pequeñas utilidades. Es muy utilizado para la administración de sistemas UNIX. Son ejecutados por un intérprete de línea de órdenes y usualmente son archivos de texto.

Glosario de términos

14. **Scripting:** Un *script* (cuya traducción literal es *guión*) o archivo de órdenes o archivo de procesamiento por lotes es un programa usualmente simple, que por lo regular se almacena en un archivo de texto plano.
15. **SDK de eclipse:** Software development kit, o sea, kit de desarrollo de software, es el id del eclipse, como un set de herramientas integradas orientadas al desarrollo del software.
16. **SSL:** Secure socket layer, es un protocolo de comunicación, se encuentra en la capa TCP/IP del modelo OSI.
17. **PHP:** Lenguaje de programación para desarrollo web.
18. **.World Wide Web:** En informática, la World Wide Web, cuya traducción podría ser *Red Global Mundial* o "Red de Amplitud Mundial", es un sistema de documentos de hipertexto o hipermedios enlazados y accesibles a través de Internet.