



Universidad de las Ciencias Informáticas



cujae

Instituto Superior Politécnico "José Antonio Echeverría"  
INGENIERIA INFORMÁTICA

# Sistema de Control de Acceso

Trabajo para optar por el Título de Ingeniería en Informática

**Autor**

Manuel Alejandro Gil Martín

**Tutores**

Ing. Renier Pérez García  
Ing. Marlon Ramiro Muñoz Miranda

Ciudad de la Habana  
Julio de 2005

## DECLARACION DE AUTORIA

Yo: **Manuel Alejandro Gil Martín** me declaro como único autor de este trabajo y autorizo la Universidad de las Ciencias Informáticas (UCI) y al Centro de Estudios de Ingeniería y Sistemas (CEIS) de la CUJAE para que hagan el uso de este trabajo de la manera que estimen conveniente.

Y para que así conste firmo la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del 2005.

\_\_\_\_\_  
Firma del Autor

\_\_\_\_\_  
Firma del Tutor

# OPINIÓN DEL USUARIO DEL TRABAJO DE DIPLOMA

El Trabajo de Diploma, titulado: “**Sistema de Control de Acceso**”, fue realizado en la Universidad de Ciencias Informáticas. Este centro considera que, en correspondencia con los objetivos trazados, el trabajo realizado le satisface:

- Totalmente
- Parcialmente en un \_\_\_\_\_ %

Los resultados de este Trabajo de Diploma le reportan a esta Universidad los beneficios siguientes:

---

---

---

---

---

---

---

---

---

---

Como resultado de la implantación de este trabajo se reportará un efecto económico que asciende a **<valor en MN o USD del efecto económico>**

Y para que así conste, se firma la presente a los \_\_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_

\_\_\_\_\_  
Representante de la entidad

\_\_\_\_\_  
Cargo

\_\_\_\_\_  
Firma

\_\_\_\_\_  
Cuño

# OPINIÓN DEL TUTOR DEL TRABAJO DE DIPLOMA

**Título:** Sistema de Control de Acceso

**Autor:** Manuel Alejandro Gil Martín

El tutor del presente Trabajo de Diploma considera que durante su ejecución el estudiante mostró las cualidades que a continuación se detallan.

<Aquí el tutor debe expresar cualitativamente su opinión y medir (usando la escala: muy alta, alta, adecuada) entre otras las cualidades siguientes:

- Independencia
  - Originalidad
  - Creatividad
  - Laboriosidad
  - Responsabilidad >

< Además, debe evaluar la calidad científico-técnica del trabajo realizado (resultados y documento) y expresar su opinión sobre el valor de los resultados obtenidos (aplicación y beneficios) >

Por todo lo anteriormente expresado considero que el estudiante está apto para ejercer como Ingeniero Informático; y propongo que se le otorgue al Trabajo de Diploma la calificación de <nota>. <Además, si considera que los resultados poseen valor para ser publicados, debe expresarlo también>

---

Firma

---

Fecha

***“Se debe hacer todo tan sencillo como sea posible, pero no más sencillo.”***

***Albert Einstein***

## **AGRADECIMIENTOS**

A mis padres, por haberme guiado siempre.

A Irela, mi novia, por su amor y por su sonrisa.

A mis amigos Ronny González, José Fidalgo, Julio Cesar Isaza, Yanier Castellanos, Yudiel Tamayo y Yanesky Montero, “El Escuadrón Metelapata”, por hacer que estos 5 años fueran inolvidables.

A Yahima Vigo, Alexis Naranjo, Enrique Altuna, Maykel Pereira, José Enrique Saura, por haber puesto sus granitos de arena en la construcción del sistema.

A Maykel Santiesteban, Giorbis Lorie Montalvo, Damián Ilizastegui, José Antonio Pla, Asnioby Hernández, por su colaboración a lo largo del curso.

A Manuel Vázquez, Febe Angel Ciudad Ricardo, Wilberto Martínez, Yaneisy Cruz, Arturo Cesar Areas Orizondo, Lissette Soto, por dedicarme su tiempo para aclarar mis dudas, revisar la tesis, o ayudarme con la impresión.

A la Universidad de Camagüey, y la CUJAE.

A Renier Pérez y Marlon Muñoz, mis tutores.

A la Revolución, por haberme dado la posibilidad de realizar mis estudios y obtener los conocimientos necesarios para realizar este trabajo.

...y a todos los que no puse, pero que de una forma u otra han contribuido a mi formación profesional y personal, que no se me olvidan, pero no habría espacio, a todos, muchas gracias.

*...a mis padres*

*...a Irela*

## RESUMEN

El Control de Acceso a la Universidad de Ciencias Informáticas es un proceso de suma importancia en el contexto de la seguridad en la Universidad. Actualmente, este flujo de información, ocurre de manera manual, lo que trae como consecuencia demoras, ineficiencia, gasto de recursos, y vulnerabilidades en la seguridad del centro. En la Universidad hay un gran flujo de personas, tanto trabajadores como visitantes, esto debe estar sometido a un control eficaz para garantizar el buen funcionamiento interno.

Así surge el Sistema de Control de Acceso a la UCI, y tiene como **objetivo concreto** automatizar el flujo de información y los procesos que tienen que ver con el acceso de las personas pertenecientes al centro y la gestión de visitas de personas externas al mismo. En conjunto con otros sistemas, como el de Acreditación y Reservación de Pase, entre otros, puede ayudar a la automatización de muchos procesos que se realizan manualmente y a la generación automática de reportes estadísticos y otras facilidades, brindando a los directivos una herramienta para controlar procesos de importancia.

En este documento se plasman los resultados del estudio realizado en UCI para la construcción del sistema, se explican los conceptos relacionados con el mismo, se hace un análisis de la propuesta del sistema, y se dejan algunas recomendaciones para el mejoramiento futuro del mismo.



## ÍNDICE

<b>INTRODUCCIÓN.....</b>	<b>1</b>
<b>CAPÍTULO 1. FUNDAMENTACIÓN DEL TEMA.....</b>	<b>7</b>
1.1 INTRODUCCIÓN .....	7
1.2 QUÉ ES, Y POR QUÉ ES NECESARIO EL CONTROL DE ACCESO.....	8
1.3 SISTEMAS DE IDENTIFICACIÓN .....	8
1.4 OBJETO DE ESTUDIO .....	9
1.5 DESCRIPCIÓN DEL PROCESO DE NEGOCIO ACTUAL .....	9
1.6 SITUACIÓN PROBLÉMICA .....	10
1.7 SISTEMAS AUTOMATIZADOS EXISTENTES VINCULADOS AL CAMPO DE ACCIÓN .....	11
1.8 SOLUCIONES EXISTENTES .....	12
1.9 PROPUESTA DE SOLUCIÓN.....	12
1.10 FUNDAMENTACIÓN DE LOS OBJETIVOS QUE SE PROPONE EL TRABAJO.....	13
1.10.1 <i>Objetivo General</i> .....	13
1.10.2 <i>Objetivos Específicos</i> .....	13
1.11 CONCLUSIONES .....	13
<b>CAPÍTULO 2. TENDENCIAS Y TECNOLOGÍAS ACTUALES A CONSIDERAR .....</b>	<b>14</b>
2.1 INTRODUCCIÓN .....	14
2.2 FUNDAMENTACIÓN DE LAS TECNOLOGÍAS EN LAS QUE SE BASA LA PROPUESTA.....	15
2.3 TECNOLOGÍAS DE HARDWARE UTILIZADAS PARA EL CONTROL DEL ACCESO .....	15
2.3.1 <i>Claves por Teclado</i> .....	15
2.3.2 <i>Tarjetas de Banda Magnética</i> .....	15
2.3.3 <i>Tarjetas de Código de Barras</i> .....	16
2.3.4 <i>Touch Memories</i> .....	16
2.3.5 <i>Tarjetas de Proximidad o Radio Frecuencia (RF)</i> .....	17
2.3.6 <i>Sistemas Biométricos</i> .....	17
2.4 TECNOLOGÍA UTILIZADA EN LA UCI: EL CÓDIGO DE BARRAS .....	18
2.4.1 <i>Beneficios del código de barras</i> .....	19
2.4.2 <i>Tipos de lectores de códigos de barras</i> .....	19
2.5 TECNOLOGÍAS DE SOFTWARE A CONOCER .....	22
2.5.1 <i>Plataforma .NET</i> .....	22
2.5.2 <i>ASP.NET</i> .....	22
2.5.3 <i>Lenguaje C#</i> .....	23
2.5.4 <i>Web Services</i> .....	24
2.5.5 <i>SQL</i> .....	25
2.6 FUNDAMENTACIÓN DEL GESTOR DE BASES DE DATOS UTILIZADO .....	26
2.7 FUNDAMENTACIÓN DE LA METODOLOGÍA A UTILIZAR .....	26
2.7.1 <i>RUP (Rational Unified Process)</i> .....	26
2.7.2 <i>UML (Unified Modeling Language)</i> .....	27
2.8 HERRAMIENTAS UTILIZADAS .....	28
2.9 LA PROPUESTA. ....	29
2.10 CONCLUSIONES .....	30
<b>CAPÍTULO 3. DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA.....</b>	<b>31</b>
3.1 INTRODUCCIÓN. ....	31
3.2 REGLAS DE NEGOCIO. ....	32
3.3 ACTORES Y TRABAJADORES DEL NEGOCIO .....	33
3.4 DESCRIPCIÓN DE LOS CASOS DE USO DEL MODELO DEL NEGOCIO ACTUAL .....	34

3.4.1 Diagrama de Casos de Uso del modelo del negocio actual .....	34
3.4.2. Expansión de los Casos de Uso .....	35
3.5 DIAGRAMA DE CLASES DEL MODELO DE OBJETOS .....	42
3.6 REQUERIMIENTOS FUNCIONALES .....	43
3.7 REQUERIMIENTOS NO FUNCIONALES .....	44
3.8 DESCRIPCIÓN DEL SISTEMA PROPUESTO .....	46
3.9 DEFINICIÓN DE LOS ACTORES DEL SISTEMA.....	48
3.10 MODELO DE CASOS DE USO DEL SISTEMA.....	48
3.11 EXPANSIÓN DE LOS CASOS DE USO .....	50
3.12 CONCLUSIONES .....	67
<b>CAPÍTULO 4. CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA .....</b>	<b>68</b>
4.1 INTRODUCCIÓN.....	68
4.2 DIAGRAMA DE CLASES .....	69
4.3 DISEÑO DE LA BASE DE DATOS .....	78
4.3.1 Diagrama de clases persistentes.....	78
4.3.2 Modelo de datos.....	79
4.4 PRINCIPIOS DE DISEÑO .....	81
4.4.1 Estándares de la Interfaz de la aplicación.....	81
4.4.2 Formato de Reportes .....	82
4.4.3 Concepción general de la ayuda.....	83
4.4.4 Tratamiento de excepciones.....	83
4.5 ESTÁNDARES DE CODIFICACIÓN .....	84
4.6 MODELO DE DESPLIEGUE.....	86
4.7 MODELO DE COMPONENTES.....	87
4.7.1 Explicación de los componentes .....	89
4.8 CONCLUSIONES .....	93
<b>CAPÍTULO 5. ESTUDIO DE FACTIBILIDAD .....</b>	<b>94</b>
5.1 INTRODUCCIÓN .....	94
5.2 PLANIFICACIÓN.....	94
5.3 COSTOS .....	97
5.4 BENEFICIOS TANGIBLES E INTANGIBLES .....	99
5.5 ANÁLISIS DE COSTOS Y BENEFICIOS .....	100
5.6 CONCLUSIONES .....	101
<b>CONCLUSIONES .....</b>	<b>102</b>
<b>RECOMENDACIONES .....</b>	<b>103</b>
<b>GLOSARIO DE TÉRMINOS .....</b>	<b>104</b>
<b>REFERENCIAS BIBLIOGRÁFICAS.....</b>	<b>105</b>
<b>BIBLIOGRAFÍA .....</b>	<b>106</b>

## ÍNDICE DE FIGURAS

FIGURA 1. <i>CÓDIGO 39.</i> .....	18
FIGURA 2. <i>LOS SERVICIOS WEB PUEDEN CONECTAR A DIVERSOS TIPOS DE APLICACIONES.</i> .....	25
FIGURA 3. <i>DIAGRAMA DE CASOS DE USO DEL NEGOCIO.</i> .....	34
FIGURA 4. <i>DIAGRAMA DE ACTIVIDADES DEL C.U. "SOLICITAR VISITAS"</i> .....	36
FIGURA 5. <i>DIAGRAMA DE ACTIVIDADES DEL C.U. "REALIZAR VISITA"</i> .....	38
FIGURA 6. <i>DIAGRAMA DE ACTIVIDADES DEL C.U. "SOLICITAR PERMISO PARA MEDIO"</i> .....	39
FIGURA 7. <i>DIAGRAMA DE ACTIVIDADES DEL C.U. "ENTRAR O SALIR DE LA UCI"</i> .....	41
FIGURA 8. <i>DIAGRAMA DE CLASES DEL MODELO DE OBJETOS DEL NEGOCIO</i> .....	42
FIGURA 9. <i>DIAGRAMA DE CASOS DE USO DEL SISTEMA</i> .....	49
FIGURA 10. <i>DIAGRAMA DE CLASES DE DISEÑO</i> .....	69
FIGURA 11. <i>DIAGRAMA DE CLASES PERSISTENTES.</i> .....	78
FIGURA 12. <i>MODELO DE DATOS.</i> .....	79
FIGURA 13. <i>MODELO DE DATOS. (CONT.)</i> .....	80
FIGURA 14. <i>IMAGEN EN LA PARTE SUPERIOR DE TODAS LAS PÁGINAS</i> .....	81
FIGURA 15. <i>USO DEL COLOR ROJO PARA MENSAJES DE ERROR</i> .....	82
FIGURA 16. <i>EJEMPLO DE REPORTE</i> .....	83
FIGURA 17. <i>VERIFICACIÓN DE CAMPOS REQUERIDOS Y EN FORMATO CORRECTO</i> .....	84
FIGURA 18. <i>EJEMPLO DE CODIFICACIÓN UTILIZADA</i> .....	85
FIGURA 19. <i>DIAGRAMA DE DESPLIEGUE</i> .....	86
FIGURA 20. <i>DIAGRAMA DE COMPONENTES</i> .....	87
FIGURA 21. <i>DIAGRAMA DE COMPONENTES. (CONT.)</i> .....	88
FIGURA 22. <i>DIAGRAMA DE COMPONENTES. (CONT.)</i> .....	89

## INTRODUCCIÓN

La Universidad de Ciencias Informáticas recibe una gran cantidad de accesos diarios, tanto de trabajadores como visitantes, esperados y no esperados.

Actualmente, gestionar la entrada de todas estas personas es una tarea engorrosa. En la Universidad se requiere llevar un control estricto de la seguridad y protección física. El control de visitantes se realiza de forma manual, por no existir un sistema capaz de gestionar todo el flujo de información sobre accesos y visitas al centro eficientemente, trayendo como consecuencia que exista la siguiente **situación problemática**: los accesos no son controlados como se debe, propiciando que entren personas no autorizadas al centro, por otra parte, las solicitudes de visitas llevan una larga cadena de acciones, pasando desde el Solicitante, por el Directivo de su área, hasta el Técnico de Seguridad y Protección que se encuentra ubicado en la garita, todo este proceso se realiza manualmente. Como el registro de la visita es manual, se corre el riesgo de perder o extraviar cualquier documento importante. Además se crea una cola de personas cuyo acceso a la UCI esta siendo gestionado en los puntos de entrada en ese momento.

Por todo esto, se hizo necesario desarrollar un sistema que permitiera automatizar todas estas tareas garantizando la seguridad de la información y eliminando el engorroso trabajo manual que hasta ahora era necesario. Así, surge el Sistema de Control de Acceso.

Para la implementación del sistema se escogió el desarrollo de la aplicación con una interfaz Web para que los usuarios tuvieran acceso a la versión más reciente del mismo, además de permitir una experiencia agradable mediante una interfaz amigable. El sistema no sólo ofrece servicios a través de una interfaz gráfica, sino que puede y será utilizado por otras aplicaciones de la intranet que así lo requieran, mediante el uso de WebServices.

Con este trabajo se espera automatizar muchos de los procesos que hasta ahora se hacían de forma manual, aumentando la confiabilidad y seguridad de los datos, proporcionando facilidades a los usuarios del sistema a través de una interfaz amigable y sencilla y a su vez reducir los costos económicos que existen, en materiales de oficina. El principal beneficio del sistema, sin embargo, será que se incrementará el control de la seguridad interna, al tener una herramienta confiable para determinar quienes entran a la UCI en determinado momento, posibilitando que aumente la seguridad de los recursos materiales y de las personas en el campus universitario.

Se les controla el acceso a los trabajadores, registrándose el horario de entrada y salida, verificando además si tienen permisos para salir con algún medio, si se da el caso. Para las solicitudes de visitas, los usuarios que realizan las mismas son las personas que pertenecen de una forma u otra a la UCI (estudiantes, profesores, dirigentes, trabajadores del centro) y los que realizan el control de las visitas realizadas son los Técnicos de Seguridad Protección Física, mientras que los encargados de autorizar las solicitudes son los directivos, por ejemplo, el Vicerrector de Extensión Universitaria, Vicedecanos, Decanos, etc. Los mismos pueden delegar estas responsabilidades a otras personas.

Para el control del acceso se parte de la identificación de la persona mediante su solapín y el código de barras impreso al dorso de este, la manera de leer el código puede variar, en este trabajo se proponen distintas variantes para seleccionar el tipo de lector adecuado a cada caso. Una vez que se introduce el número del solapín, el sistema verifica la validez de las credenciales y que no ocurran salidas o entradas consecutivas de una misma credencial (acción conocida como *passback*). Por parte del control de visitantes, el sistema esta concebido para que los trabajadores, estudiantes y otras personas que pertenezcan a la UCI soliciten visitas de personas externas a la Universidad. Para esto, el usuario hace una solicitud suministrando los datos de la visita, como fecha y hora, quién viene, qué lugar será el visitado, si trae un medio de transporte o de otro tipo, etc. Esta solicitud se le refleja entonces al directivo del área a donde pertenece el solicitante, este directivo es el encargado de aprobarla o denegarla, sea cual sea la acción, el sistema permite notificar al solicitante de la decisión tomada. El solicitante además puede ver sus solicitudes, cambiar algún dato, eliminarlas, etc. Una vez aprobada una solicitud, todo esta listo para que el visitante efectúe la visita,

cuando llega, este muestra su carné de identidad y el Técnico de Seguridad y Protección consulta el sistema para verificar si existe una solicitud aprobada para esta persona, de ser así, se registra la entrada, se le entrega un pase que deberá devolver a la salida, y entra a la Universidad.

Para los visitantes inesperados, el Oficial de Seguridad, siguiendo el mecanismo correspondiente le informa al puesto de mando de la situación, se localiza la persona a visitar y se decide si se le da acceso o no. Si es permitido, se le toman sus datos y se procede del mismo modo que con las visitas esperadas.

El Sistema de Control de Acceso formará parte de toda una infraestructura organizativa para la gestión interna de la Universidad, donde existen sistemas como: el Sistema de Acreditación, Sistema de Reservación de Pase (para los estudiantes), Sistema del Comedor, entre otros.

Por tanto el objeto de estudio de este trabajo es todo lo referente al control de la seguridad y protección de la Universidad de las Ciencias Informáticas.

De aquí, se deriva que el campo de acción queda enmarcado específicamente en la parte de control de accesos y visitas a la Universidad.

Como Hipótesis se parte de la idea de que si se desarrolla una aplicación con interfaz Web, basada en un gestor de bases de datos potente, y en una plataforma poderosa como lo es .NET; es posible lograr la gestión de accesos y visitas a la UCI de una manera rápida, eficiente y segura, eliminando muchos problemas de seguridad y control presentes en la actualidad.

El **objetivo general** de este trabajo será: *Desarrollar una Aplicación con interfaz Web que permita el control de accesos y visitas a la UCI, además de brindar distintos servicios de reportes e información a usuarios y sistemas de la intranet universitaria.*

De aquí, se derivan los siguientes **objetivos específicos**:

- Hacer un estudio de todas las tecnologías disponibles en el mercado sobre Sistemas de identificación para elegir las más adecuadas a las condiciones de funcionamiento de la UCI.
- Confección de un sistema para el registro de las entradas y salidas de trabajadores al centro.
- Brindar un medio automatizado para que el personal interno de UCI solicite visitas de personas ajenas a la misma.
- Tributar información sobre las entradas y salidas de personas y transportes a la UCI, a otros sistemas de la intranet.
- Brindar información sobre los accesos a la UCI, mediante reportes en línea.
- Brindar información sobre las solicitudes de visitas registradas mediante reportes en línea.

Para cumplir los objetivos trazados, se desarrollaron las siguientes **tareas**:

- ✓ Realizar un estudio del entorno de trabajo.
- ✓ Identificar las necesidades de la institución.
- ✓ Declarar los requisitos que debe cumplir el sistema.
- ✓ Descripción de los procesos que se van a implementar en el sistema.
- ✓ Declaración de los ciclos de desarrollo.
- ✓ Especificación de los procesos que se van a implementar en el primer ciclo de desarrollo.
- ✓ Modelar conceptualmente las clases que están implicadas en el sistema.
- ✓ Desarrollar los diagramas de actividad.
- ✓ Desarrollar los diagramas que describen el diseño Web del sistema.
- ✓ Descripción de las clases del diseño.
- ✓ Diseño de la Base de Datos.
- ✓ Diseño de la interfaz
- ✓ Implementación de la aplicación

Se utiliza la metodología RUP para el desarrollo del sistema y UML como lenguaje de modelación para describir su estructura.



Este trabajo ha sido organizado de la siguiente manera:

**Capítulo 1:** Describe cómo se realiza el proceso de control de acceso en la Universidad actualmente. En él se mencionan los principales problemas que generaron la necesidad del cambio; como conclusión, se obtienen los objetivos generales y específicos a cumplir.

**Capítulo 2:** Trata la situación de las tecnologías a utilizar en el desarrollo de la aplicación, se comparan y seleccionan las mejores propuestas para el trabajo, y se explican los conceptos principales que se van a tratar.

**Capítulo 3:** Describe el negocio a través de un Modelo de Negocio, se hace el análisis del sistema a desarrollar. Se definen las funcionalidades del sistema y se describen detalladamente, utilizando herramientas de modelación, los principales procesos del mismo.

**Capítulo 4:** Trata sobre la elaboración de la solución mediante diagramas de clases, se plantean los principios para el diseño y la implementación. Aquí se desglosan y explican las funcionalidades que se definieron en el capítulo anterior.

**Capítulo 5:** Es un estudio de factibilidad de la construcción del sistema, observando los beneficios tangibles e intangibles y analizando los costos del desarrollo de la propuesta.



## FUNDAMENTACIÓN DEL TEMA

### 1.1 Introducción

En el presente capítulo se brinda una visión general de los aspectos relacionados con el Sistema de Control de Acceso. Se informa sobre la descripción de los principales conceptos asociados al problema que son necesarios para entender el negocio y la propuesta de solución. Serán objetos de comparación las tecnologías punteras del mercado en este sector, su incidencia y relación con otros sistemas.

Además se describen los procesos del negocio que se relacionan con el objeto de estudio de este trabajo. Se identifican los principales problemas que fundamentan la propuesta de solución, y se marcan los objetivos generales y específicos.

## 1.2 Qué es, y por qué es necesario el control de acceso

El Control de Acceso es el proceso de gestionar la entrada y salida de personas o medios de transporte a un lugar determinado, registrándose datos necesarios para garantizar el cumplimiento de una política de permisos que afiance la seguridad del lugar. El control de acceso es necesario porque es la primera barrera de defensa contra personas no deseadas, que pudieran causar algún tipo de problema dentro de la institución.

## 1.3 Sistemas de Identificación

Para registrar el acceso a cierto lugar, primero hay que definir un método de reconocimiento de identidad. Los sistemas de identificación permiten reconocer la identidad de una persona mediante algún dispositivo que ésta debe portar o mediante alguna acción que debe hacer para verificar su autenticidad. Existen muchos tipos de sistemas de identificación, las ventajas y desventajas de los más conocidos se expondrán más adelante. Algunos de estos sistemas son:

- ✓ Claves por Teclado
- ✓ Tarjetas de Banda Magnética
- ✓ Tarjetas de Código de Barras
- ✓ Touch Memory
- ✓ Tarjetas de Proximidad ó Radio Frecuencia (RF)
- ✓ Biométricos.

## 1.4 Objeto de Estudio

El objeto de estudio de este trabajo es la seguridad y protección en la Universidad de las Ciencias Informáticas (UCI).

La UCI es un centro de estudios de nuevo tipo. Fue creada en septiembre del 2002 en la Ciudad de la Habana como uno de los mayores proyectos de la Revolución. Su principal reto es la formación de miles de jóvenes de todo el país como futuros profesionales en la rama de la informática.

En ella se garantiza una preparación académica de gran calidad, así como el entrenamiento profesional de los estudiantes, por medio de la participación de forma directa y sistemática en la producción. Se brinda una gran flexibilidad en los diseños curriculares, como nueva alternativa para la formación y capacitación del capital humano vinculado a la informática.

Las perspectivas apuntan al papel de esta Universidad como decisivo en el desarrollo de la industria nacional del *software* y del programa de informatización. Se están dando grandes pasos en los sectores de la salud y educación, con la realización de proyectos productivos en los mismos. Para ello hacen uso de una moderna infraestructura tecnológica y método novedoso de organización, todo ello tributando a una calidad como la requerida en las normas internacionales.

## 1.5 Descripción del proceso de negocio actual

El control de acceso a la UCI que se realiza en los puntos de entrada a la Universidad, lo realiza un Agente de Seguridad y Protección (SEPCOM) sólo mirando si la persona es portadora de un solapín. Esto trae consigo que puedan entrar personas ajenas al centro sin chequear correctamente la validez del solapín, o simplemente portando otro solapín que no es el suyo. Los solapines están codificados por colores atendiendo al área a la que pertenece el portador y tienen la foto del mismo impresa (para los no eventuales), muchas veces no se mira si la foto es la correspondiente al que lo trae. Para el acceso de los visitantes, se emiten documentos con las personas esperadas, y se envían a la garita. En caso de que los visitantes lleguen de improviso, se trata de localizar a la persona a la que vienen a ver, por teléfono, para que gestione su entrada

con algún directivo. Antes de entrar se le recogen sus datos personales, anotando si entran con algún medio o paquete. Si el visitante viene en transporte, se recogen los datos del chofer y otros datos del vehículo.

### 1.6 Situación Problémica

En la UCI existe una gran cantidad de visitas y accesos de personas que necesitan ser controladas para que no pase personal no autorizado, actualmente existe ineficiencia para gestionar los permisos de quién puede y quién no puede entrar.

En la actualidad, el control de acceso lo realiza un Agente de Seguridad y Protección, sólo mirando el solapín de los portadores, esta revisión está lejos de ser minuciosa, ya que no se verifica la fecha de vencimiento del solapín, y muchas veces ni se mira bien la fotografía impresa, esto no es intencional, frecuentemente, hay una larga cola de autobuses esperando para entrar o salir y el proceso debe ser rápido, pues los trabajadores deben cumplir con el horario de la Universidad. Para salir, los trabajadores enseñan de igual modo su solapín, y si traen medios, tienen que mostrar un permiso emitido por el Jefe de Seguridad y Protección de la Universidad, este permiso es un documento que se deteriora y en ocasiones hay problemas por este motivo. Los solapines pueden perderse y ser usados de esta forma por personas no autorizadas. Para el control de los visitantes, cuando llega una persona a la garita, se busca si hay alguna solicitud de visita aprobada por algún directivo, si no, se localiza en ese momento un responsable dentro de la Universidad que lo autorice a entrar, esto representa pérdida de tiempo, y propicia a que se formen colas en la entrada. Las solicitudes pueden perderse o deteriorarse, y una vez que ésta se ha enviado en a la garita, resulta trabajoso hacer algún cambio en la misma por parte de los directivos.

Como no se cuenta con un sistema que registre las entradas y salidas de los trabajadores, resulta imposible saber a ciencia cierta cuantas personas y vehículos han entrado de la Universidad en un momento determinado, propiciando a que haya cierto descontrol para otros procesos que dependen de este.

### 1.7 Sistemas automatizados existentes vinculados al campo de acción

La Universidad ya cuenta con ciertos sistemas en su intranet que brindan información para poder construir otros sistemas que se nutran de ésta, así, podemos señalar los más importantes:

- **Sistema de Acreditación.**
  - ✓ Tiene almacenado los datos de todas las personas, trabajadoras o estudiantes de la UCI.
  - ✓ Tiene asociado a cada persona, un código único.
  - ✓ Brinda un servicio de certificación de Identidad, a otros sistemas informáticos
  
- **Sistema de Control de Acceso al Comedor.**
  - ✓ Controla el acceso a los comedores, gracias al código de barras impreso al dorso de los solapines.
  - ✓ Se controla el acceso diariamente, a más de 6000 personas, en las tres sesiones de servicios de los comedores.
  
- **Akademos:** Contiene la información de los estudiantes del centro.

El Sistema de Control de Acceso tributará información a otros sistemas de la intranet, el objetivo principal es utilizar esta información para saber, en determinado momento, cuantas personas ajenas al centro están dentro del mismo, qué medios de transporte tienen, etc. y de ahí derivar otros muchos usos de esta valiosa información.

## 1.8 Soluciones Existentes

Luego de realizar una búsqueda de información sobre otros sistemas de control de accesos y visitas, se llegó a la conclusión de que en el mundo existen numerosos de estos sistemas. La mayoría son muy caros o no se adaptan fácilmente a las condiciones de la Universidad.

Los más conocidos son:

- API-VISITAS
- JAS WIN Software de Control de Acceso

## 1.9 Propuesta de Solución

Después de realizar un análisis sobre cual es la situación actual sobre el objeto de estudio que tiene este trabajo, se concluye que se hace necesario implementar un sistema que pueda garantizar el control de accesos y visitas a la UCI.

Existen muchos sistemas de control de acceso, la mayoría caros, y que no se ajustan a los requerimientos y política interna de la Universidad, por esto se ha decidido efectuar la confección de dicho sistema, de forma que garantice la seguridad, confiabilidad y disponibilidad de los datos en tiempo real para la toma de decisiones.

La aplicación se integrará con sistemas existentes en la Intranet universitaria, utilizará el sistema de acreditación como base para verificar la información de las personas pertenecientes a la UCI, brindándole así, a los agentes de seguridad, un mecanismo confiable y rápido para actuar acorde a las distintas situaciones de seguridad que puedan presentarse.

## 1.10 Fundamentación de los objetivos que se propone el trabajo.

Para darle respuesta a la situación problemática planteada, se propone para este trabajo un conjunto de objetivos:

### **1.10.1 Objetivo General**

El objetivo es brindar una propuesta de solución general integral, para controlar el acceso a la Universidad, confeccionando un sistema único que permita la automatización de dicho proceso, el mismo debe permitir registrar accesos de trabajadores, brindar la posibilidad a estudiantes y demás personas que vivan en la UCI, de solicitar visitas de personas externas al centro, brindar reportes en línea y por correo electrónico a directivos.

### **1.10.2 Objetivos Específicos**

Este trabajo tiene los siguientes objetivos específicos:

- Hacer un estudio sobre la tecnología necesaria y factible a implementar en los puntos de control de acceso a la Universidad.
- Confección de un sistema único de control de acceso que permita:
  - Registrar entradas y salidas de personas pertenecientes a la UCI
  - Registrar visitantes externos
  - Gestionar solicitudes de visitas
  - Brindar reportes en línea sobre accesos, a directivos.
  - Enviar reportes a directivos, mediante el correo electrónico.

## 1.11 Conclusiones

En este capítulo se detallaron las condiciones y problemas que rodean al objeto de estudio; y a través de los conceptos y definiciones planteadas, se determinaron las condiciones específicas que rodean al problema y en base a esto se obtuvieron los objetivos generales y específicos para este trabajo.





## **TENDENCIAS Y TECNOLOGÍAS ACTUALES A CONSIDERAR**

### 2.1 Introducción

En el presente capítulo, se hace un análisis de cómo se encuentran en la arena internacional las tecnologías que pueden ser adecuadas para llevar a cabo el sistema que se pretende desarrollar. En algunos casos se hacen comparaciones que fundamentarán la propuesta final.

## 2.2 Fundamentación de las tecnologías en las que se basa la propuesta

Una variada gama de tecnologías de hardware y software posibilita brindar una solución efectiva tanto en las grandes empresas que requieren máxima seguridad, robustez y flexibilidad de programación, como en pequeños comercios que necesitan precios económicos y facilidad de uso. Un software de control de acceso, con sus accesorios, permite llevar el control de horarios a lugares en los cuales antes no era posible por cuestiones de costo u operativas (oficinas remotas, obras, personal móvil e inspectores, etc.) [1]

El reto, entonces, es encontrar esa solución que garantice una relación costo/beneficio y que requiera la menor cantidad de esfuerzo en su implantación y uso.

## 2.3 Tecnologías de hardware utilizadas para el control del acceso

En primer lugar es necesario conocer cuáles son los diferentes tipos de lectores y tarjetas, analizando las ventajas y desventajas de cada uno.

### **2.3.1 Claves por Teclado**

Realmente esta opción es la más económica, pero la menos segura. Hace tiempo que han caído en desuso y no se han generado hasta el momento nuevas aplicaciones donde puedan resurgir como una opción válida.

### **2.3.2 Tarjetas de Banda Magnética**

Es la tecnología más conocida y difundida, ya que se utiliza en todos los sistemas de tarjetas de crédito y compra (de hecho se pueden utilizar esas mismas tarjetas en muchos sistemas de control de acceso). Su ventaja es su difusión, popularidad y el bajo costo, pero en sí, es, de todos los medios de identificación, el más vulnerable. La banda magnética de la tarjeta, debe ser tratada con cierto cuidado para evitar

que se raye o sea expuesta a campos magnéticos que la borren, por tal motivo, no son recomendables para usar en ambientes industriales. Sólo se recomiendan en oficinas o establecimiento administrativos.

### **2.3.3 Tarjetas de Código de Barras**

“El código de barras es una tecnología de identificación automática. Permite recolectar datos con precisión y rapidez.

Un código de barras consiste en una serie de barras adyacentes paralelas y espacios. Los diseños predeterminados de anchura se utilizan para codificar datos en el código. Para leer información en un símbolo de código de barras, un dispositivo de lectura, tal como un lápiz óptico, se desliza a través del símbolo de un lado al otro, la anchura de barras y los espacios son analizados por el decodificador del lector, y los datos originales se recuperan.

La aplicación más visible en esta tecnología es en la industria de supermercado, donde ha estado en el uso desde 1970.” [6]

### **2.3.4 Touch Memories**

El elemento en sí es una pastilla electrónica, encapsulada en acero inoxidable de unos 16 mm. de diámetro, que se transporta con un soporte plástico de unos 5 cm. de largo con un ojalillo en su parte superior para poder colgarlo en un llavero.

Comúnmente se los denomina llave electrónica y brindan un alto nivel de seguridad, ya que son altamente resistentes al desgaste, siendo ideales para ambientes industriales en donde la probabilidad de falla, vandalismo o sabotaje sea alta, aunque no son recomendables para ambientes con alto grado de generación de corriente estática (Ej.: oficinas con mucha alfombra y ambientes muy secos). Su tecnología de avanzada evita la posibilidad de duplicarlas. En precio hay que tener en cuenta que son unos de los

medios más caros, sin embargo, nunca se desgastan, como puede suceder con una tarjeta magnética, ya que en lo que al lector respecta, es también de acero. [1]

### **2.3.5 Tarjetas de Proximidad o Radio Frecuencia (RF)**

“Un sistema de RF consta de dos partes; 1) La unidad lectora y 2) El tag, transponder o tarjeta. Cuando la tarjeta posee alimentación interna, se la denomina proximidad activa y cuando la tarjeta no tiene esa batería interna se la denomina proximidad pasiva. Mayormente son utilizados los sistemas de proximidad pasiva. El lector emite una onda electromagnética que genera una corriente interna en la tarjeta, sirviendo para su alimentación. Dentro de la tarjeta existe un microchip que posee: una memoria que guarda el código de la misma; una antena que es la encargada de recibir y transmitir.

Al inducirse esa corriente en la tarjeta, la misma extrae de la memoria el dato codificado y lo devuelve, enviándolo al lector. El lector lo recibe, lo decodifica, lo filtra, lo amplía y lo envía a la unidad de control de accesos a la que esté conectada.

Es una tarjeta que por su diseño tecnológico, es prácticamente imposible que pueda duplicarse. Hoy en día es una de las tecnologías más modernas y efectivas, por ser práctica y de bajo costo de mantenimiento.” [1]

### **2.3.6 Sistemas Biométricos**

Su funcionamiento se basa en la lectura o reconocimiento de alguna parte del cuerpo humano; de la huella dactilar, geometría de la mano, frecuencia de la voz, por la retina o reconocimiento facial; eliminando por completo el uso de las tarjetas.

Los más conocidos son los lectores de huellas digitales, geometría de la mano e iris del ojo. Sus desventajas son la velocidad y el precio, además, deben ir acompañados de un teclado (para anteponer un código para acelerar el proceso de búsqueda), y por último la poca posibilidad de ser autónomos (generalmente por su complicada lógica se ven obligados a trabajar con un software de análisis y una PC conectada directa al lector, lo cual es poco práctico y más caro), pero seguramente con el tiempo se irán superando

estas dificultades y en un futuro no lejano, llegarán a ser una opción más asequible en el mercado.

## 2.4 Tecnología utilizada en la UCI: El código de barras

La tecnología de código de barras es muy económica, y de fácil implementación, ya que con una impresora de calidad pueden ser impresos los carnés de identificación con códigos de barras. Existen varios tipos de codificación, actualmente, estos son los más importantes:

- Código 39
- Código 39 ASCII Total
- Codabar
- Intercalado 2 de 5
- Código 128
- UPC (Código Universal de Producto)
- EAN (European Article Numbering o Sistema de Numeración Europeo)

En la UCI se utiliza el **Código 39**, es el más utilizado porque codifica números, letras mayúsculas, y algunas marcas de puntuación (mayúsculas A-Z, Números 0-9, "espacios" y símbolos: -, +, /, \$, ., %). El Código 39 puede ser variable en longitud, permitiendo hacer códigos de cualquier número de dígitos. Este formato ha pasado a ser la norma para el gobierno, industria, educación y aplicaciones de negocios.



Figura 1. Código 39.

### 2.4.1 Beneficios del código de barras

Los beneficios de utilizar códigos de barras como sistema de recolección de datos automática son muy sencillos: la velocidad y exactitud. Una y otra vez se ha probado que capturar datos con códigos de barras es por lo menos 100 veces más rápido y más exacto que la captura normal por teclado, que se traduce en un aumento dramático en la eficiencia y productividad para cualquier operación.[6]

### 2.4.2 Tipos de lectores de códigos de barras

En los sistemas de control de acceso actuales los códigos de barras pueden ser leídos de muchas formas utilizando diferentes dispositivos. Generalmente un sistema de lectura se compone de dos partes: una interfaz, llamada por lo regular "decodificador", y lo que se conoce por el término de "dispositivo de entrada". Existen básicamente tres tipos de lectores:

#### 2.4.2.1 Sistemas en Línea

Estos sistemas están conectados de una manera semi-permanente a una computadora o terminal. Su función principal es leer un código de barras, interpretarlo (decodificarlo) y transmitirlo inmediatamente. Por lo general, los sistemas En Línea toman dos formas: conexión directa a teclado o "wedge", conexión serial y tarjeta de expansión. La conexión "wedge" se utiliza principalmente en microcomputadoras (IBM PC, Apple y compatibles), así como en terminales de sistemas de cómputo de IBM. El "wedge" provee una conexión directa entre el lector de código de barras y el teclado de la PC o terminal.

Los sistemas *En Línea* son programables, por lo general leyendo comandos codificados en código de barras impresos en el manual que se incluye con estos equipos. Todos los parámetros se almacenan en la memoria no volátil del decodificador, y la única manera de cambiarlos es leyendo nuevos comandos que modifiquen la configuración existente. ¿Por qué razón se querría programar un decodificador? Bien, se puede requerir que se envíe un "enter" automáticamente después de cada lectura, o bien que se transmitan

códigos de teclados de computadoras tipo XT en lugar de AT. Con ciertos equipos, actualmente es posible editar, separar o filtrar la información del código de barras antes de enviarlo al sistema de cómputo.

#### 2.4.2.2 Sistemas Fuera de Línea

“Un Lector Portátil de código de barras es una microcomputadora integrada, que incluye un display, teclado para captura datos a mano, capacidad para descifrar códigos de barras, puerto RS-232 que proporciona acceso para comunicaciones con la PC, baterías y memoria para el almacenamiento de programa y datos. Para recaudo informático de códigos de barras, el sistema puede equiparse con lectores tipo lápiz, CCD, o pistolas Láser.

El sistema provee los operadores con indicadores, con las cuales los operadores dan de alta datos por medio de las teclas o lectores ópticos. Los datos se graban, hasta el que la terminal portátil transfiera los datos (por medio de puerto de acceso RS-232) al PC, en donde los datos son grabados típicamente en los archivos delimitados DE ASCII, para la importación fácil de programas como Access, FoxPro, Excel, y otras aplicaciones.

Estas terminales en algunos casos contienen programas preinstalados (generalmente para el levantamiento de inventarios) o pueden ser escritos de acuerdo a las necesidades específicas de cada trabajo que desee realizar, para esto las terminales portátiles cuentan con software generadores de aplicaciones y muchas veces utilizan un lenguaje Basic o C para su programación.

Existen otras terminales que ya poseen el lector integrado, de decir un lector óptico tipo *pistola* integrado en la parte superior de la terminal. La ventaja de utilizar este tipo de terminal portátil es que el operador puede ejecutar la operación informática de recaudo con una sola mano (vs. dos manos cuando utiliza un terminal portátil sin láser integrado).” [6]

#### 2.4.2.3 Sistemas de Radiofrecuencia (RF)

Los lectores de radio frecuencia son como terminales en línea de la computadora central, pero sin cables. La comunicación permanente con la computadora posibilita el flujo de información en ambos sentidos: del lector al mainframe y viceversa. Esto hace posible, aparte de actualizar la información de la base de datos al instante, recibir instrucciones actualizadas sobre qué hacer, incluso cambios de última hora. Existen dos tipos básicos de lectores mediante RF: 1) Los que emulan terminales o PCs. 2) Lectores simples que se comunican a una computadora vía un puerto serial. [6]

Actualmente hay dos tecnologías que compiten en el mercado de RF: Banda Angosta ("*narrow band*") y Espectro Extendido ("*spread spectrum*"). La primera es la más tradicional, requiriendo el uso de una frecuencia exclusiva en el área geográfica de cobertura y una licencia gubernamental. Los sistemas más recientes utilizan Espectro Extendido, no requieren de licencia y se asemejan a la tecnología celular utilizada en la telefonía móvil. [1]



## 2.5 Tecnologías de software a conocer

### 2.5.1 Plataforma .NET

“El Framework .Net es una infraestructura sobre la que se reúne todo un conjunto de lenguajes y servicios que simplifican el desarrollo de aplicaciones. Mediante esta herramienta se ofrece un entorno de ejecución distribuido.

.Net Framework soporta múltiples lenguajes de programación y, es posible desarrollar cualquier tipo de aplicación con cualquiera de estos lenguajes: C# (C Sharp), Visual Basic, C++, J# (Java #), Jscript, además, se han adaptado lenguajes existentes, como Perl, Python, Fortran, Cobol, entre otros.” [7]

La idea central detrás de la plataforma .NET es la de servicio. Más concretamente **software como servicio** y de cómo construir, instalar, consumir, estos servicios para que puedan ser accedidos mediante Internet. La plataforma .NET permite usar Internet y su capacidad de distribución para que los usuarios accedan desde cualquier dispositivo, en cualquier sistema operativo y lugar a la funcionalidad que los servicios Web proveen.

### 2.5.2 ASP.NET

ASP.NET es un marco de trabajo de programación generado en Common Language Runtime que puede utilizarse en un servidor para generar eficaces aplicaciones Web. ASP.NET ofrece varias ventajas importantes acerca de los modelos de programación Web anteriores:

- **Mejor rendimiento.**
- **Compatibilidad con herramientas de primer nivel.**
- **Eficacia y flexibilidad.**
- **Simplicidad.**

- **Facilidad de uso.**
- **Escalabilidad y disponibilidad.**
- **Posibilidad de personalización y extensibilidad.**
- **Seguridad.**

### 2.5.3 Lenguaje C#

C# es un lenguaje de programación poderoso, diseñado para escribir aplicaciones empresariales. Es una evolución de los lenguajes C y C++. Utiliza muchas de las características de C++ en las áreas de instrucciones, expresiones y operadores.

C# presenta considerables mejoras e innovaciones en áreas como seguridad de tipos, control de versiones, eventos y recolección de elementos no utilizados (liberación de memoria).

El lenguaje proporciona la capacidad de generar componentes de sistema duraderos en virtud de las siguientes características:

- Total compatibilidad entre COM y plataforma para integración de código existente.
- Gran robustez, gracias a la recolección de elementos no utilizados (liberación de memoria) y a la seguridad en el tratamiento de tipos.
- Seguridad implementada por medio de mecanismos de confianza intrínsecos del código.
- Plena compatibilidad con conceptos de metadatos extensibles.

Además, es posible interactuar con otros lenguajes, entre plataformas distintas, y con datos heredados, en virtud de las siguientes características:

- Plena interoperabilidad por medio de los servicios de COM+ 1.0 y .NET Framework con un acceso limitado basado en bibliotecas.
- Compatibilidad con XML para interacción con componentes basados en tecnología Web.

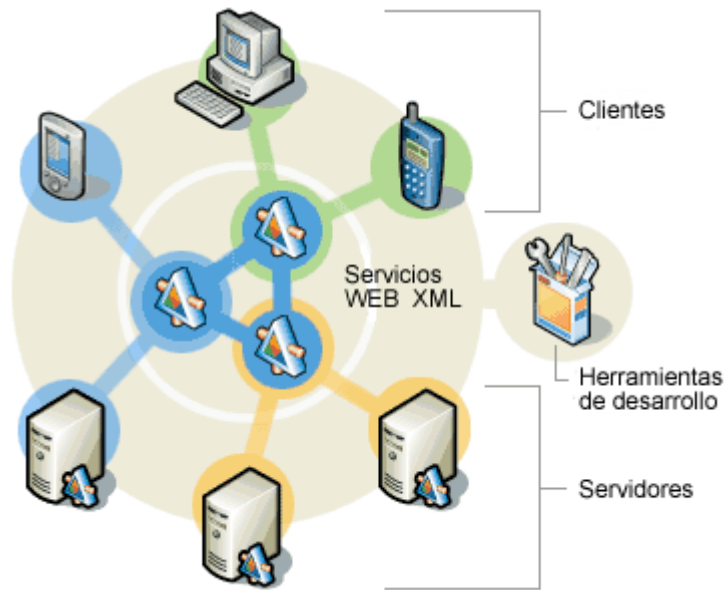
- Capacidad de control de versiones para facilitar la administración y la implementación.

#### **2.5.4 Web Services.**

Los servicios Web son la revolución informática de la nueva generación de aplicaciones que trabajan colaborativamente en las cuales el software esta distribuido en diferentes servidores.

Los servicios Web XML permiten que las aplicaciones trabajen en conjunto, haciendo uso de de funcionalidades brindadas por otras aplicaciones independientemente de cómo se hayan creado, cuál sea el sistema operativo o la plataforma en que se ejecutan y cuáles los dispositivos utilizados para obtener acceso a ellas. Aunque los servicios Web XML son independientes entre sí, pueden vincularse y formar un grupo de colaboración para realizar una tarea determinada.

Los servicios XML WebServices son los elementos fundamentales en la evolución hacia la computación distribuida a través de Internet. Se están convirtiendo en la plataforma de integración de aplicaciones gracias a los estándares abiertos y al énfasis en la comunicación y colaboración entre personas y aplicaciones. Las aplicaciones se crean utilizando los servicios XML WebServices múltiples de origen distinto que funcionan conjuntamente, sin importar su ubicación o la forma en que se implementaron.



**Figura 2.** Los servicios Web pueden conectar a diversos tipos de aplicaciones.

En la UCI todas las aplicaciones que se utilizan en la intranet ofrecen o “consumen” servicios Web de otras, es decir, existe una interrelación entre los sistemas de la red para lograr la reutilización y la funcionalidad de estas.

### 2.5.5 SQL

“Debido a la diversidad de lenguajes y de bases de datos existentes, la manera de comunicar entre unos y otras sería realmente complicada de gestionar a no ser por la existencia de estándares que nos permiten el realizar las operaciones básicas de una forma universal.

Es de eso de lo que trata el SQL (Structured Query Language) que no es más que un lenguaje estándar de comunicación con bases de datos. Hablamos por tanto de un lenguaje normalizado que nos permite trabajar con cualquier tipo de lenguaje en combinación con cualquier tipo de base de datos (MS Access, SQL Server, MySQL, etc.).

El hecho de que sea estándar no quiere decir que sea idéntico para cada base de datos. En efecto, determinadas bases de datos implementan funciones específicas.“ [2]

## 2.6 Fundamentación del Gestor de Bases de Datos utilizado

Los principales objetivos de un Gestor de Base de Datos (SGBD) son: Evitar la redundancia de los datos, eliminando así la inconsistencia de los mismos, mejorar los mecanismos de seguridad de los datos y la privacidad. Podemos distinguir cuatro tipos de contextos para usar mecanismos de seguridad: seguridad contra accesos indebidos a los datos, seguridad contra accesos no autorizados a la BD, seguridad contra destrucción causada por el entorno (fuego, inundación, robo,...), seguridad contra fallos del propio sistema (fallos del hardware, del software, etc). Debido a la naturaleza de los datos que se almacenarán en el sistema propuesto, se ha decidido utilizar fue Microsoft SQL Server 2000 como gestor de base de datos, por ser una aplicación poderosa, robusta, que permite gran seguridad de los datos, ostenta marcas de referencia en cuanto a escalabilidad y confiabilidad, que son críticas para el éxito de bases de datos de gran tamaño. El SQL Server permite lograr una gran velocidad en el procesamiento de transacciones, y agilidad en todas sus operaciones, además, es el utilizado por la mayoría de aplicaciones en la Universidad.

## 2.7 Fundamentación de la metodología a utilizar

### **2.7.1 RUP (Rational Unified Process)**

Para desarrollar un software se necesita una forma ordenada de trabajo. Un proceso que integre las múltiples facetas del desarrollo. Se necesita un método común, un proceso que:

1. Proporcione una guía para ordenar las actividades de un equipo.
2. Dirija las tareas de cada desarrollador por separado y del equipo como un todo
3. Especifique los artefactos que deben desarrollarse
4. Ofrezca criterios para el control y la medición de los productos y actividades de proyectos

“El Proceso Unificado es un proceso de desarrollo de Software. Un proceso de desarrollo de software es el conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema software. Sin embargo, el Proceso Unificado es más que un simple proceso; es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas software, para diferentes áreas de aplicación, diferentes tipos de organización, diferentes niveles de aptitud y diferentes tamaños de proyecto...El Proceso Unificado está *basado en componentes*, lo cual quiere decir que el sistema software en construcción está formado por componentes software interconectados a través de interfaces bien definidas.” [3]

“El Proceso Unificado utiliza el *Lenguaje Unificado de Modelado* (Unified Modeling Language, UML) para preparar todos esquemas de un sistema software , De hecho, UML, es una parte esencial del Proceso Unificado – sus desarrollos fueron paralelos”. [3]

No obstante, los verdaderos aspectos definitorios del Proceso Unificado se resumen en tres fases claves – *dirigido por caso de uso, centrado en la arquitectura e iterativo e incremental*. Esto es lo que hace único al Proceso Unificado.

### 2.7.2 UML (Unified Modeling Language)

“UML son las siglas de Unified Modeling Language (Lenguaje Unificado de Modelado), notación (esquemática en su mayor parte) con que se construyen sistemas por medio de conceptos orientados a objetos”. [3].

El Lenguaje Unificado de Modelado (UML - Unified Modeling Language) es un lenguaje gráfico para visualizar, especificar y documentar cada una de las partes que comprende el desarrollo de software. UML permite una forma de modelar cosas conceptuales como lo son procesos de negocio y funciones de sistema, además de cosas concretas como lo son escribir clases en un lenguaje determinado, esquemas de base de datos y componentes de software reusables.

UML permite modelar, construir y documentar los elementos que forman un sistema software orientado a objetos. Se ha convertido en el estándar *de facto* de la industria, debido a que ha sido impulsado por los autores de los tres métodos más usados de

orientación a objetos: Grady Booch, Ivar Jacobson y Jim Rumbaugh. Tiene como objetivo brindar un material de apoyo que le permita al lector poder definir diagramas propios como también entender diagramas ya existentes.

El Lenguaje Unificado de Modelado prescribe un conjunto de notaciones y diagramas estándar para modelar sistemas orientados a objetos, y describe la semántica esencial de lo que estos diagramas y símbolos significan. Mientras que ha habido muchas notaciones y métodos usados para el diseño orientado a objetos, ahora los modeladores sólo tienen que aprender una única notación.

UML se puede usar para modelar distintos tipos de sistemas: sistemas de software, sistemas de hardware, y organizaciones del mundo real.

## 2.8 Herramientas Utilizadas

Para la confección de esta aplicación se han utilizado numerosas herramientas, para el diseño Web, se utilizó **Macromedia Dreamweaver MX 2004** por su gran flexibilidad y comodidad para hacer diseños complicados, y **Visual Studio .NET**, por ser la herramienta más poderosa, y completa para programar cualquier aplicación en .NET, el Visual Studio .NET además permite, mediante un único IDE, la elaboración de aplicaciones Web, bases de datos, servicios Web, hojas de estilo, y otros tipos de componentes. Para modelar la aplicación se utilizó **Rational Rose**, herramienta líder para este propósito, así como **XDE**, plug-in de Rational Rose para Visual Studio.

### **Otras Herramientas utilizadas**

- *Embarcadero Studio* para la modelación de la Base de Datos.

Esta aplicación permite modelar conceptualmente la Base datos y establecer relaciones entre sus entidades, la herramienta luego genera el código necesario del modelo físico, ahorrando tiempo y minimizando las posibilidades de errores.

- *Adobe Photoshop CS* para el tratamiento de los gráficos.

Adobe Photoshop es un herramienta muy poderosa para crear cualquier tipo de gráficos, su integración con Adobe ImageReady hacen que crear complicados gráficos para la Web sea una tarea muy fácil.

## 2.9 La propuesta.

Tomando como base los elementos antes expuestos de algunas de las herramientas y tecnologías disponibles actualmente, se decidió utilizar ASP.NET y Visual Studio .NET para el desarrollo de una aplicación que permita aprovechar al máximo las ventajas de la plataforma .NET.

Se utilizó SQL Server como Gestor de Base de Datos por su fortaleza, capacidad para grandes volúmenes de información, y estar disponible en los servidores de la Universidad.

Es importante señalar que aunque el Visual Studio .NET no es gratis, si lo son la Plataforma .NET y las versiones **Express** de Visual Studio y SQL Server. Las cuales permiten el desarrollo de aplicaciones Web robustas en .NET bajo ningún costo adicional. [8]

La implantación en la UCI de tecnología de código abierto, eventualmente, obligará a las aplicaciones existentes en la intranet construidas en .NET a migrar hacia plataformas como Mono, la versión de código abierto de .Net. La solución aquí planteada puede ser migrada a esta plataforma de una manera segura, ya que no se han utilizado ninguna característica no estándar de la misma.



### 2.10 Conclusiones.

En este capítulo se profundizó en el conocimiento de algunos conceptos necesarios para la comprensión de este trabajo. Además se realizó un análisis completo de las tecnologías que serán utilizadas a lo largo del desarrollo del sistema propuesto, se fundamentaron las elecciones de lenguaje, sistema gestor de bases de datos, y la metodología a utilizar. Una vez conocidas las herramientas óptimas, y los conceptos a utilizar, se puede empezar a desarrollar la propuesta de sistema.



## DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA

### 3.1 Introducción.

En el presente capítulo se hace la descripción de la propuesta que trae este trabajo, para ello se describen los procesos del negocio que tiene que ver con el objeto de estudio. Es importante conocer cómo funcionan estos procesos, ya que ayuda a tener una concepción más clara del funcionamiento de la Universidad para elaborar una aplicación que se ajuste mejor a las reglas de la misma.

Además, se enumeran los requisitos funcionales y no funcionales que debe tener el sistema que proponemos, lo que permite hacer una concepción general del sistema, e identificar mediante un Diagrama de Casos de Uso, las relaciones de los actores que interactúan con el sistema, y las secuencias de acciones con las que interactúan.

### 3.2 Reglas de negocio.

Las reglas de negocio describen políticas que deben cumplirse o condiciones que deben satisfacerse, por lo que regulan algún aspecto del negocio. Primeramente, se explicarán algunos conceptos asociados al modelo del negocio:

**Trabajador:** Toda persona perteneciente a la UCI que no es estudiante: profesores, personal de la cocina, médicos, personal de mantenimiento, etc.

**Área:** La UCI esta estructuralmente organizada en áreas jerárquicas. Cada área tiene su jefe o responsable.

**Solicitud de Visita:** Petición de entrada de un visitante por parte de un trabajador o estudiante a el director de su área.

**Pase:** Documento que recoge los datos de un visitante

**Medio:** Cualquier objeto o paquete que traiga un visitante o trabajador, por ejemplo: Laptops, equipos electrodomésticos, etc.

**Registro de Vehículos:** Registro que se lleva en la garita donde se recogen los datos de los transportes que traen los visitantes autorizados a entrar a la UCI.

**Registro de Visitantes:** Registro que se lleva en la garita con los visitantes que entran a la UCI.

Las solicitudes de visita de directivos no pasan por proceso de aprobación, estas ya están automáticamente aprobadas porque los directivos tienen potestad para esto. Ningún visitante puede entrar si no ha sido aprobado por un directivo. Ningún visitante o trabajador puede salir del centro con algún medio, a no ser que tenga un permiso para esto.

### 3.3 Actores y trabajadores del negocio

ACTORES DEL NEGOCIO	JUSTIFICACIÓN
Solicitante	Cualquier persona de la UCI que solicite visitas de personal ajeno a la Universidad
Visitante	Persona que no pertenece a la UCI, que viene a visitar a alguien. Puede ser esperado o presentarse de imprevisto.
Trabajador	Persona que trabaja en la UCI, tiene un solapín con sus datos personales, puede vivir dentro o fuera de la UCI

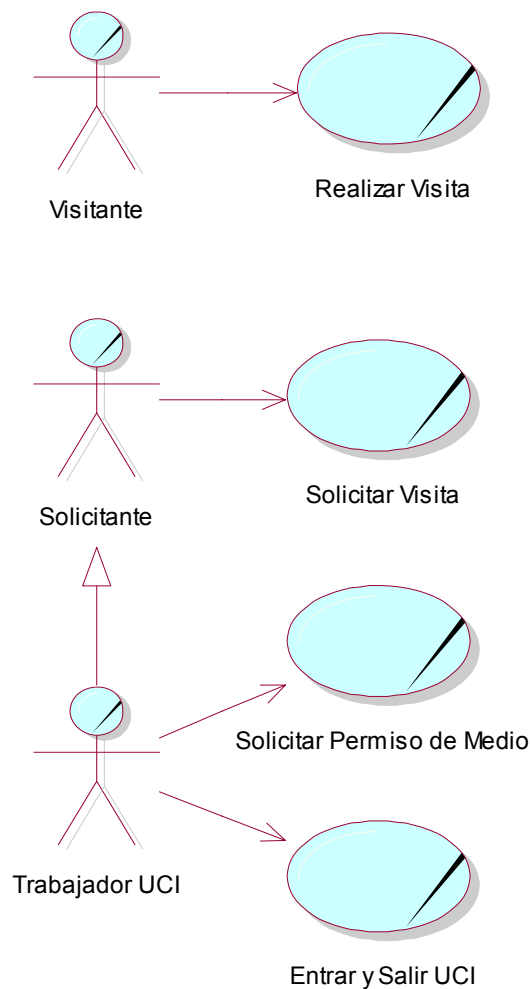
TRABAJADORES DEL NEGOCIO	JUSTIFICACIÓN
Directivo	Persona trabajadora de la UCI con cargos en la dirección de la Universidad, pertenece a determinada área, puede aprobar solicitudes de visita a su área.
Responsable de Seguridad	Agentes de seguridad de la UCI. Son los encargados de pedirles y verificar las credenciales de las personas que entren a la UCI.
Técnico de Seguridad y Protección	Responsable del control de los medios básicos y controla el trabajo de los Responsables de Seguridad, gestionan la autorización de las personas ajenas al centro, que deseen entrar, revisando si hay alguna solicitud de entrada previa o haciendo la gestión de su permiso.

### 3.4 Descripción de los Casos de Uso del Modelo del Negocio Actual

Los casos de uso le proporcionan a los analistas del sistema y clientes, llegar a un acuerdo sobre los requerimientos del mismo.

#### 3.4.1 Diagrama de Casos de Uso del modelo del negocio actual

Un diagrama de casos de uso del negocio representa gráficamente los procesos del negocio y su interacción con los actores del negocio.



**Figura 3.** Diagrama de casos de uso del negocio

### 3.4.2. Expansión de los Casos de Uso

Los casos de uso del negocio, representan procesos que ocurren en el negocio. A continuación se explican los detalles de los casos de uso para su comprensión por parte de miembros del equipo de desarrollo y clientes.

<b>Nombre del caso de uso:</b>	CU1 – Solicitar Visita	
<b>Actores del negocio:</b>	Solicitante, Directivo	
<b>Propósito:</b>	Solicitar la entrada a la UCI de personas ajenas a la misma.	
<b>Resumen:</b>	El caso de uso inicia cuando un solicitante va a ver a su director de área (directivo) para que le autorice la entrada a un Visitante. Este aprueba o no la entrada.	
<b>Casos de uso asociados:</b>		
<b>Flujo de trabajo</b>		
<b>Acción del actor</b>	<b>Respuesta del negocio</b>	
1 El solicitante informa al directivo los datos del Visitante.	2 El directivo analiza la solicitud y la aprueba. Luego, informa mediante un documento a la Dirección de Seguridad y Protección la fecha en que vendrá el visitante y suministra sus datos.	
3 El solicitante se retira. Fin del proceso.		
<b>Prioridad:</b>	Alta	
<b>Mejoras:</b>	La automatización de este proceso de evaluación reducirá el tiempo de respuesta y permitirá a estos directivos y solicitantes, mejorar su gestión, además, un solicitante podrá acceder a sus solicitudes para hacer cambios, si no ha sido aprobada. No habrá que ir físicamente a ver al directivo. El directivo no tendrá que llevar ningún documento a Seguridad y Protección.	
<b>Cursos alternos:</b>		
<b>Acción del actor</b>	<b>Respuesta del negocio</b>	
	2. El Directivo considera que no puede entrar el visitante, deniega la solicitud e informa al solicitante. (ir a paso 3).	

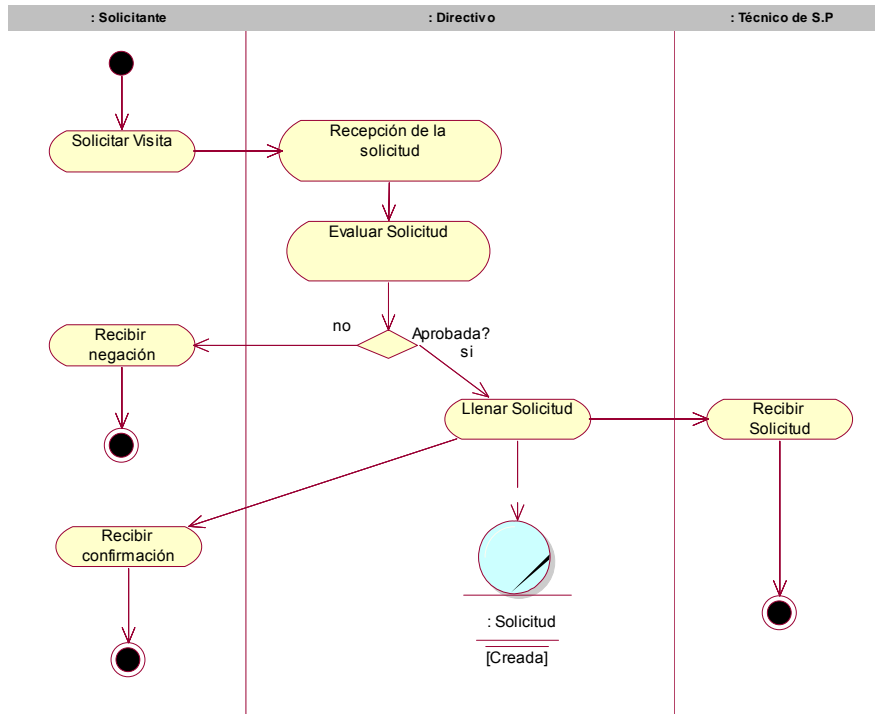


Figura 4. Diagrama de actividades del C.U. "Solicitar Visitas"

<b>Nombre del caso de uso:</b>	CU2 – Realizar Visita	
<b>Actores del negocio:</b>	Responsable Seguridad, Visitante, Técnico de Seguridad y Protección.	
<b>Propósito:</b>	Que un visitante entre a la UCI	
<p>Resumen: El caso de uso inicia cuando un visitante llega a la garita, para ver a alguien dentro de la UCI. Se gestiona si puede entrar, si es así, se le recogen sus datos y se le entrega un pase.</p>		
<b>Casos de uso asociados:</b>		
<b>Flujo de trabajo</b>		
<b>Acción del actor</b>	<b>Respuesta del negocio</b>	
1 El visitante llega a la garita, informando a quien desea ver.	2 El Técnico de S.P. busca su solicitud, le pide el carné de identidad, llena un pase con sus datos, especificando la hora de llegada, retiene el carné de identidad y le entrega el pase.	
3 El visitante recoge el pase, entra a la UCI.	5 El Técnico de S.P recoge el pase, verificando que no salga con ningun medio que no se haya registrado a su entrada, le devuelve su carné de indentidad. Y actualiza el pase.	
4 El Visitante se presenta en la garita para salir de la UCI. Entrega el pase.		
6 El Visitante se retira. Fin del Proceso.		
<b>Cursos Alternos:</b>		
<b>Acción del Actor:</b>	<b>Respuesta del Negocio:</b>	
	2 Si <b>no</b> hay una solicitud, se contacta a la persona que el Visitante viene a ver, si ésta acepta la visita, se gestiona la entrada con el Directivo correspondiente (Ir a paso 3)	
<b>Prioridad:</b>	Alta	
<b>Mejoras:</b>	La automatización de este proceso permitirá una mejora considerable en la velocidad de los trámites para recoger la información de los visitantes o verificar si ya había una solicitud aprobada para el mismo.	



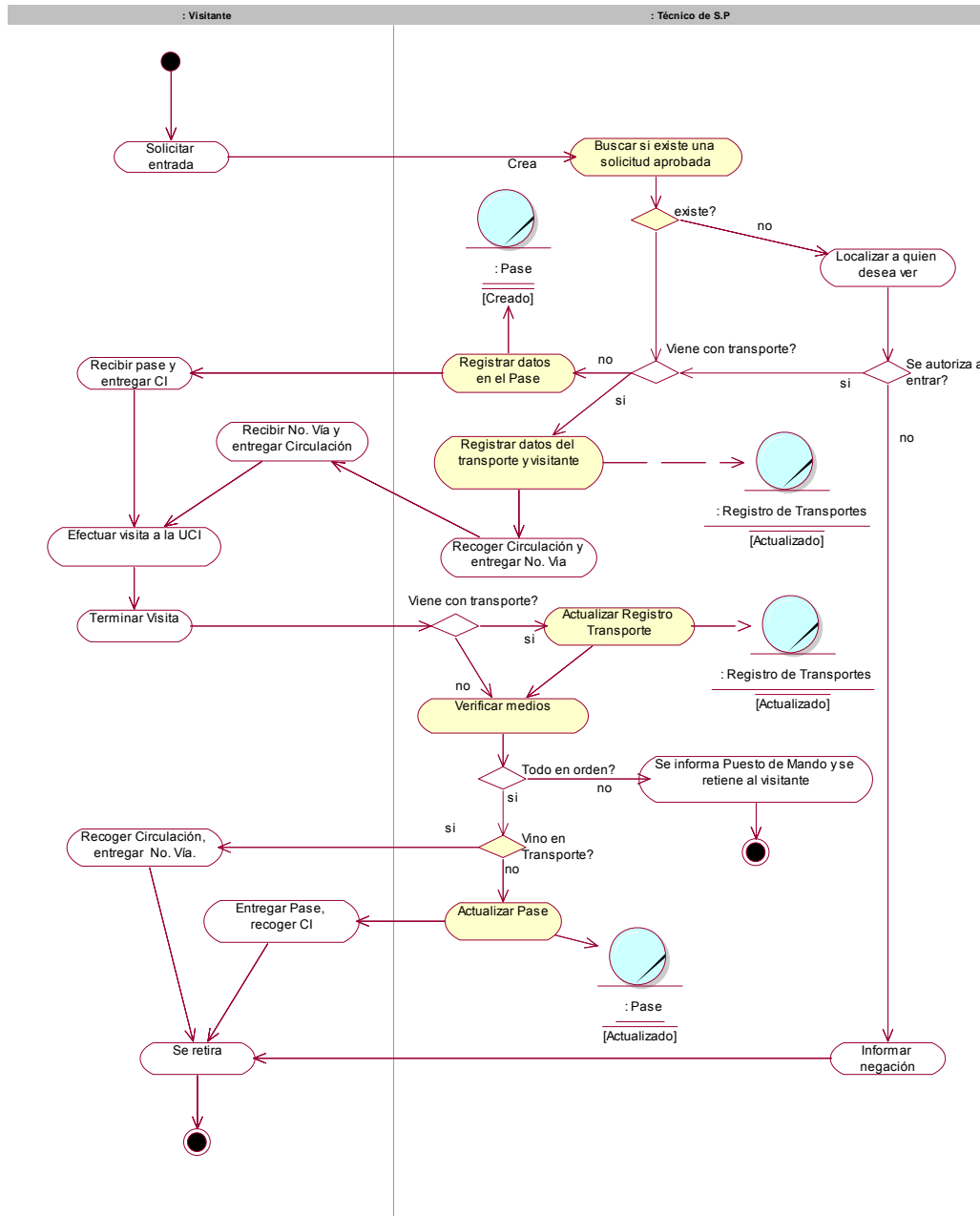


Figura 5. Diagrama de actividades del C.U. "Realizar Visita"

<b>Nombre del caso de uso:</b>	CU3 – Solicitar permiso para medio
<b>Actores del negocio:</b>	Trabajador, Directivo
<b>Propósito:</b>	Permitirle a un trabajador que entre y salga con un medio al centro.
<b>Resumen:</b> El caso de uso inicia cuando un trabajador le solicita permiso a un directivo para entrar y salir del centro con un medio.	
<b>Casos de uso asociados:</b>	
<b>Flujo de trabajo</b>	
<b>Acción del actor</b>	<b>Respuesta del negocio</b>
1 El Trabajador solicita permiso para medio.	2 El Directivo le entrega un documento que contiene datos del medio, como marca, serie, etc.
<b>Prioridad:</b>	Baja.
<b>Mejoras:</b>	Este proceso se puede mejorar automatizándolo, ya que se hará mucho más rápido, seguro, el permiso no se deteriorará, al no ser un documento de papel.
<b>Cursos alternos:</b> -	

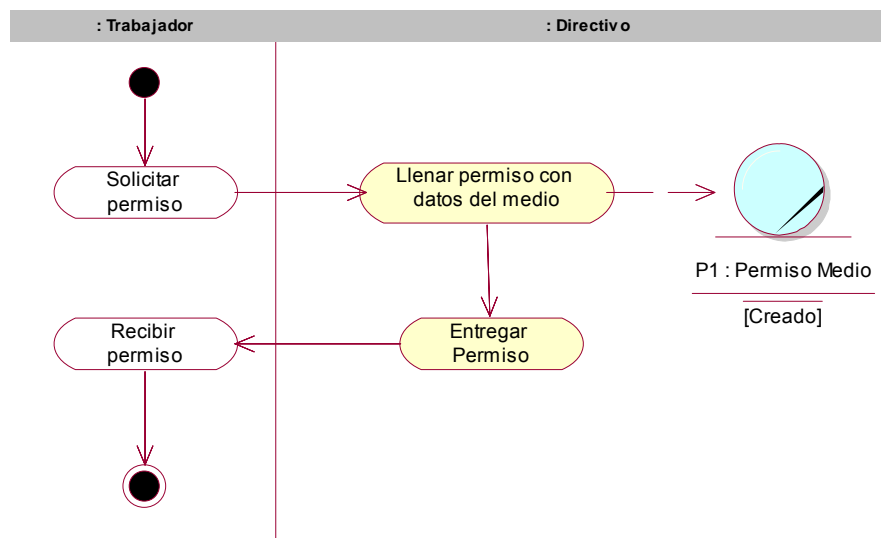


Figura 6. Diagrama de actividades del C.U. "Solicitar Permiso para Medio"

<b>Nombre del caso de uso del negocio:</b>	CU4 – Entrar o Salir de la UCI	
<b>Actores del negocio:</b>	Responsable Seguridad, Trabajador	
<b>Propósito:</b>	Entrar o Salir de la UCI	
<b>Resumen:</b>	<p>El caso de uso inicia cuando un trabajador va entrar a la Universidad o salir de la misma. El Responsable de Seguridad le pide las credenciales y si son validas se le permite la acción que vaya a realizar.</p>	
<b>Casos de uso asociados:</b>		
<b>Flujo de trabajo</b>		
<b>Acción del actor</b>	<b>Respuesta del negocio</b>	
1 El Trabajador llega a la garita u omnibus.	2 El Responsable Seguridad le pide su solapín y comprueba la identidad del portador. Verifica que si sale con medios tenga permiso para sacarlo.	
3 El trabajador entra o sale de la UCI. Fin del proceso.		
<b>Prioridad:</b>	Alta prioridad, porque se verifica la identidad de los trabajadores cuando entren o salgan.	
<b>Mejoras:</b>	La automatización de este proceso permitirá eliminar cualquier intento de pasar con solapines falsos o vencidos, todo de una manera mucho mas segura, rápida y confiable, lo que aumenta la seguridad en la Universidad.	
<b>Cursos Alternos:</b>		
<b>Acción del Actor:</b>	<b>Respuesta del Negocio:</b>	
	2- Si no son validos, se le deniega su entrada o salida (ir a paso 4)	

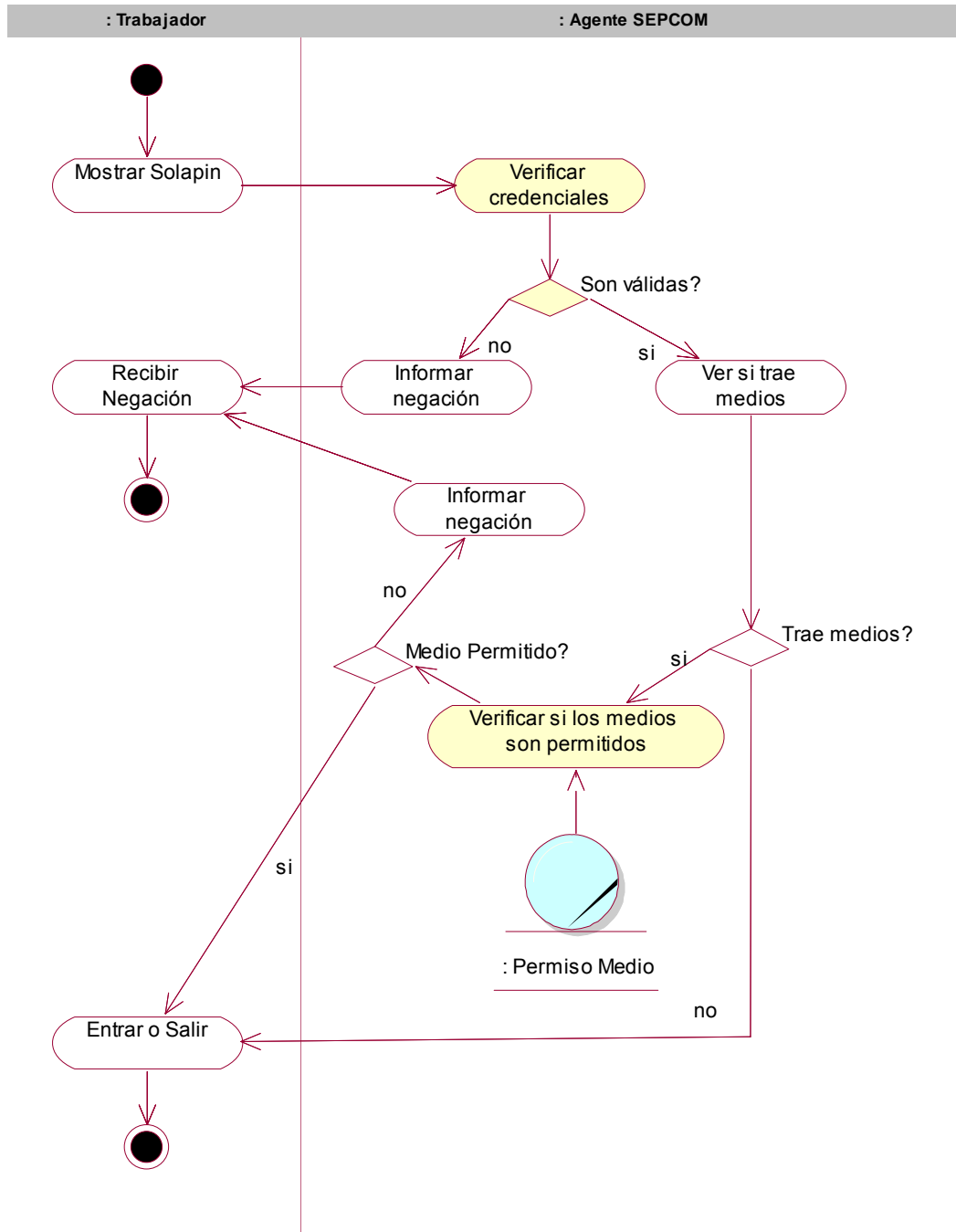


Figura 7. Diagrama de actividades del C.U. "Entrar o Salir de la UCI"

### 3.5 Diagrama de Clases del Modelo de Objetos

El diagrama de clases, como artefacto que se construye para describir el modelo de objetos del negocio, muestra la participación de los trabajadores y entidades del negocio y la relación entre ellos.

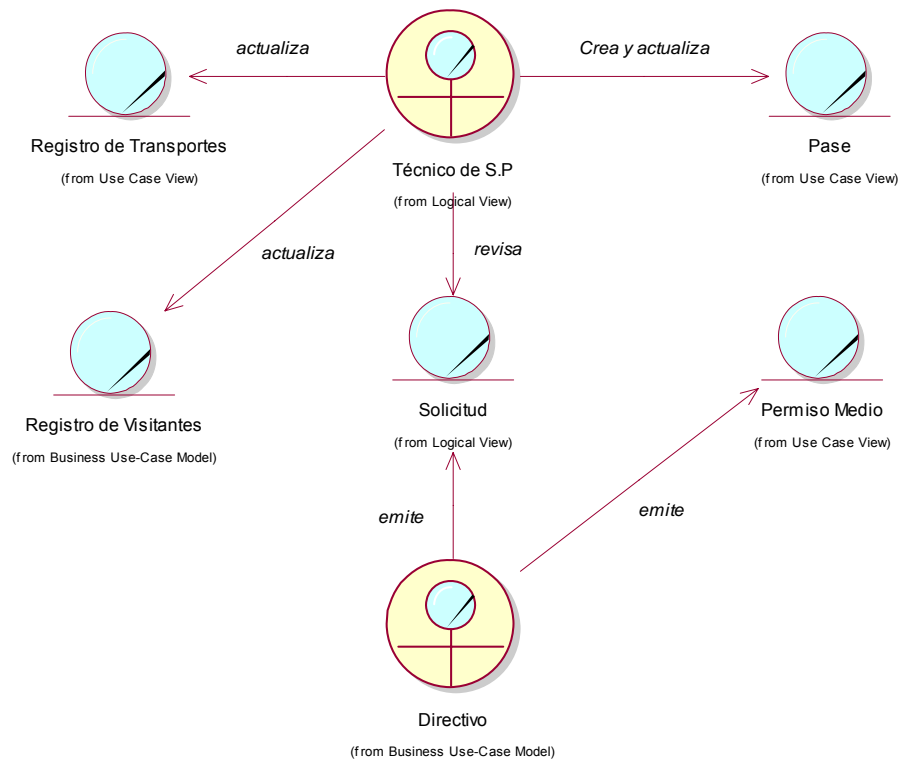


Figura 8. Diagrama de Clases del Modelo de Objetos del Negocio

### 3.6 Requerimientos Funcionales

De acuerdo con los objetivos planteados el sistema debe ser capaz de:

1. Registrar las entradas y salidas (accesos) de los trabajadores de la UCI
  - 1.1. Registrar si el acceso ha sido en algún transporte
  - 1.2. Verificar si una credencial sale o entra dos veces consecutivamente.
2. Solicitar visitas de personal externo a la misma a la Universidad.
3. Aprobar o denegar solicitudes (los Directivos).
4. Registrar entradas y salidas de visitantes, esperados o no, recogiendo los datos pertinentes, si han sido permitidos a entrar.
5. Configurar el sistema.
  - 5.1 Configurar responsables de aprobar solicitudes por área.
  - 5.2 Configurar Grupos y Usuarios del sistema.
  - 5.3 Configurar choferes y responsables de los transportes de la UCI.
  - 5.4 Configurar medios permitidos a trabajadores, para entrar y salir.
  - 5.5 Configurar nomencladores del sistema.
6. Brindar reporte de visitantes, con posibilidad de búsqueda por fecha, datos de los visitantes, etc.
7. Brindar reporte de transportes que han entrado a la UCI, con posibilidad de búsqueda por fecha, datos de los mismos, etc.
8. Brindar reportes de accesos de trabajadores, con posibilidad de búsqueda por fecha, etc.
9. Suministrar información a otros sistemas de la intranet mediante el uso de Servicios Web.
10. Enviar reportes a directivos, a través de correo electrónico.
11. Conocer los datos de un trabajador, dado su solapín.
12. Conocer los datos de un transporte, dado su código o chapa.

### 3.7 Requerimientos no funcionales

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener.

#### **Apariencia o interfaz externa:**

- Diseño sencillo, permitiendo que no sea necesario mucho entrenamiento para utilizar el sistema.
- Paginación de reportes de búsqueda, y listados.
- Diseño perfectamente encuadrado para resoluciones de 800x600, pero preparado para verse en otras resoluciones.

#### **Usabilidad:**

- El sistema podrá ser usado por cualquier persona que posea conocimientos básicos en el manejo de la computadora y de un ambiente Web en sentido general.
- Deberá visualizarse bien en los principales navegadores en la Universidad (Internet Explorer, Firefox, Opera, Maxthon).

#### **Rendimiento:**

- El sistema operará con grandes volúmenes de información, por tanto, se hacen necesarios tiempos de respuestas cortos, al igual que la velocidad de procesamiento de la información, para evitar colas en los puntos de acceso.

#### **Soporte:**

- Se requiere un servidor de bases de datos con las siguientes características:
  - Soporte para grandes volúmenes de datos y velocidad de procesamiento.
  - Tiempo de respuesta rápido.
- Plataforma .NET versiones 1.0 / 1.1 o bien plataforma Mono.
- Por parte del cliente se requiere un navegador capaz de interpretar JavaScript.

**Portabilidad:**

- Necesidad de que el sistema sea multiplataformas (Al menos Windows y Linux).

**Seguridad:**

- Identificar al usuario antes de que pueda realizar cualquier acción sobre el sistema.
- Garantizar que la información sea vista únicamente por quien tiene derecho a verla.
- Garantizar que las funcionalidades del sistema se muestren de acuerdo al nivel de usuario que este activo.
- Protección contra acciones no autorizadas o que puedan afectar la integridad de los datos.
- Verificación sobre acciones irreversibles (eliminaciones).

**Legales:**

Debe cumplir con:

- Decreto Ley – 186 / 98  
2da Sección, Artículos 5, 8, 11, Inciso D.
- Reglamento del Decreto Ley 186 (Resolución No. 2 / 2001 del Ministerio del Interior). Artículo 1. Capítulo IV (Artículos 3, 4, 90)
- El Plan de Seguridad y Protección del Centro.

**Confiabilidad:**

- El sistema de gestión de bases de datos debe tener soporte para recuperación ante fallos y errores.

**Funcionalidad:**

- Capacidad de búsqueda con un tiempo menor que 5 segundos.



- Mínima cantidad de páginas para ejecutar todas las funciones posibles, es decir, agrupar funciones afines en las mismas páginas.

**Software:**

- Navegador Internet Explorer 5.5. o superior, o compatible con Mozilla.
- Microsoft SQL Server 2000.
- Internet Information Server 5.0 ó posterior
- Plataforma .NET 1.0 ó posterior

### 3.8 Descripción del sistema propuesto

Para cumplir los objetivos propuestos al inicio de este trabajo, y teniendo en cuenta todos los requerimientos planteados, el sistema que se propone se ha dividido en tres módulos lógicos principales, Visitantes, Accesos, y Administración.

Se definen además varios roles para diferenciar el nivel de acceso al sistema, los solicitantes, los directivos, los administradores, y los agentes de seguridad.

La parte de Visitantes gestiona todo lo referente a la solicitud de visitas a la UCI por parte de trabajadores y estudiantes, de personas ajenas al centro. También gestiona las entradas y salidas de visitantes, esperados o no. Para solicitar una visita, el solicitante debe llenar los datos pertinentes, como nombre, apellidos y carné de identidad de los visitantes, si vienen en transportes, debe decir la chapa, tipo de transporte y carné del chofer, entre otros datos. Del mismo modo, si el visitante trajera algún medio, como una laptop por ejemplo, debe incluirlo en los datos de la solicitud, especificando la marca, color, etc. Debe además informar la fecha y hora de inicio y fin de la visita, así como el motivo.

Los directivos pueden ver las solicitudes hechas a su área, y son los encargados de aprobar o denegar la solicitud, el sistema permite avisar al solicitante cuando su solicitud sea aprobada o denegada, a través de un mensaje de correo electrónico.

El sistema registra la entrada de los visitantes, una vez en la UCI, para esto, el técnico en la garita debe verificar los datos del visitante y que exista una solicitud aprobada para el mismo, si es así, se le permite la entrada y se actualizan los datos de la visita, igualmente se registra la hora de salida.

Otra de las utilidades del sistema es la gestión de accesos de personas, aquí se controlan las entradas y salidas de trabajadores del centro. La identificación de la persona, se realiza mediante un código de barra impreso al dorso del solapín personal de cada trabajador. Los dispositivos de lectura de código de barras son de varios tipos, de acuerdo con el lugar donde se vayan a utilizar, por ejemplo, fijos, para la garita, o portátiles para que lo lleven los agentes de seguridad.

El sistema permite a directivos ver reportes en línea, sobre el estado de los visitantes a la UCI en determinado momento, ya que se pueden hacer búsquedas por ciertos campos de los mismos. Permite además conocer si estos visitantes han entrado con transportes y/o medios. Otros reportes disponibles, son los de accesos, los cuales permiten conocer en determinado momento, cuáles y cuántas personas y/o transportes han entrado o salido de la Universidad. Todos estos datos son necesarios para el control de la seguridad interna.

El tercer módulo es el de administración. Este módulo permite a usuarios con ciertos privilegios, modificar el funcionamiento y configuración del sistema. En este módulo se controlan quiénes pueden aprobar o denegar visitas, se definen los trabajadores que pueden entrar con medios o transportes, etc. Además, existe la posibilidad de agrupar a los usuarios del sistema en grupos genéricos para luego brindar una separación de los permisos y lograr niveles de acceso diferenciados, a las distintas partes del sistema.

### 3.9 Definición de los actores del sistema

ACTORES DEL SISTEMA	JUSTIFICACIÓN
Solicitante	Cualquier persona, estudiante o trabajador de la UCI que solicite visitas de personal ajeno a la Universidad
Directivo	Persona trabajadora de la UCI con cargo en la dirección de la Universidad, pertenece a determinada área, puede aprobar solicitudes de visita a su área.
Timer	Actor genérico para indicar un temporizador en el sistema operativo del servidor, el cual ejecuta ciertas tareas programadas del sistema.
Sistema Externo	Sistemas de la intranet universitaria, que solicitan reportes a través de WebServices.
Técnico de Seguridad y Protección	Encargado de responder por el control de los medios básicos y controlar el trabajo de los agentes del SEPCOM. Gestiona la autorización de las personas ajenas al centro, que deseen entrar, revisando si hay alguna solicitud de entrada previa o haciendo la gestión de su permiso.
Administrador	Controla la configuración del sistema, define los usuarios del mismos, distribuyéndolos en grupos con nivel de acceso diferenciados.
Responsable Seguridad	Encargado de la seguridad interna, en este trabajo se les llama también Agentes SEPCOM indistintamente.

### 3.10 Modelo de casos de uso del sistema

El modelado de Casos de Uso es la técnica más efectiva y a la vez la más simple para modelar los requisitos del sistema desde la perspectiva del usuario. Los Casos de Uso se utilizan para modelar cómo un sistema o negocio funciona, o cómo los usuarios desean que funcione el futuro sistema. No es realmente una aproximación a la orientación a objetos; es una forma de modelar procesos. Es una manera muy buena de dirigirse hacia el análisis de sistemas orientado a objetos. Los casos de uso son generalmente el punto de partida del análisis orientado a objetos con UML.

A continuación se presentan los casos de uso identificados para satisfacer los requerimientos funcionales del sistema:

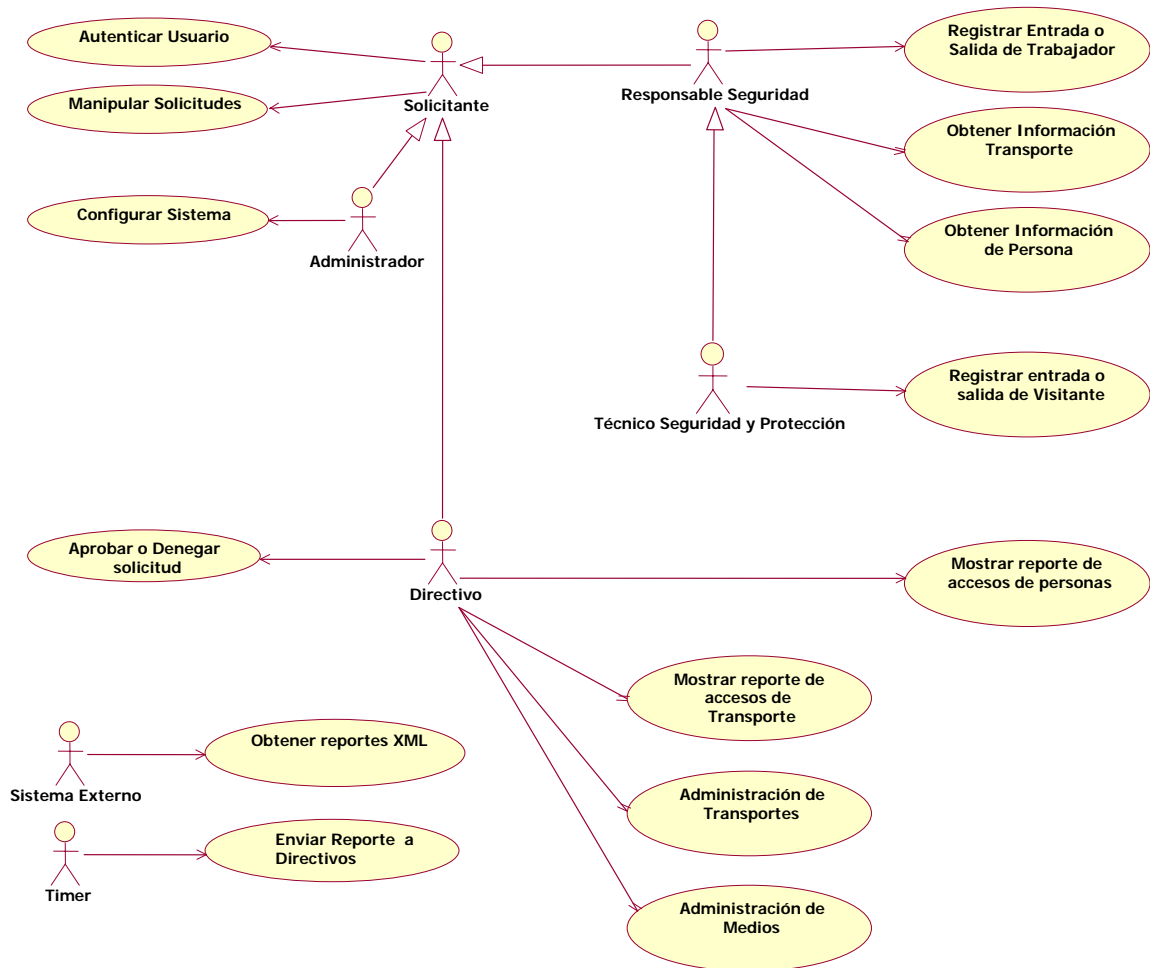
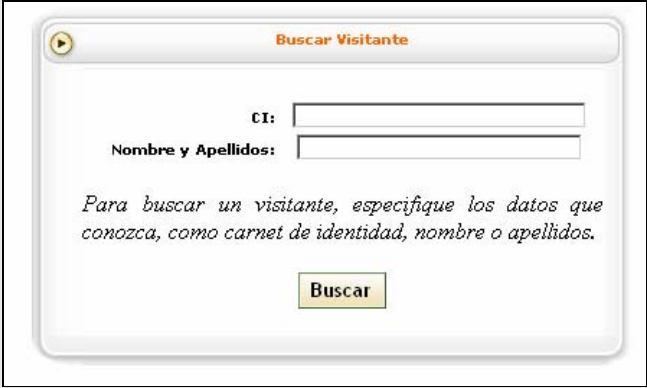


Figura 9. Diagrama de Casos de Uso del Sistema

### 3.11 Expansión de los Casos de Uso

Mediante los casos de uso expandidos se describe en detalle el funcionamiento de un caso de uso. A continuación se muestra la explicación de cada caso de uso, y se presentan prototipos de la interfaz.

<b>Caso de Uso:</b>	<b>Registrar salida o entrada de visitante</b>
Actor(es):	Técnico de Seguridad y Protección (Inicia)
Propósito:	Que el Técnico de S.P. registre la entrada o salida de un visitante.
Resumen:	<p>El caso de uso se inicia cuando llega un visitante solicitando entrar o salir de la UCI. El visitante puede ser esperado o no, si es esperado, el Técnico de S.P. busca la solicitud (<b>Pantalla 1</b>), verifica los datos, y registra la entrada (<b>Pantalla 2</b>) (A). Si el visitante no es esperado, se gestiona su entrada, si es permitido por un Directivo, se le toman los datos y se registra la entrada (<b>Pantalla 2</b>) (A). Si el visitante viene con algún medio o transporte, se registra para que pueda salir con él. (<b>Pantalla 2</b>, B y C). Mientras la visita está efectuándose, ésta se encuentra en el estado de “<b>En curso</b>”.</p> <p>Para salir, el Técnico de S.P. busca la visita creada anteriormente (<b>Pantalla 1</b>) y registra su salida (<b>Pantalla 3</b>, A), el sistema almacena la hora cuando esto sucede.</p> <p>Hay que destacar que la <b>Pantalla 1</b>, sirve para buscar visitantes, tanto dentro de la UCI, como pendientes de visita.</p>
Referencias:	R4
Precondiciones:	
Poscondiciones:	Al finalizar el caso de uso, la visita queda almacenada con el estado de <b>Efectuada</b> , si salio, o “ <b>En Curso</b> ” si entró.
Requisitos Especiales:	
Prototipo	 <p style="text-align: center;"><b>Pantalla 1</b></p>

UCI Acceso Sistema para el control de accesos y visitas

Mis solicitudes Nueva solicitud Manuel Alejandro Gil [ Salir ]

**Datos de la visita**

Fechas:  
Inicio: 11/05/2005 09:55 AM  
Fin:

Otros datos:  
Motivo: Reunion de Trab.  
Local a visitar: Informatización

Comentarios:  
Visitante a la dirección de informatización.

Registrar Entrada Cancelar

**Datos del visitante**

Nombre y Apellidos: Juan Martinez  
Sexo: M  
CI: 8002051485  
Organismo:

Viene con transporte

+ Nuevo medio

Pantalla 2

UCI Acceso Sistema para el control de accesos y visitas

Mis solicitudes Nueva solicitud Manuel Alejandro Gil [ Salir ]

**Datos de la visita**

Fechas:  
Inicio: 4-mayo-2005 9:19 AM

Otros datos:  
Motivo: Visita de Trabajo  
Local a visitar: Informatización

Comentarios:

Registrar Salida Cancelar

**Datos del visitante**

Nombre y Apellidos: Juan Martinez  
Sexo: M  
CI: 80020518547  
Organismo:

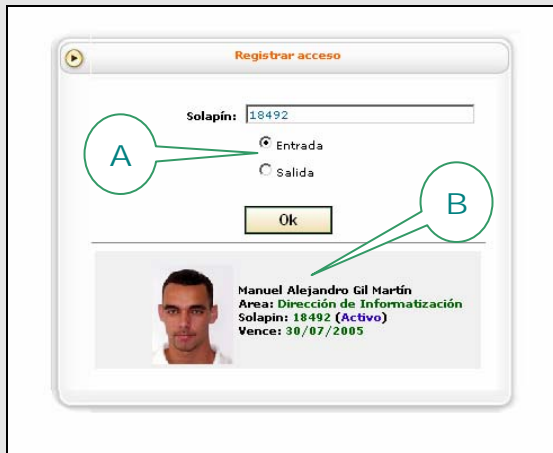
Viene con transporte

+ Nuevo medio

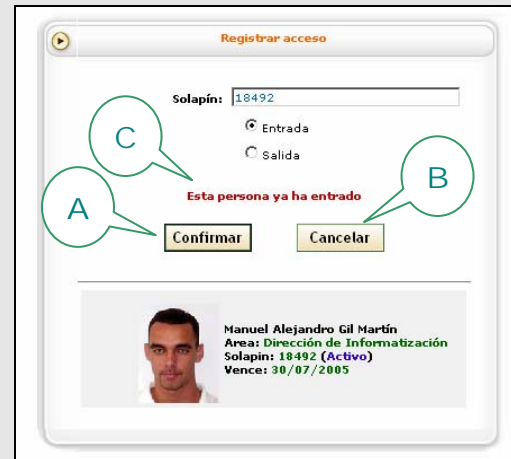
Pantalla 3

<b>Caso de Uso:</b>	<b>Registrar Entrada o Salida de Trabajador</b>
Actor(es):	Responsable Seguridad(Inicia)
Propósito:	Registrar la entrada o salida de un trabajador o transporte, verificando la validez del solapín o código.
Resumen:	<p>El caso de uso se inicia cuando se le pide el solapín a un trabajador para registrar su entrada o salida de la UCI (<b>Pantalla 1, A</b>), o se registra la entrada o salida de un transporte. El sistema informa los datos del portador (<b>Pantalla 1, B</b>), mostrando si el solapín es válido, su fecha de vencimiento, y otros datos, además de la foto del portador. El caso de uso puede configurarse para que se le de entrada o salida automáticamente al leer el solapín, para reducir las operaciones y el tiempo, cuando el proceso se haga para entradas o salidas masivas. En caso el caso de los transportes, se muestran datos como chapa, responsables, choferes, color, etc. (<b>Pantalla 3</b>)</p> <p>Si un solapín no es válido, en el caso de los trabajadores, porque esté inactivo u otra causa, se le informa al Responsable Seguridad o Técnico de S.P. Este verifica la situación y puede continuar con la acción o cancelarla. Si se detecta un <i>pass-back</i>, el sistema lo informa (<b>Pantalla 2, C</b>) y pregunta qué se desea hacer, si continuar con la acción (<b>Pantalla 2, A</b>) o cancelarla (<b>Pantalla 2, B</b>), en ambos casos, se crea un registro sobre el problema, para un eventual análisis.</p>
Referencias:	R 1
Precondiciones:	
Poscondiciones:	Se ha registrado la entrada o salida de un trabajador o transporte, con la fecha y hora actual.
Requisitos Especiales:	La realización de este caso de uso debe ser rápida, no más de 2 segundos por lectura.

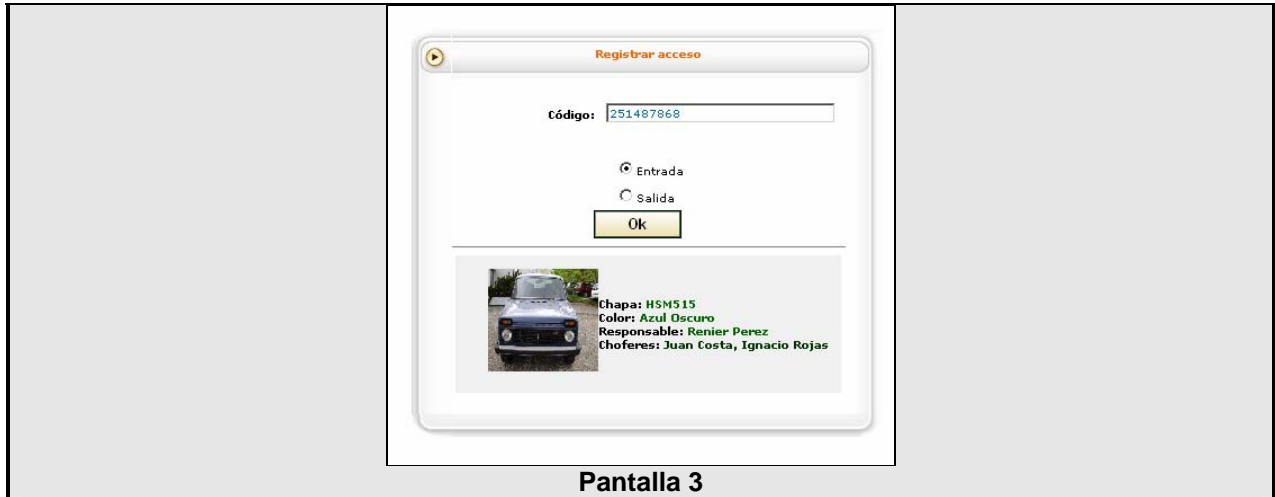
Prototipo



Pantalla 1



Pantalla 2




<b>Caso de Uso:</b>	<b>Obtener Información de Persona</b>
Actor(es):	Responsable Seguridad(Inicia)
Propósito:	Obtener la información de una persona trabajadora de la UCI
Resumen:	El caso de uso se inicia cuando un Responsable Seguridad solicita información de una persona, para verificar la veracidad de su solapín. Para esto, introduce el número del solapín y el sistema le devuelve los datos del mismo, incluyendo la foto y el estado del solapín (Activado o Desactivado)
Referencias:	R11
Precondiciones:	
Poscondiciones:	
Requisitos Especiales:	

Prototipo

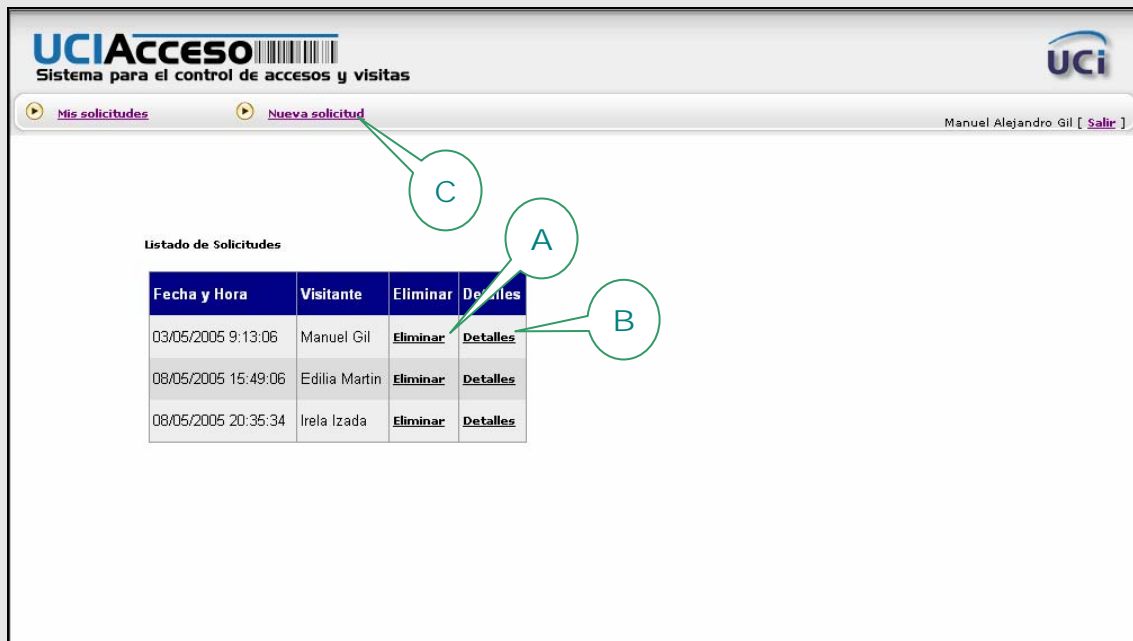




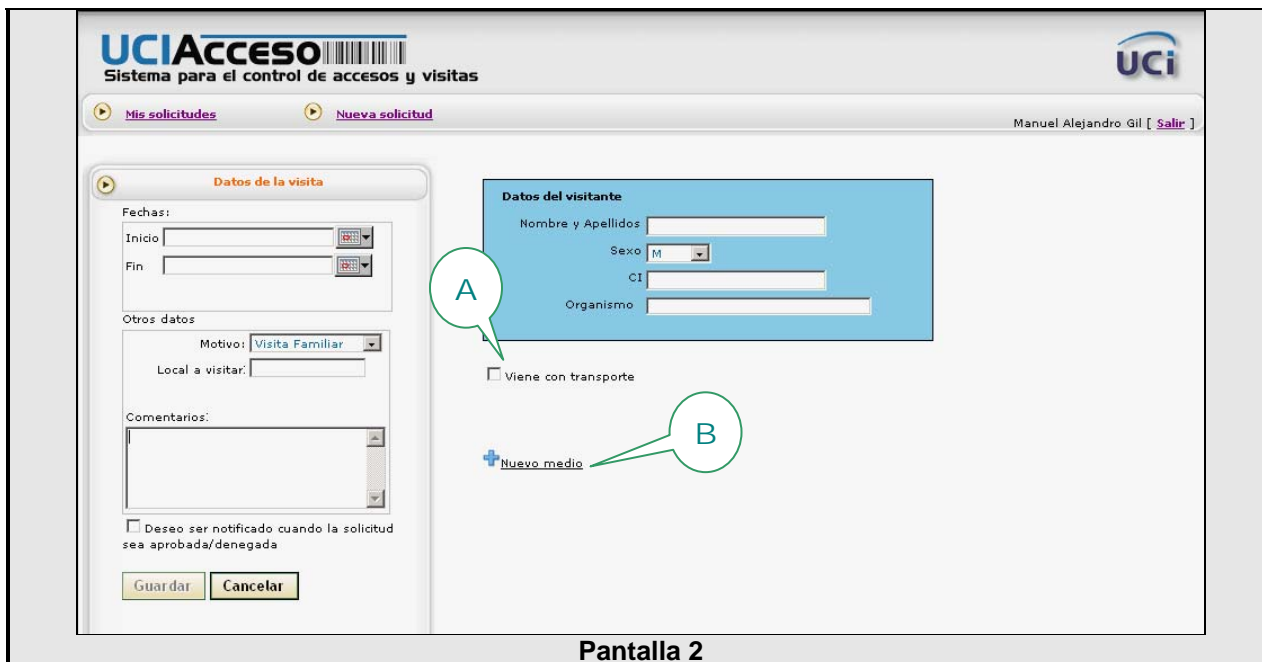
<b>Caso de Uso:</b>	<b>Obtener Información de Transporte</b>
Actor(es):	Responsable Seguridad(Inicia)
Propósito:	Obtener información de un transporte, dado su chapa o código
Resumen:	El caso de uso se inicia cuando un Responsable Seguridad solicita información de un transporte perteneciente a la UCI, El caso de uso permite hacer la búsqueda por la chapa o por el código del transporte, el cual se encuentra codificado con barras en su exterior. El caso de uso permite ver los responsables y choferes del transporte.
Referencias:	R12
Precondiciones:	
Poscondiciones:	
Requisitos Especiales:	
Prototipo	<div style="text-align: center;"><p><b>Pantalla 1</b></p></div>

<b>Caso de Uso:</b>	<b>Manipular Solicitudes</b>
Actor(es):	Solicitante (Inicia)
Propósito:	Crear, modificar y eliminar las solicitudes de un Solicitante.
Resumen:	<p>El caso de uso se inicia cuando el solicitante necesita crear, modificar o eliminar una solicitud. Primeramente se le muestra un listado de sus solicitudes anteriores, se le da la posibilidad de eliminarlas (A), ver sus detalles (B), y crear nuevas (C). (<b>Pantalla 1</b>).</p> <p>En la <b>Pantalla 2</b>, se le presentan al usuario los datos que debe llenar de la solicitud, si el visitante viene con transporte o medios, debe especificar los datos de los mismos (A y B). Puede salvar los cambios, o cancelar la acción.</p> <p>Si el visitante elimina alguna solicitud de visita que ya ha sido efectuada, ésta se le deja de mostrar pero no se elimina físicamente de la base de datos, para reportes futuros a los directivos. Si hay algún problema, se le informa al solicitante, y se genera un registro para posterior análisis del administrador.</p> <p>El solicitante puede especificar si desea ser notificado por correo electrónico cuando su solicitud haya sido aprobada o denegada.</p>
Referencias:	R2
Precondiciones:	
Poscondiciones:	Al efectuarse el caso de uso, se ha creado una nueva solicitud o se han actualizado los datos de una existente.
Requisitos Especiales:	

Prototipo



Pantalla 1



Caso de Uso:	Aprobar o denegar solicitud
Actor(es):	Directivo (Inicia)
Propósito:	Permitir que los directivos aprueben o denieguen las solicitudes de visitas hechas por Solicitantes en su área.
Resumen:	El caso de uso se inicia cuando el directivo revisa las solicitudes hechas en su área ( <b>Pantalla 1</b> ). Puede eliminarlas o ver sus detalles ( <b>Pantalla 2</b> ).  Si está de acuerdo, la aprueba ( <b>Pantalla 2</b> ) (A), sino, la deniega (B). Si el Directivo considera que es otro Directivo el encargado de manipular esta solicitud, la remite al mismo (C).
Referencias:	R3
Precondiciones:	Que exista al menos una solicitud
Poscondiciones:	La solicitud queda aprobada o denegada, si es remitida a otro Directivo, continúa pendiente.
Requisitos Especiales:	

Prototipo

**UCI Acceso**  
Sistema para el control de accesos y visitas

Mis solicitudes Nueva solicitud Manuel Alejandro Gil [ Salir ]

Listado de Solicitudes de la Facultad 9

Fecha y Hora	Solicitante	Eliminar	Detalles
03/05/2005 9:13:06	Manuel Gil	Eliminar	Detalles
08/05/2005 15:49:06	Edilia Martin	Eliminar	Detalles
08/05/2005 20:35:34	Irela Izada	Eliminar	Detalles

Pantalla 1

**UCI Acceso**  
Sistema para el control de accesos y visitas

Mis solicitudes Nueva solicitud Manuel Alejandro Gil [ Salir ]

**Datos de la visita**

Fechas:  
Inicio: 12-mayo-2005 2:00 PM  
Fin: 12-mayo-2005 5:00 PM

Otros datos:  
Motivo: Visita Familiar  
Local a visitar: Beca  
Comentarios:

**Datos del visitante**

Nombre y Apellidos: Juan Cuesta  
Sexo: M  
CI: 80020814928  
Organismo: ETECSA

Viene con transporte

**Datos del transporte**


Chapa: HTM515  
Marca: Lada  
Observaciones:

Aprobar Denegar Cancelar

Remitir solicitud a... Melchor Gil Ok

Pantalla 2

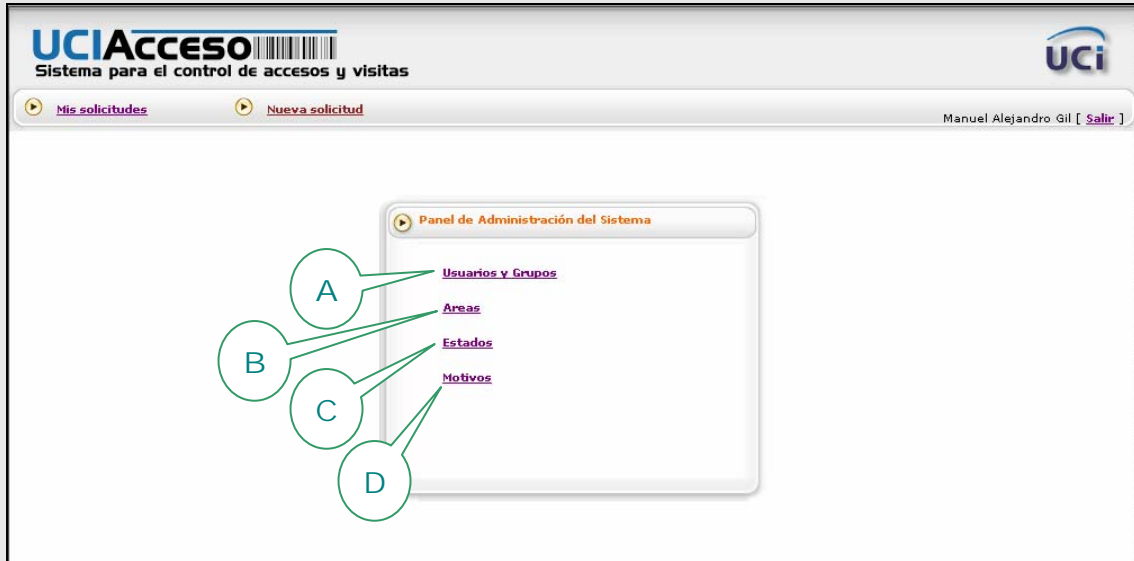
<b>Caso de Uso:</b>	<b>Autenticar Usuario</b>
Actor(es):	Solicitante (Inicia)
Propósito:	Posibilita a los usuarios identificarse contra el sistema para acceder a las opciones disponibles para su nivel de acceso.
Resumen:	El caso de uso se inicia cuando en Solicitante introduce sus credenciales del dominio para que el sistema las verifique y haga todas las funciones de ahí en adelante en su nombre. Es importante destacar que el usuario ya esta autenticado cuando entra a cualquier parte del sistema, ya que se utiliza Autenticación Windows, sin embargo, el caso de uso permite a un usuario, trabajar en una computadora en la que otro usuario este logueado, sin tener que iniciar la misma con sus credenciales.
Referencias:	
Precondiciones:	Que el usuario posea una cuenta de dominio en la UCI.

Poscondiciones:	El sistema se ejecuta ahora como el usuario recientemente autenticado, no como el usuario al que pertenece la sesión con la que se inició la computadora.
Requisitos Especiales:	
Prototipo	
	
<b>Pantalla 1</b>	

<b>Caso de Uso:</b>	<b>Configurar Sistema</b>
Actor(es):	Administrador(Inicia)
Propósito:	Permitirle a un Administrador modificar la configuración del sistema, para definir cosas como: Responsables por área, usuarios y grupos del sistema, nomencladores, etc.
Resumen:	<p>El caso de uso se inicia cuando el Administrador accede a la página de administración. Puede seguir varios caminos, en este punto, tiene la posibilidad de configurar <i>Estados de Visitas</i> (C), <i>Motivos de Visitas</i> (D), <i>Usuarios y Grupos</i> (A), <i>áreas y sus responsables</i> (B).</p> <p>En los <i>estados de visitas y motivos de visitas</i> (<b>Pantallas 2 y 3</b> respectivamente), configura los posibles estados en los que se puede encontrar una visita o solicitud y los motivos que la ocasionan. En <i>Usuarios y Grupos</i> (<b>Pantalla 4</b>), se pueden adicionar usuarios existentes en el dominio para que formen parte del sistema, agrupándolos por grupos que son también configurables, brindando así un nivel diferenciado de acceso a las distintas partes del sistema. En <i>Configuración de áreas</i> (<b>Pantalla 5</b>) se definen Directivos responsables a autorizar visitas por cada área.</p> <p>Las áreas (<b>Pantalla 5</b>) no se pueden cambiar ni eliminar, pues son las áreas en las que está dividida la UCI y se obtienen de otro sistema de la Intranet, a través de un WebService.</p> <p>Siempre se le pide confirmación al administrador antes de eliminar cualquier</p>

	cosa, previniéndole de una pérdida no intencional de los datos. Si ocurre algún problema, este se registra para su posterior análisis.
Referencias:	R5.1 R5.2 R5.5
Precondiciones:	
Poscondiciones:	El sistema queda actualizado con los cambios efectuados.
Requisitos Especiales:	

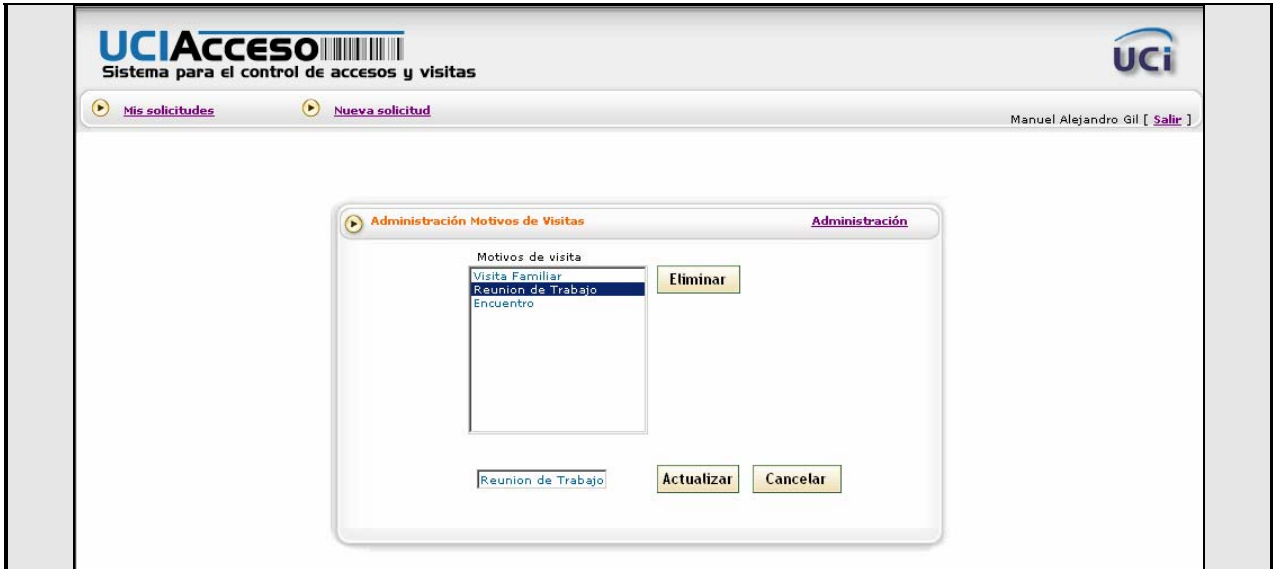
Prototipo



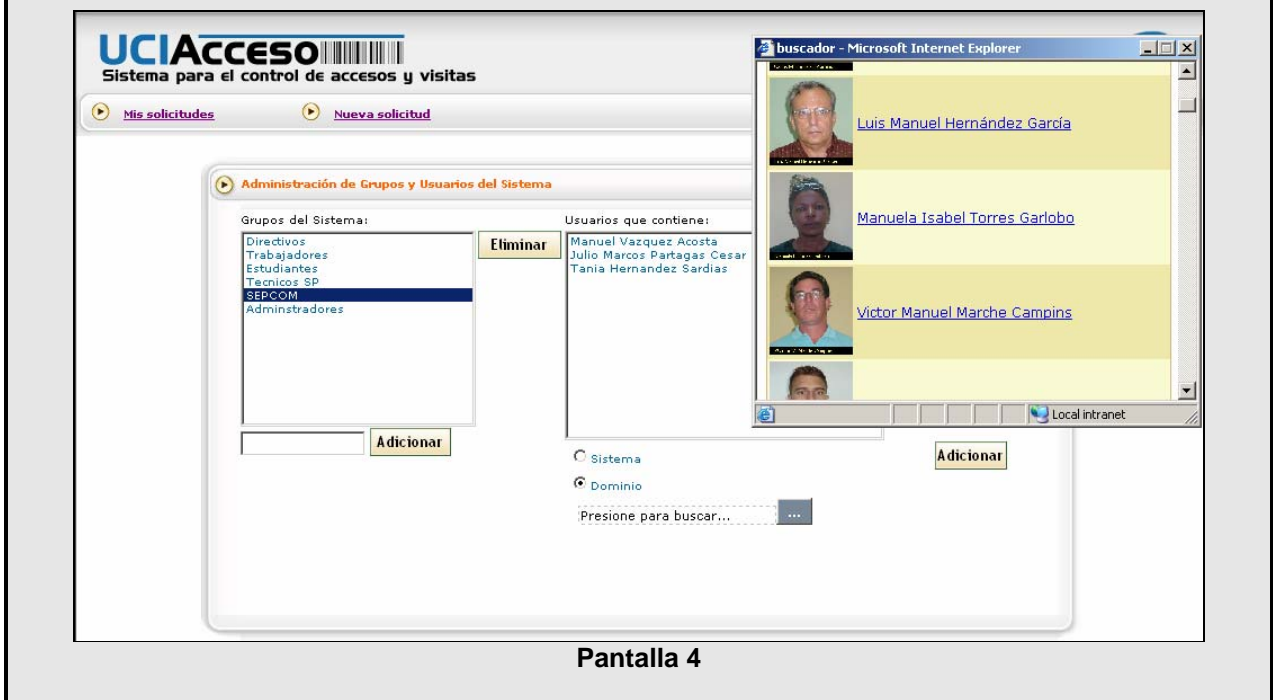
Pantalla 1



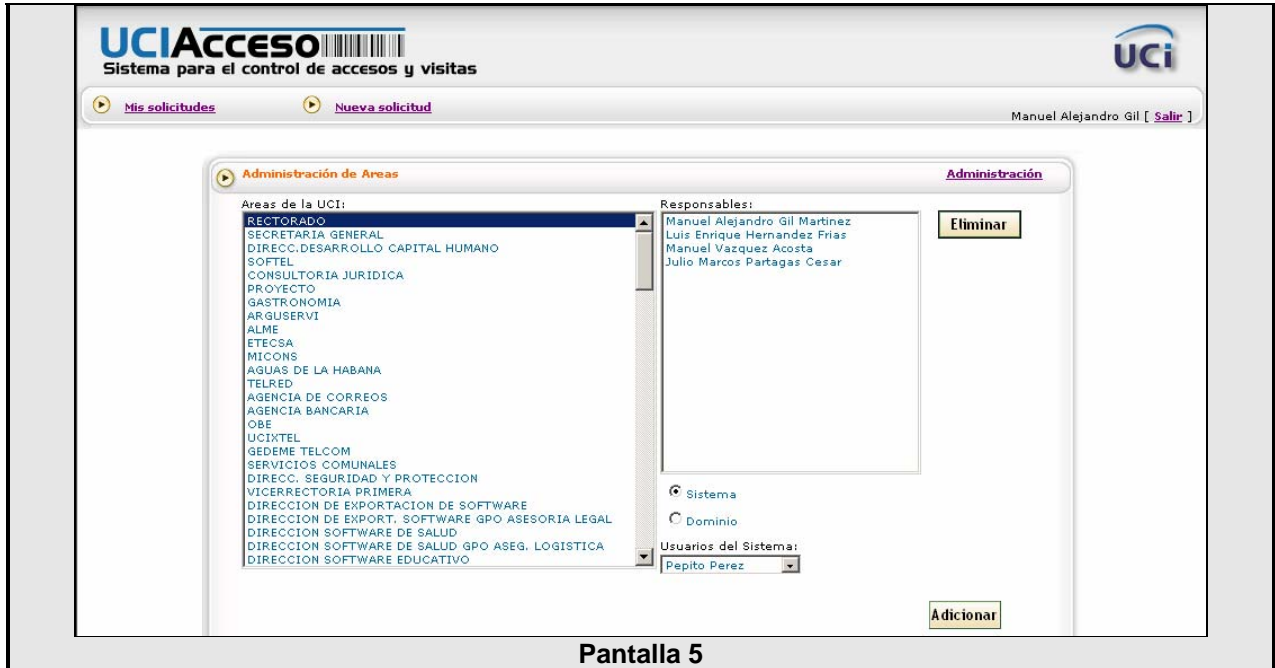
Pantalla 2



Pantalla 3



Pantalla 4



Pantalla 5

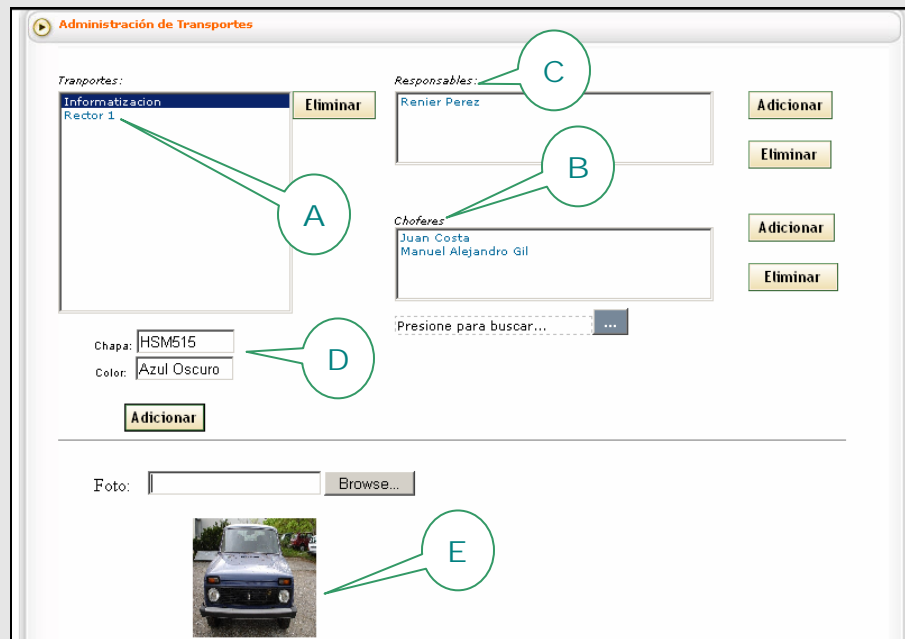


<b>Caso de Uso:</b>	<b>Enviar reporte a Directivos</b>
Actor(es):	Timer (Inicia)
Propósito:	Enviar reportes a Directivos con resúmenes estadísticos de los accesos de trabajadores y visitantes del día.
Resumen:	<p>El caso de uso se inicia cuando el sistema operativo del servidor donde corre la aplicación ejecuta la misma, una vez al día, a una hora configurada por el administrador del servidor. La aplicación contiene un archivo de configuración con los reportes que debe enviar, y a quiénes. Los reportes son los mismos accesibles online. Pero con ciertas modificaciones para poder ser enviados por correo electrónico.</p> <p>El módulo puede instalarse en un servidor aparte de la aplicación Web, ya que es completamente independiente, permitiendo portabilidad. Además puede ser reutilizado para los reportes de otras aplicaciones de la intranet que requieran el envío de correo.</p>
Referencias:	R10
Precondiciones:	Los reportes de los casos de uso “Mostrar reporte de accesos de Trabajadores” y “Mostrar reporte de accesos de transportes” deben ser funcionar perfectamente.
Poscondiciones:	
Requisitos Especiales:	El sistema operativo donde se monte el módulo, debe permitir la ejecución de tareas programadas. El servidor donde se ejecute este módulo, debe poder acceder por SMTP al servidor de correo (también configurable), ya que en ocasiones existen políticas de seguridad que evitan que esto sea posible.
Prototipo – el caso de uso no tiene interfaz visual -	

<b>Caso de Uso:</b>	<b>Obtener reportes XML</b>
Actor(es):	Sistema Remoto (Inicia)
Propósito:	Le posibilita a un sistema de la intranet obtener ciertos reportes para su uso posterior.
Resumen:	El caso de uso se inicia cuando un sistema de la intranet universitaria solicita un reporte en XML, a través de WebServices. Típicamente, estos reportes serán totales sobre accesos o visitantes, o información del horario de los accesos de un trabajador. El formato es XML porque se brinda la información pura, para que la aplicación que los utiliza pueda darle el formato que desee.
Referencias:	R9
Precondiciones:	
Poscondiciones:	
Requisitos Especiales:	
Prototipo – el caso de uso no tiene interfaz visual -	

<b>Caso de Uso:</b>	<b>Administrar Transportes</b>
<b>Actor(es):</b>	Directivo (Inicia)
<b>Propósito:</b>	Configurar la información necesaria para controlar el acceso de transportes de la UCI
<b>Resumen:</b>	El caso de uso se inicia cuando un directivo accede a la página de administración de transportes. En este punto, puede adicionar, o eliminar transportes al sistema. Dichos transportes pertenecen a la UCI y toda su información la manipula la Dirección de Transportes. Para el control del acceso solo hace falta una parte de esta información. Por cada transporte, se pueden definir choferes y responsables. La pantalla muestra un listado con los transportes definidos (A), los choferes (B), los responsables(C), datos del transporte (D), y una foto (E).
<b>Referencias:</b>	R 5.3
<b>Precondiciones:</b>	
<b>Poscondiciones:</b>	Los datos de los transportes quedan actualizados.
<b>Requisitos Especiales:</b>	

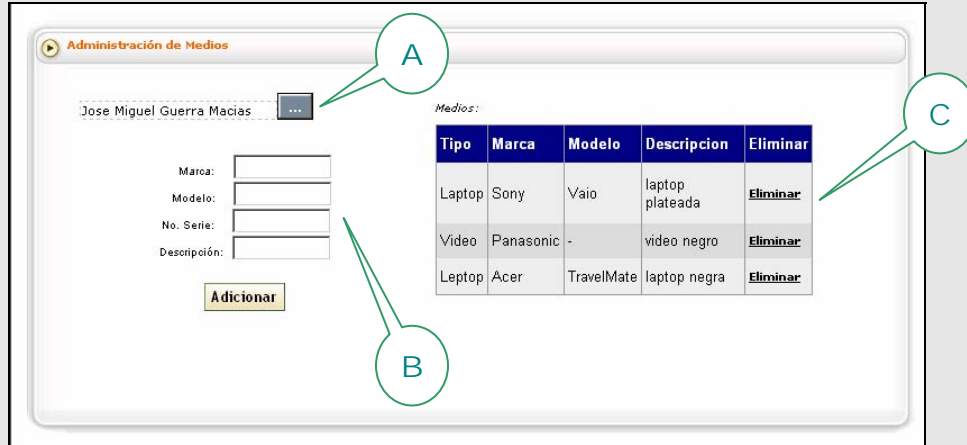
Prototipo



**Pantalla 1**

<b>Caso de Uso:</b>	<b>Administrar Medios</b>
Actor(es):	Directivo (Inicia)
Propósito:	Configurar los trabajadores que pueden salir del centro con medios.
Resumen:	El caso de uso se inicia cuando el directivo encargado de permitir la salida de medio a trabajadores, accede a la página con este fin. En la misma, el Directivo puede buscar al trabajador (A), adicionar medios especificando sus datos (B), y ver un listado con los medios permitidos para el trabajador (C). Con esta información definida, el trabajador puede salir del centro con el medio y no tener problemas cuando el Responsable Seguridad verifique los mismos.
Referencias:	R 5.4
Precondiciones:	
Poscondiciones:	Los medios permitidos por trabajador quedan actualizados.
Requisitos Especiales:	

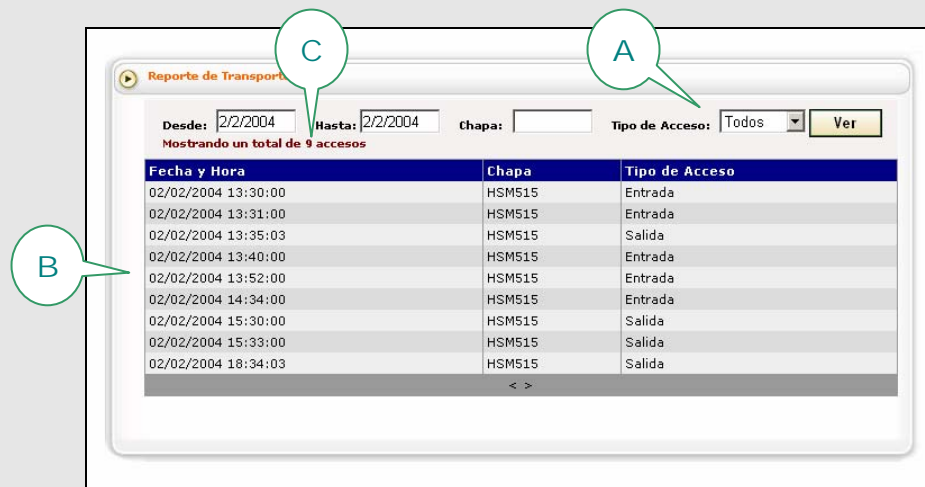
Prototipo



Pantalla 1

<b>Caso de Uso:</b>	<b>Mostrar reporte de accesos de transportes</b>
Actor(es):	Directivo (Inicia)
Propósito:	Muestra el reporte de accesos de transportes a la UCI
Resumen:	<p>El caso de uso se inicia cuando un Directivo abre la página de reporte de transportes. El caso de uso muestra al Directivo un listado (B) con los accesos de entrada o salida de los transportes de la UCI. El mismo se puede ordenar por cualquiera de los campos mostrados, haciendo click sobre el nombre del campo. Se puede filtrar el listado atendiendo a los distintos campos (A). Se muestra un total de los accesos mostrados (C). Se le llama acceso a las acciones de entrada o salida de los transportes.</p> <p>El listado puede ser paginado si el resultado es muy extenso (más de 50 accesos).</p> <p>El interés principal del caso de uso es conocer en determinado momento, cuales son los transportes dentro de la UCI, por eso, en el reporte se muestran tanto transportes de la UCI, como los pertenecientes a visitantes.</p> <p>El listado posee la propiedad de mostrar una versión imprimible si el fin es enviarlo por correo electrónico (Ver Caso de Uso “<b>Enviar Reporte</b>”).</p>
Referencias:	R7
Precondiciones:	
Poscondiciones:	
Requisitos Especiales:	

Prototipo



Pantalla 1

<b>Caso de Uso:</b>	<b>Mostrar reporte de accesos de personas</b>
Actor(es):	Directivo (Inicia)
Propósito:	Muestra el reporte de accesos de trabajadores a la UCI
Resumen:	<p>El caso de uso se inicia cuando un Directivo abre la página de reporte de trabajadores. El caso de uso muestra al Directivo un listado (B) con los accesos de entrada o salida de los trabajadores de la UCI. El mismo se puede ordenar por cualquiera de los campos mostrados, haciendo click sobre el nombre del campo. Se puede filtrar el listado atendiendo a los distintos campos (A). Se muestra un total de los accesos mostrados (C). Se le llama acceso a las acciones de entrada o salida de los trabajadores.</p> <p>El listado puede ser paginado si el resultado es muy extenso (más de 50 accesos).</p> <p>El interés principal del caso de uso es conocer en determinado momento, cuales son las personas que ha entrado a la UCI, por eso, en el reporte se muestran tanto trabajadores como visitantes.</p> <p>El listado posee la propiedad de mostrar una versión imprimible si el fin es enviarlo por correo electrónico (Ver Caso de Uso “<b>Enviar Reporte</b>”).</p>
Referencias:	R6, R8
Precondiciones:	
Poscondiciones:	
Requisitos Especiales:	

Prototipo

Reporte de accesos de trabajadores

Desde: 2/2/2004 Hasta: 2/2/2004 CI: Nombre Tipo de Acceso: Todos Ver

Mostrando un total de 9 accesos

Fecha y Hora	Nombre Completo	Tipo de Acceso
02/02/2004 13:30:00	Manuel Alejandro Gil	Entrada
02/02/2004 13:31:00	Juan Costa	Entrada
02/02/2004 13:35:03	Yudiel Tamayo	Salida
02/02/2004 13:40:00	Jose Fidalgo	Entrada
02/02/2004 13:52:00	Yanier Castellanos	Entrada
02/02/2004 14:34:00	Ronny Gonzalez	Entrada
02/02/2004 15:30:00	Julio Cesar Isaza	Salida
02/02/2004 15:33:00	Renier Perez	Salida
02/02/2004 18:34:03	Irela Izada	Salida

Pantalla 1

### 3.12 Conclusiones

En este capítulo se comenzó el desarrollo de la propuesta de solución, analizándose los procesos de negocio y los requisitos que debe tener el sistema, llegando a los casos de uso necesarios para satisfacer a estos requisitos. Gracias a ello, se puede empezar a construir el sistema, guiándose por los casos de uso.



## CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

### 4.1 Introducción.

En este capítulo se modelan los artefactos que constituyen las clases de cada paquete de la aplicación, y las relaciones entre ellos. Se muestra además, el modelo de datos y diagrama de clases persistentes, para una mejor comprensión de qué parte de la información manipulada es almacenada.

Se exponen las pautas seguidas para el diseño de la capa de presentación de la aplicación, la capa Web. Se muestra además, cómo ha sido elaborada la interfase, la estructura de la ayuda y los reportes realizados.

Además, se especifican los estándares a seguir en la codificación, para lograr una mejor organización y un estilo propio del software y por último se presentará el modelo de implementación, donde se definen los nodos y componentes que conforman la estructura física de la aplicación.

## 4.2 Diagrama de clases

Para una mejor comprensión del diagrama de clases del presente trabajo, se han separado las mismas en 4 paquetes atendiendo a su funcionalidad, dichos paquetes se han dividido a su vez en otros subpaquetes para una mayor organización y legibilidad.

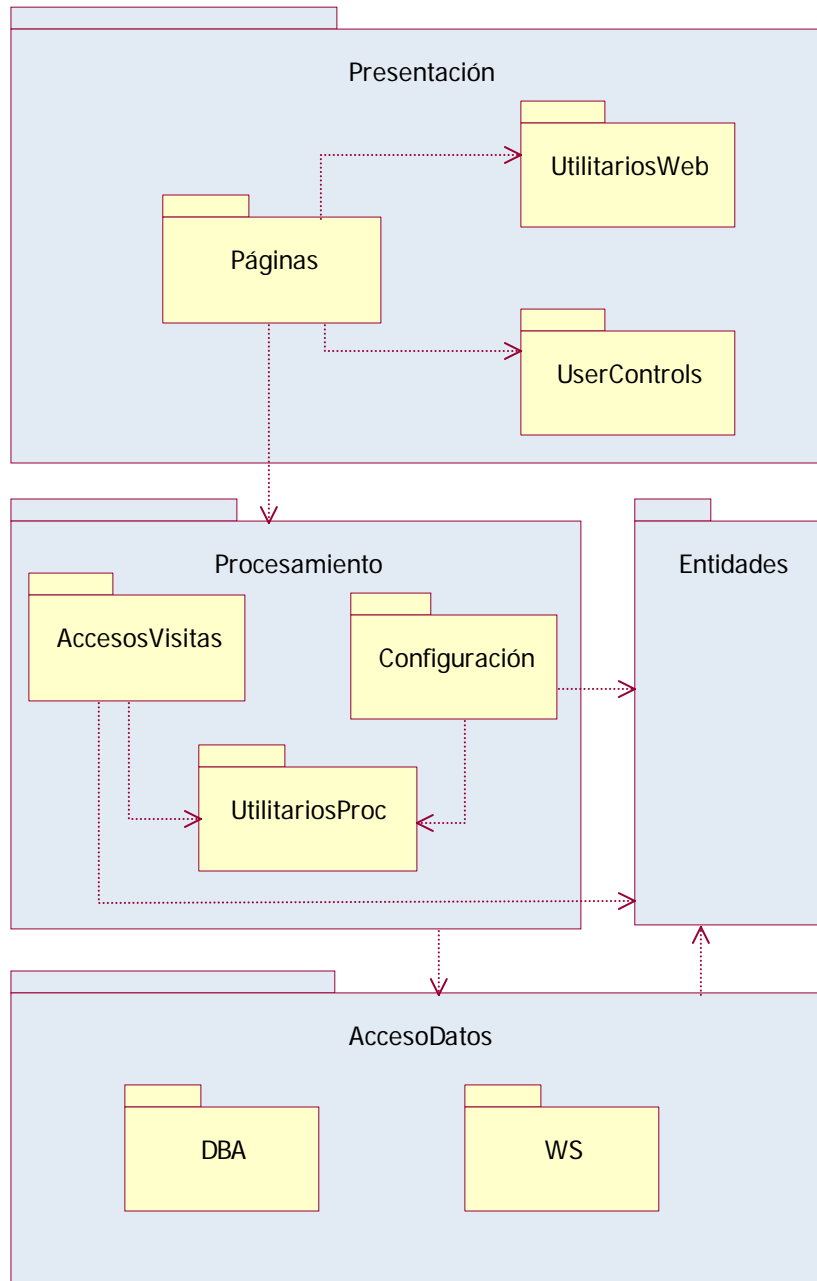


Figura 10. Diagrama de Clases de Diseño



El paquete **AccesoDatos** contiene las clases para hacer posible la persistencia y recuperación de objetos. Está dividido en dos subpaquetes: **WS** que abarca las clases para acceder a Servicios Web de la Intranet, necesarios para obtener diversas informaciones sobre personas y áreas de la UCI. El subpaquete **DBA**, contenido también dentro del paquete *AccesoDatos*, contiene las clases encargadas de acceder a la base de datos para manipular la persistencia de las entidades. El Paquete *AccesoDatos* en general permite a la aplicación abstraerse del origen de los datos y de la lógica de su persistencia, logrando un bajo acoplamiento entre sus componentes.

El paquete **Procesamiento** contiene la lógica de negocio de la aplicación. Está dividido en 3 subpaquetes: **AccesosVisitas**: Contiene las clases controladoras de las entidades relacionadas con los accesos y las visitas principalmente; el subpaquete **Configuración** que contiene clases controladoras para manipular la administración del sistema en general, definir nomencladores, usuarios, niveles de acceso, etc. Por último, el subpaquete **UtilitariosProc** contiene clases utilitarias para ayudar en diversas funciones de todo el paquete *Procesamiento* como acceso al dominio de red, validación de datos, envío de correo electrónico, etc.

El paquete **Entidades** contiene clases que no tienen comportamiento, sólo propiedades y son representaciones de entidades reales del dominio, la mayoría son clases persistentes que son accedidas por las clases de los paquetes *Procesamiento* y *AccesoDatos*.

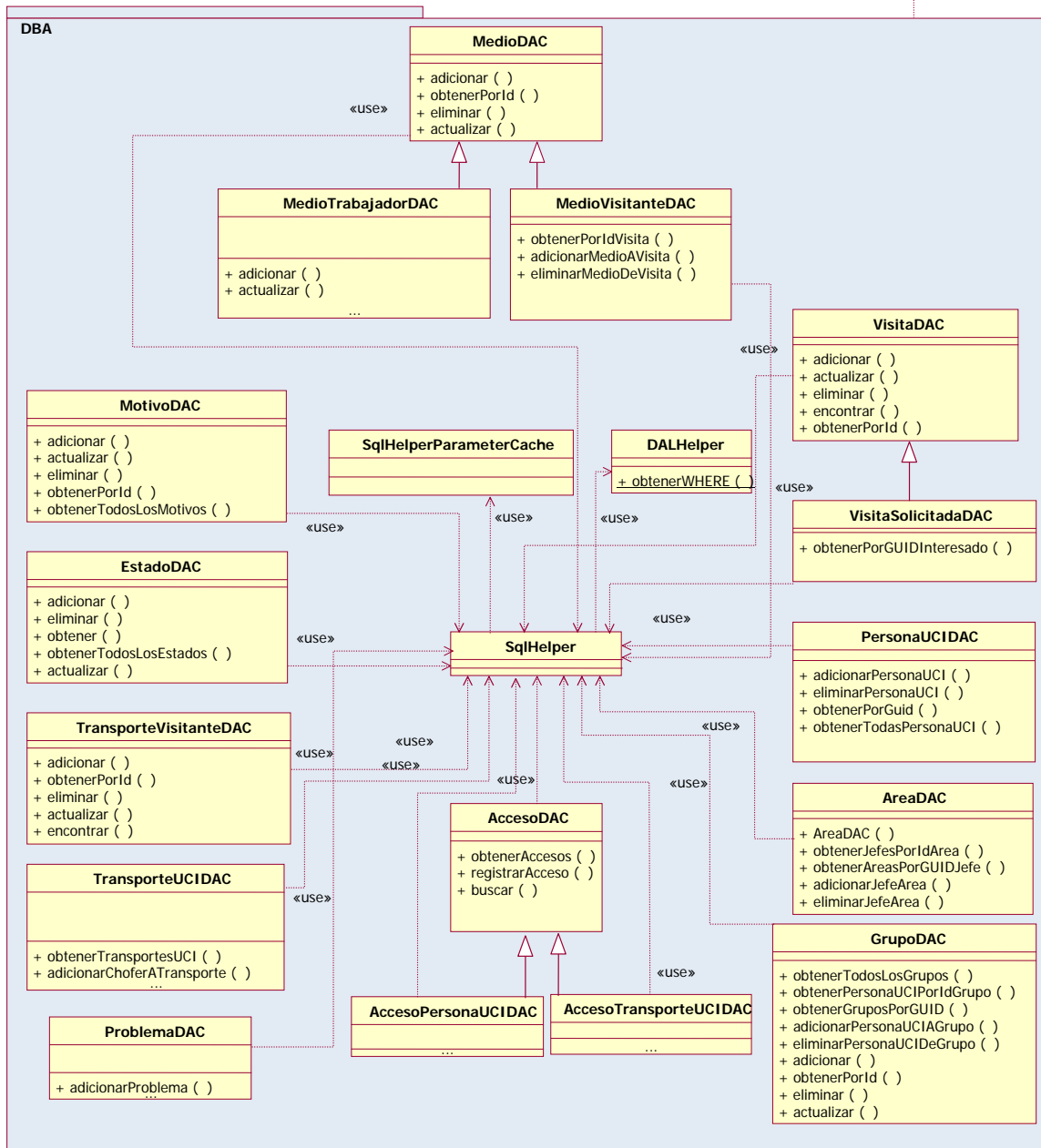
Por último, el paquete **Presentación** contiene las clases de presentación del sistema, en este caso, una interfaz web. Este paquete está dividido en 3 subpaquetes: **Configuración** que contiene las páginas relacionadas con la administración del sitio; el subpaquete **Páginas** que contiene las páginas para la gestión de accesos, visitas, reportes, consultas, etc. El subpaquete **UtilitariosWeb** contiene clases utilitarias para desarrollar ciertas funciones frecuentes, como validación, control de flujo, etc. Por último, el subpaquete **UserControls** contiene los “Controles de usuario”, concepto de la plataforma .NET para definir porciones de código que pueden reutilizarse en diversas páginas, forman una especie de librería de elementos comunes a varias páginas.

Esta división por paquetes obedece a la arquitectura dividida por capas en que se ha diseñado la aplicación y también a la funcionalidad de las clases, las cuales han sido agrupadas de esta forma para lograr mayor desacoplamiento, reutilización y legibilidad de los diagramas.

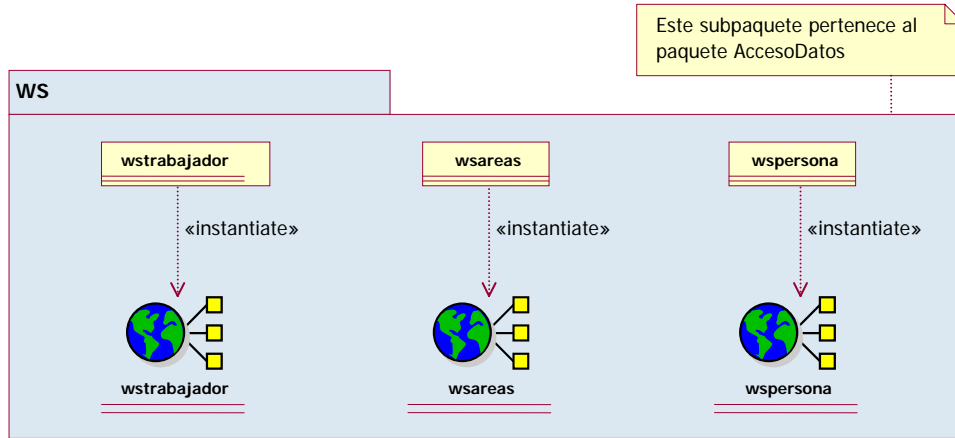
## Paquete AccesoDatos

### Subpaquete DBA

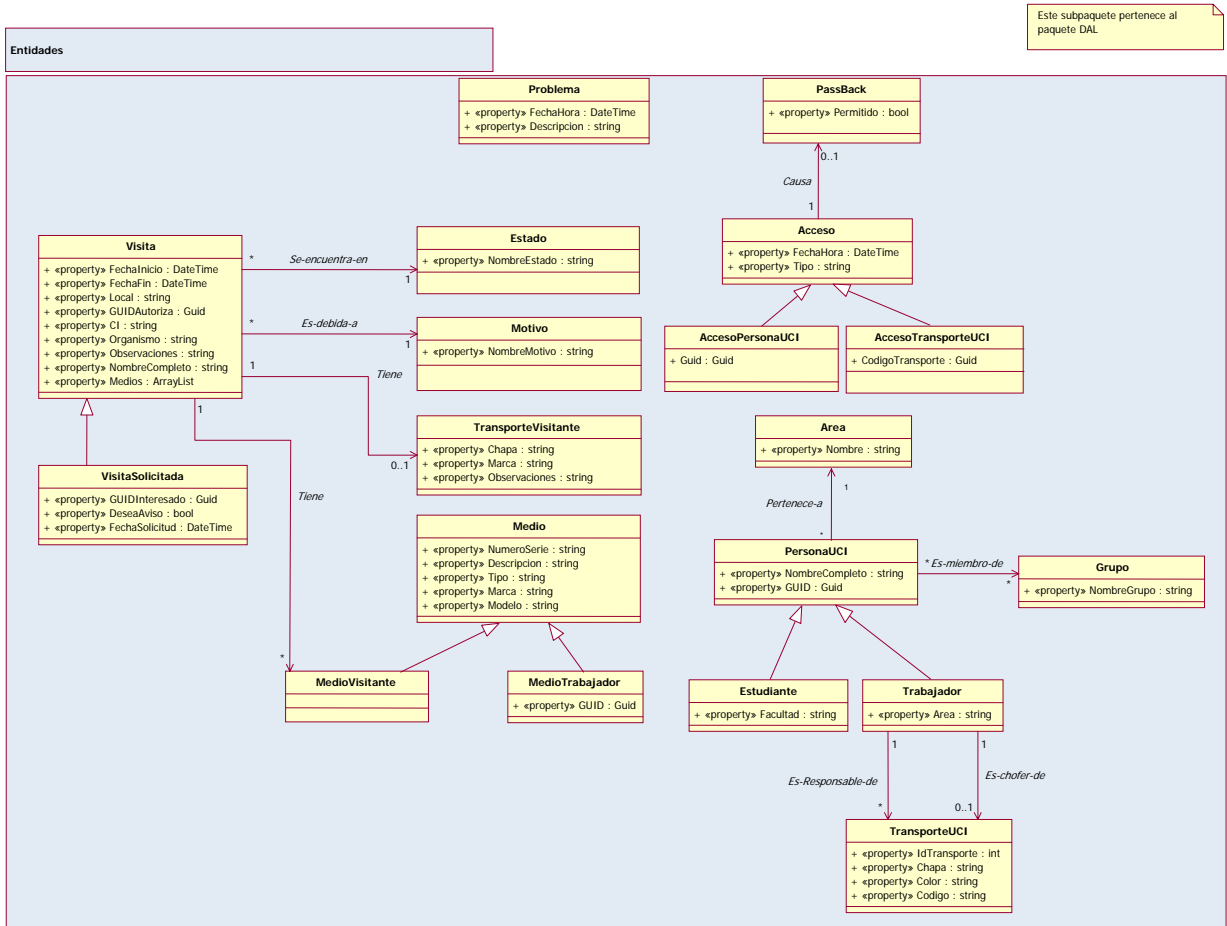
Este subpaquete se encuentra en el paquete AccesoDatos



Subpaquete WS

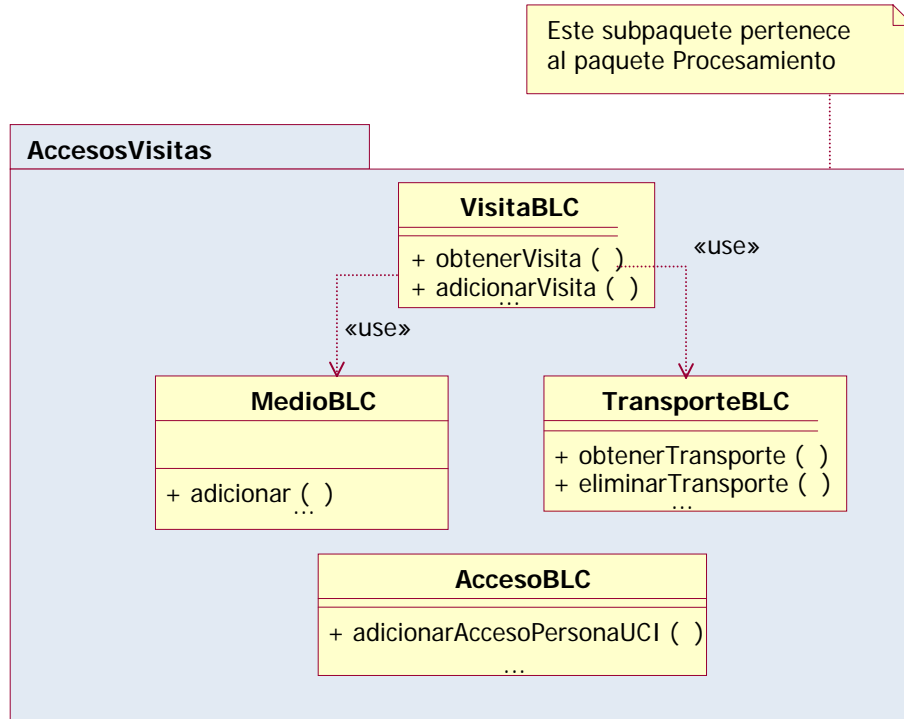


**Paquete Entidades**

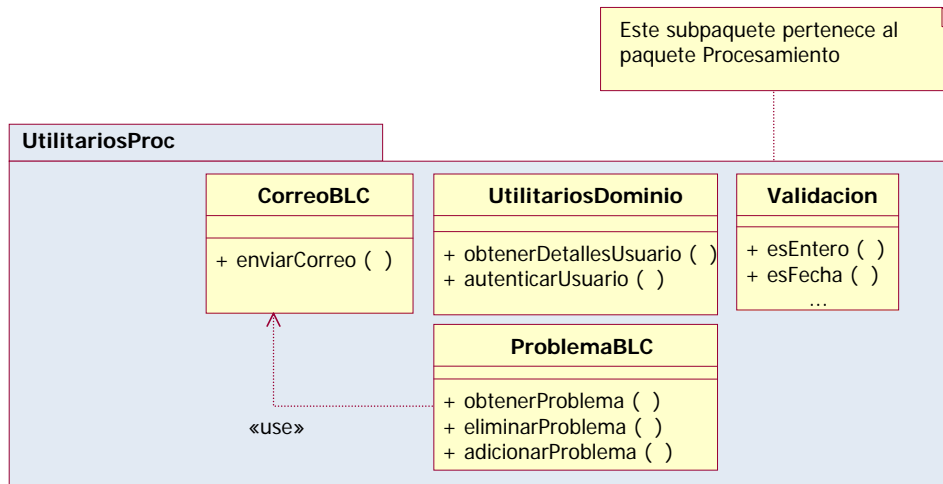


### Paquete Procesamiento

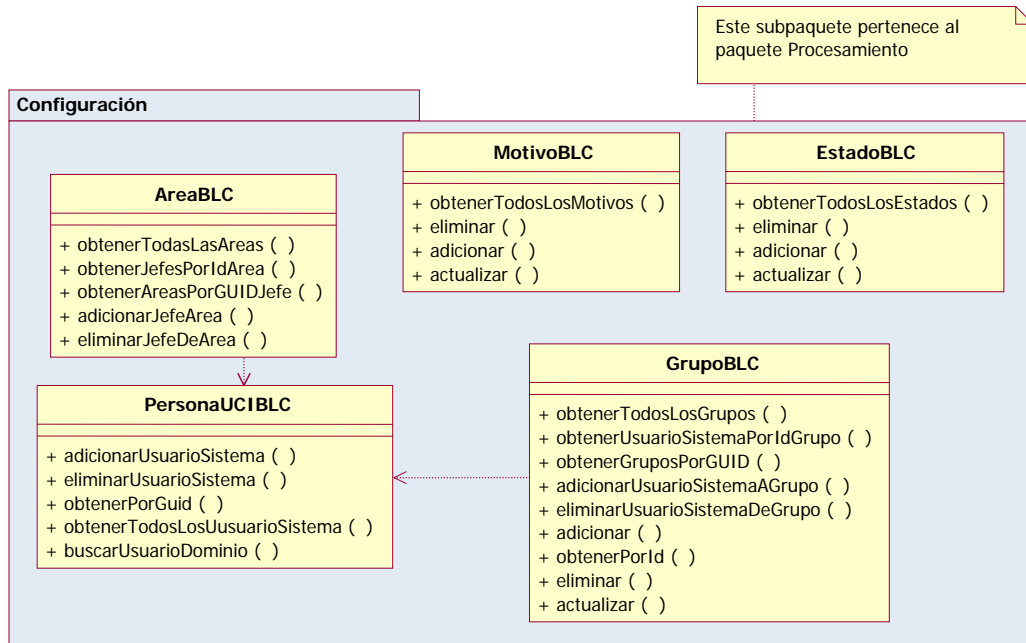
#### Subpaquete **AccesosVisitas**



#### Subpaquete **UtilitariosProc**

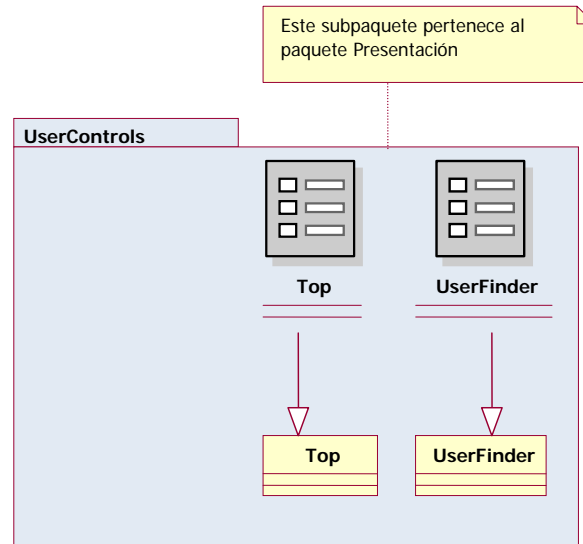


### Subpaquete Configuración

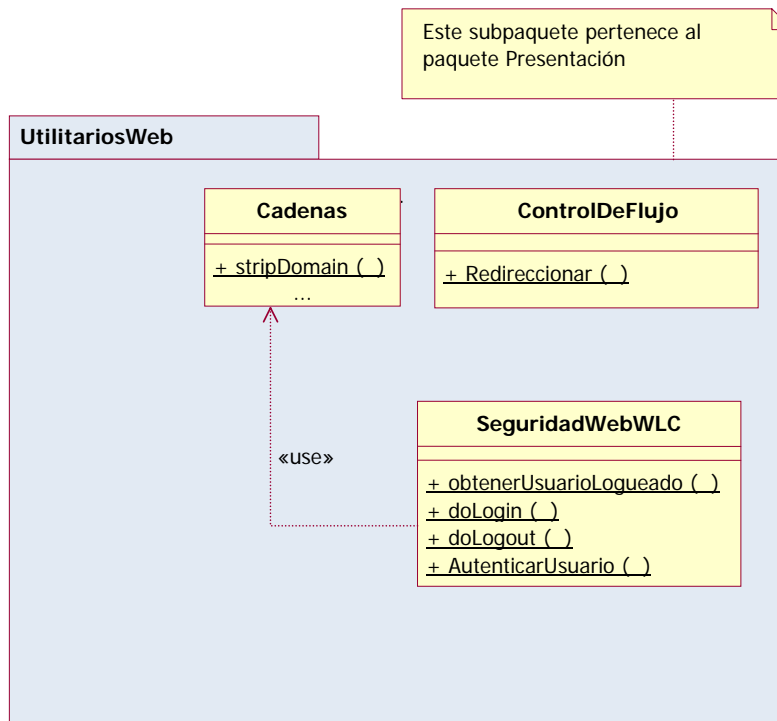


## Paquete Presentación

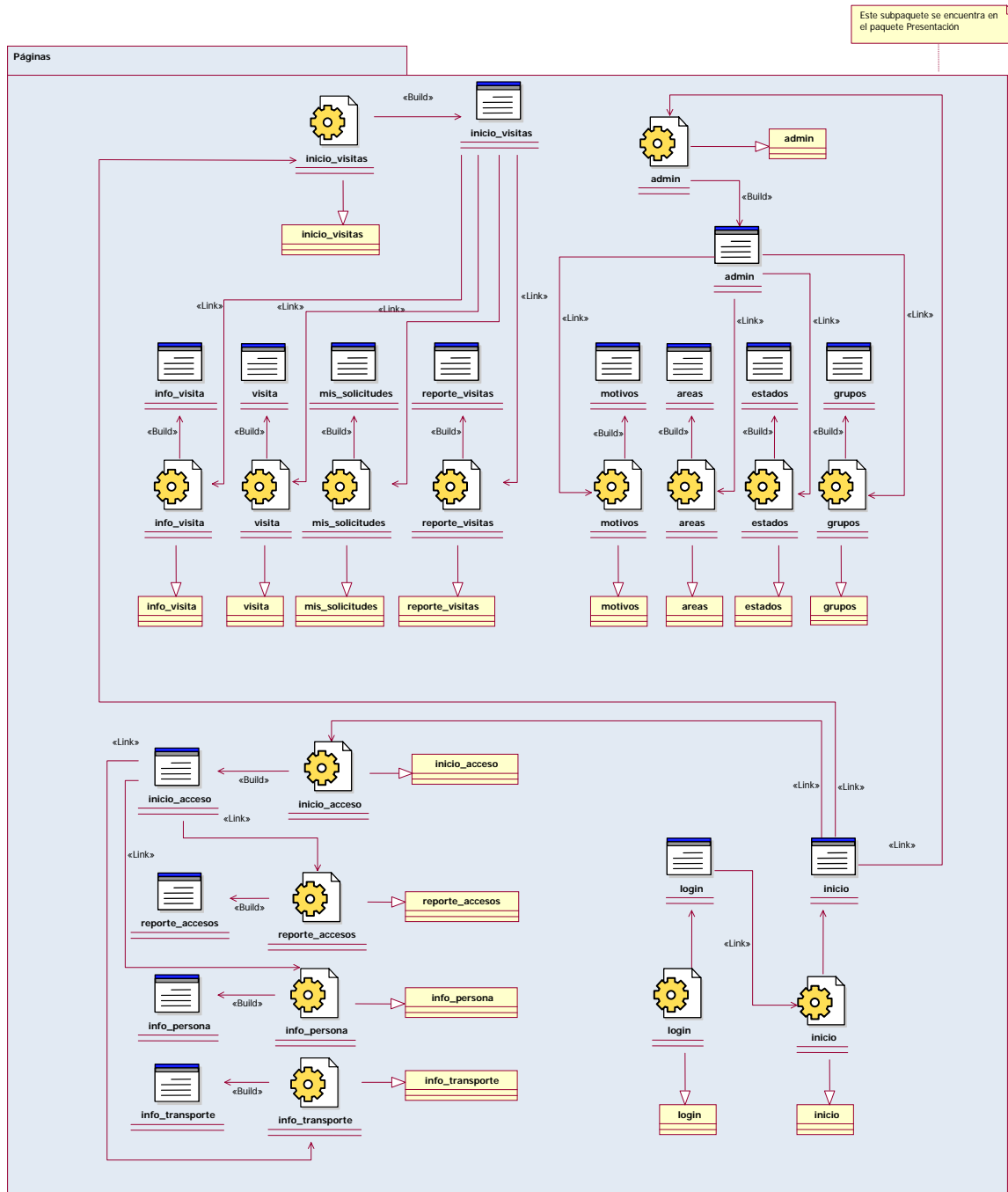
### Subpaquete **UserControls**



### Subpaquete **UtilitariosWeb**



Subpaquete Páginas





### 4.3 Diseño de la base de datos

Para asistir en la construcción de la base de datos, se utiliza el diagrama de clases persistentes y el modelo de datos, este último se corresponde con la representación física de la misma.

#### 4.3.1 Diagrama de clases persistentes

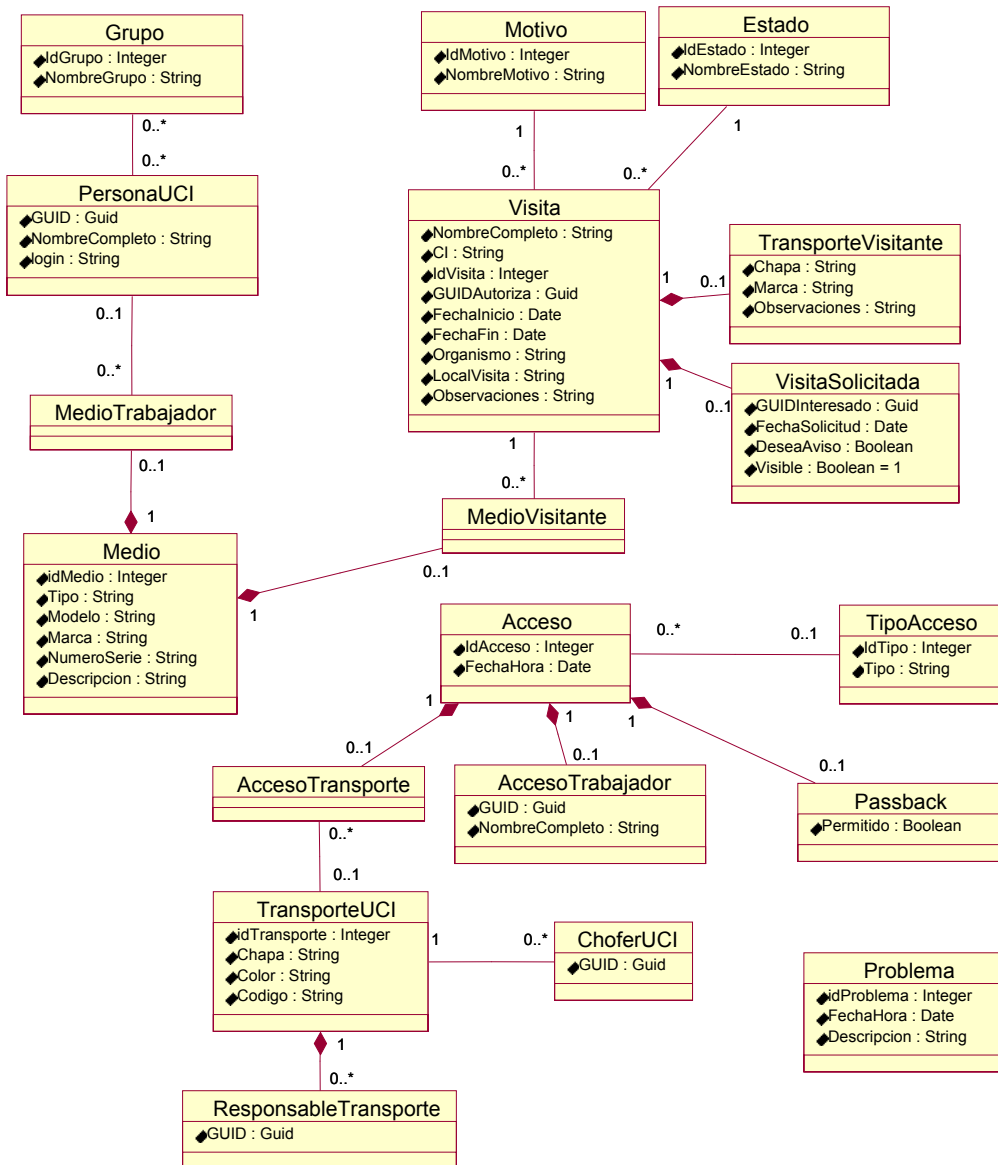


Figura 11. Diagrama de Clases Persistentes

### 4.3.2 Modelo de datos

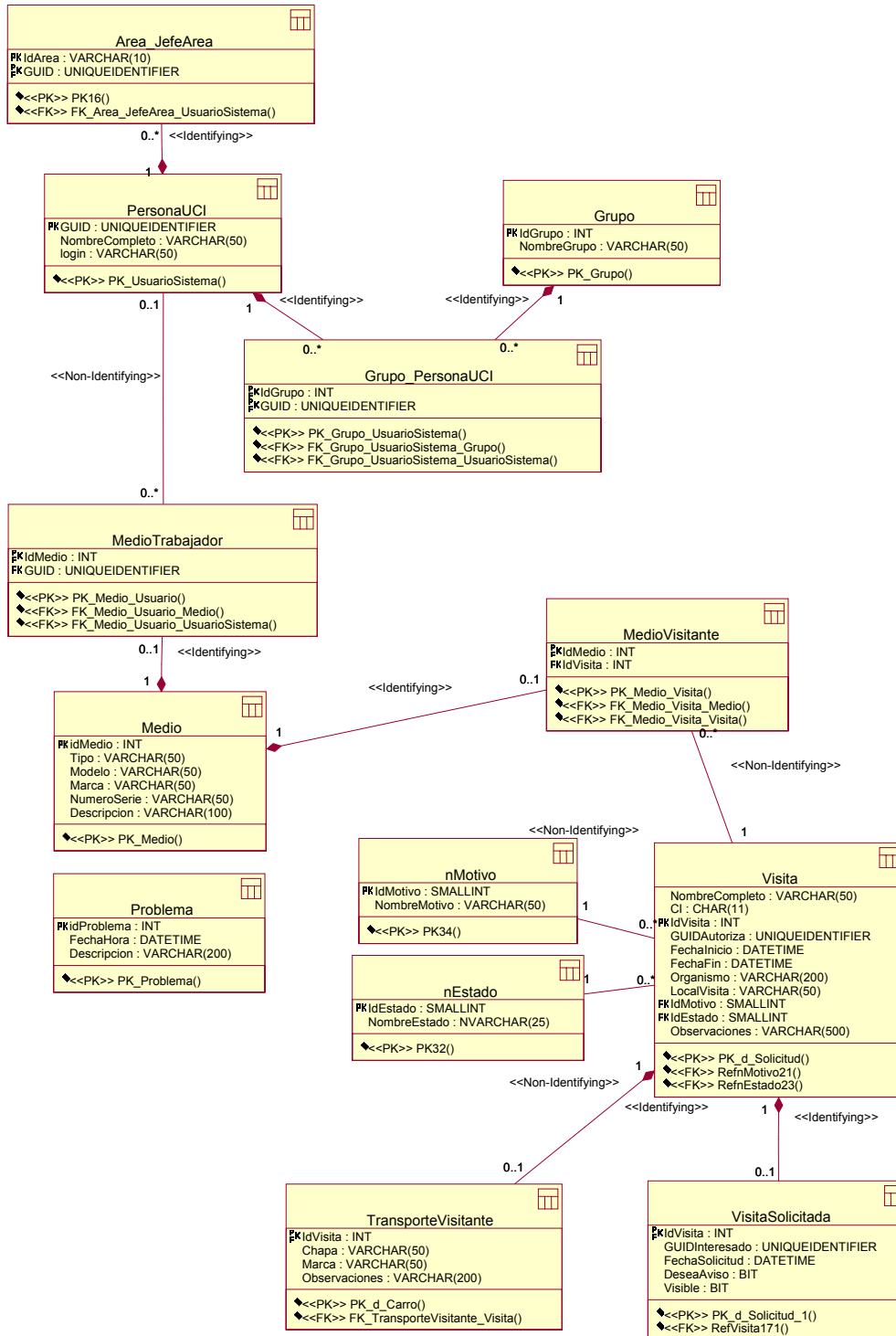


Figura 12. Modelo de Datos

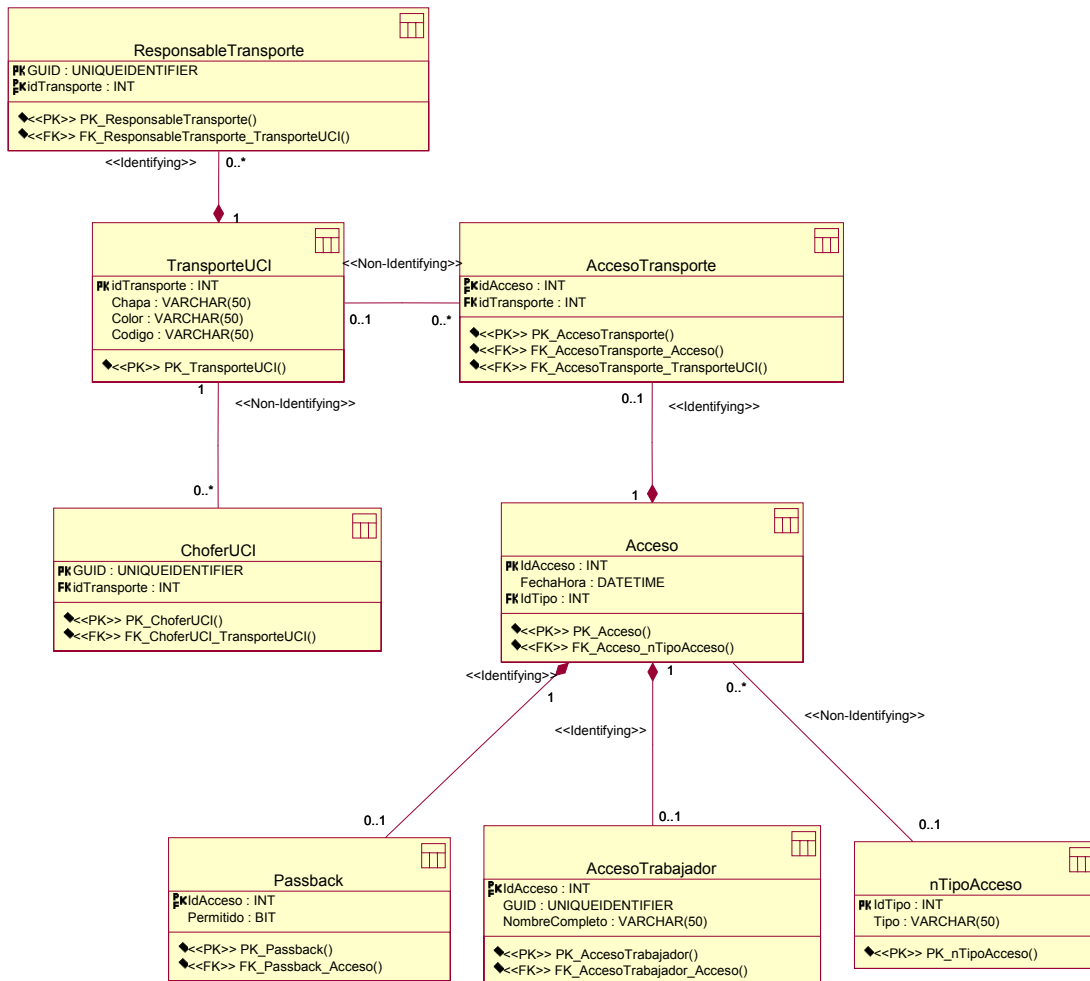


Figura 13. Modelo de Datos. (cont.)

## 4.4 Principios de diseño

El diseño ha sido elaborado pensando en los usuarios finales, que serán: estudiantes, profesores y trabajadores, los cuales, no en todos los casos poseen conocimientos de computación, por tanto, se ha elegido una interfaz amigable e intuitiva. Se ha mantenido un diseño consistente en todas las páginas, para lograr que el usuario se sienta cómodo y logre acostumbrarse rápidamente a la aplicación.

### 4.4.1 Estándares de la Interfaz de la aplicación

Todas las páginas Web tienen una estructura similar, contienen una imagen identificativa del sistema, además de opciones comunes a todos los usuarios. (Figura 14, A)

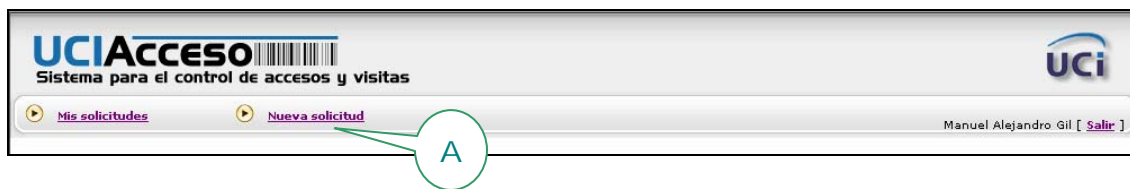


Figura 14. Imagen en la parte superior de todas las páginas

Los colores utilizados son en su mayoría los tonos de gris, azul, y negro. Se ha utilizado azul porque es un color relajante, e identifica al logotipo de la Universidad. El gris es un color neutral, provoca la sensación de estabilidad y orden, razón por la cual se utiliza como fondo en la mayoría de aplicaciones para Windows y otros sistemas operativos.

Se utiliza el rojo para resaltar errores de campos requeridos, con formato incorrecto, o mensajes de operaciones no válidas. (Figura 15, A)

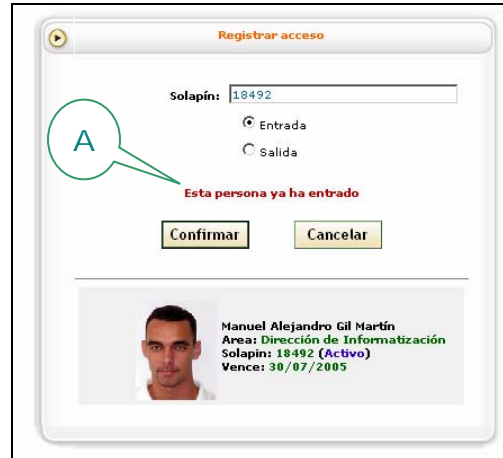


Figura 15. Uso del color rojo para mensajes de error

En general se realizan múltiples operaciones en cada página, de forma que el usuario no tenga que moverse tanto dentro de la aplicación, para completar una operación. Por ejemplo, se puede hacer la inserción, actualización y eliminación en las páginas donde se muestran listados.

Se le piden a los usuarios solamente los datos completamente necesarios, y no lo que pueda ser calculado o inferido en el sistema, como fechas, por ejemplo. Esto minimiza el margen de error.

#### 4.4.2 Formato de Reportes

El sistema brinda varios reportes en forma de tabla, y dos con información sobre personas y transportes. En los reportes en forma de tabla, se da la posibilidad de ordenar por cualquier campo (*Figura 16, A*), para facilitar la búsqueda de información, también se da la posibilidad de filtrar los resultados atendiendo a rangos de fecha, o a cierto valor especificado de un campo. Las filas de los reportes son de colores alternos, para facilitar la lectura, se han utilizado colores grises claros, resaltando los nombres de los campos con fondo azul y letras blancas.

Los reportes permiten paginado, de forma que por cada búsqueda sean visibles solamente un número limitado de registros, permitiendo moverse adelante y hacia atrás, gracias a pequeñas flechas en la parte inferior de las tablas (Figura 4.3 B).

Reporte de accesos de trabajadores

Desde: 2/2/2004 Hasta: 2/2/2004 CI: Nombre Tipo de Acceso: Todos Ver

Mostrando un total de 9 accesos

Fecha y Hora	Nombre Completo	Tipo de Acceso
02/02/2004 13:30:00	Manuel Alejandro Gil	Entrada
02/02/2004 13:31:00	Juan Costa	Entrada
02/02/2004 13:35:03	Yudiel Tamayo	Salida
02/02/2004 13:40:00	Jose Fidalgo	Entrada
02/02/2004 13:52:00	Yanier Castellanos	Entrada
02/02/2004 14:34:00	Ronny Gonzalez	Entrada
02/02/2004 15:30:00	Julio Cesar Isaza	Salida
02/02/2004 15:33:00	Renier Perez	Salida
02/02/2004 18:34:03	Irela Izada	Salida

Figura 16. Ejemplo de reporte

#### 4.4.3 Concepción general de la ayuda

El sistema cuenta con un manual de usuario que esta disponible desde cualquier página del mismo, en dicho manual, se explica su funcionamiento, requerimientos, y se muestra una lista con preguntas frecuentes y sus respuestas.

#### 4.4.4 Tratamiento de excepciones

Para prevenir errores por parte del usuario, sólo se le brindan las opciones mínimas necesarias, a la hora de efectuar cualquier operación, por ejemplo, se deshabilitan ciertos botones si el usuario no tiene que utilizarlos en ese momento.

Mediante una combinación de validación en el lado del cliente y en el lado del servidor, se garantiza que los datos suministrados por los usuarios, se almacenen

íntegros y no existan inconsistencias. Se verifican los campos obligatorios, y se revisa el tipo de datos (Figura 17, A y B).

Todos los mensajes de error se muestran en color rojo para que resalten.

Se manipulan las excepciones que podrían ocurrir en tiempo de corrida de la aplicación, generándose un registro en la base de datos del sistema, para una eventual revisión por parte del administrador.

The image shows a web form titled "Datos del visitante" with the following fields and values:

Field	Value	Validation Status
Nombre y Apellidos	Manuel Alejandro	Valid
CI	800208241A	Invalid (CI no válido)
Organismo		Valid

Callout 'A' points to the red error message "CI no válido". Callout 'B' points to a red asterisk on a dropdown menu in the background form.

Figura 17. Verificación de campos requeridos y en formato correcto

#### 4.5 Estándares de codificación

Resulta ventajoso utilizar un estándar para escribir código, entre dichas ventajas, tenemos las siguientes:

1. Reducir errores
2. Escribir un código comprensible y fácil de leer
3. Garantizar una buena comunicación entre los programadores del equipo
4. Facilitar el mantenimiento del software

En esta aplicación se ha utilizado el estándar de codificación "Camel Case" que principalmente tiene que ver con la capitalización de los caracteres. Así mismo, se ha seguido el estilo de codificación propuesto por Microsoft para programar con C# (Figura 18).

```
...  
  
public void eliminarJefeArea(string IdArea, Guid GUID)  
{  
    SqlParameter[] par = new SqlParameter[10];  
  
    par[0] = new SqlParameter("@IdArea", IdArea);  
    par[1] = new SqlParameter("@GUID", GUID);  
  
    SqlConnection connAccesoVisitas = SqlHelper.getConnection();  
  
    SqlHelper.ExecuteNonQuery(conn, CommandType.StoredProcedure,  
"Area_JefeAreaDelProc", par);  
  
    SqlHelper.closeConnection(ref connAccesoVisitas);  
}  
  
...
```

Figura 18. Ejemplo de codificación utilizada

Se ha tenido especial cuidado para nombrar clases, variables, y demás elementos, precediendo cada nombre con un prefijo para su fácil identificación, así, tenemos los más frecuentes en este sistema:

- Campos de edición: **txt**Nombre.
- Campos de selección: **lbox**Nombre o **chk**Nombre
- Campos ocultos: **hdn**Nombre.
- Botones de acción: **btn**Nombre.
- Variables de control de ciclos: **i, j, k**
- Datagrids: **dtg**Nombre
- Datasets: **ds**Nombre
- Datareaders: **dr**Nombre
- Conexión: **conn**Nombre

En el diseño de la base de datos, las tablas se han nombrado igual que la entidad que almacenan. Los campos de estas tablas, siguen la misma regla, y están nombrados igual que las propiedades de las entidades.



## 4.6 Modelo de despliegue

El modelo de despliegue contiene los nodos que conforman la topología de hardware sobre la que se ejecuta el sistema.

El sistema se ha construido siguiendo la arquitectura de tres capas, o sea, la separación de los componentes en Lógica de Negocio, Acceso a Datos, Cliente. Dentro de la capa de acceso a datos, se han separado físicamente las librería para acceder a la base de datos y la librería para acceder a los WebServices, esto es con el fin de lograr una independencia entre ambas y poder reemplazar una u otra si cambia el entorno de la aplicación.

Existe un servidor SQLServer 2000 que contiene la base de datos. El servidor Web reside en un servidor separado, contiene al Internet Information Server, tiene instalada la plataforma .NET 1.0 ó 1.1. Este servidor también contiene el módulo para enviar reportes por correo electrónico, el módulo es un ejecutable independiente, por lo que se puede instalar incluso en un servidor separado, es ejecutado por el sistema operativo una vez al día. Los clientes son los distintos usuarios desde sus terminales, accediendo al sistema mediante un navegador.

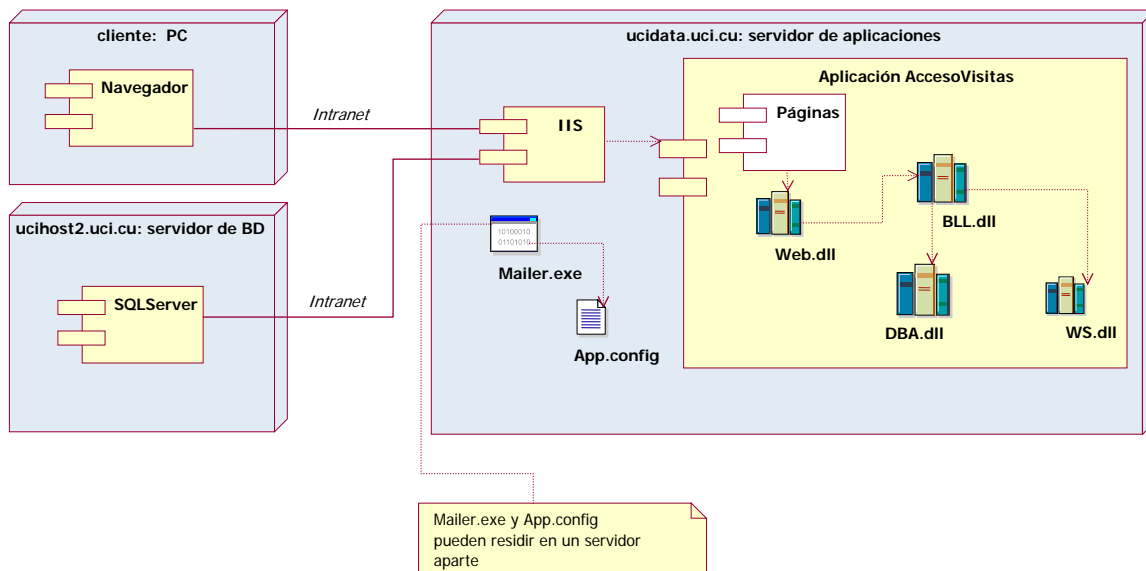


Figura 19. Diagrama de Despliegue

### 4.7 Modelo de Componentes

#### Paquete AccesoDatos

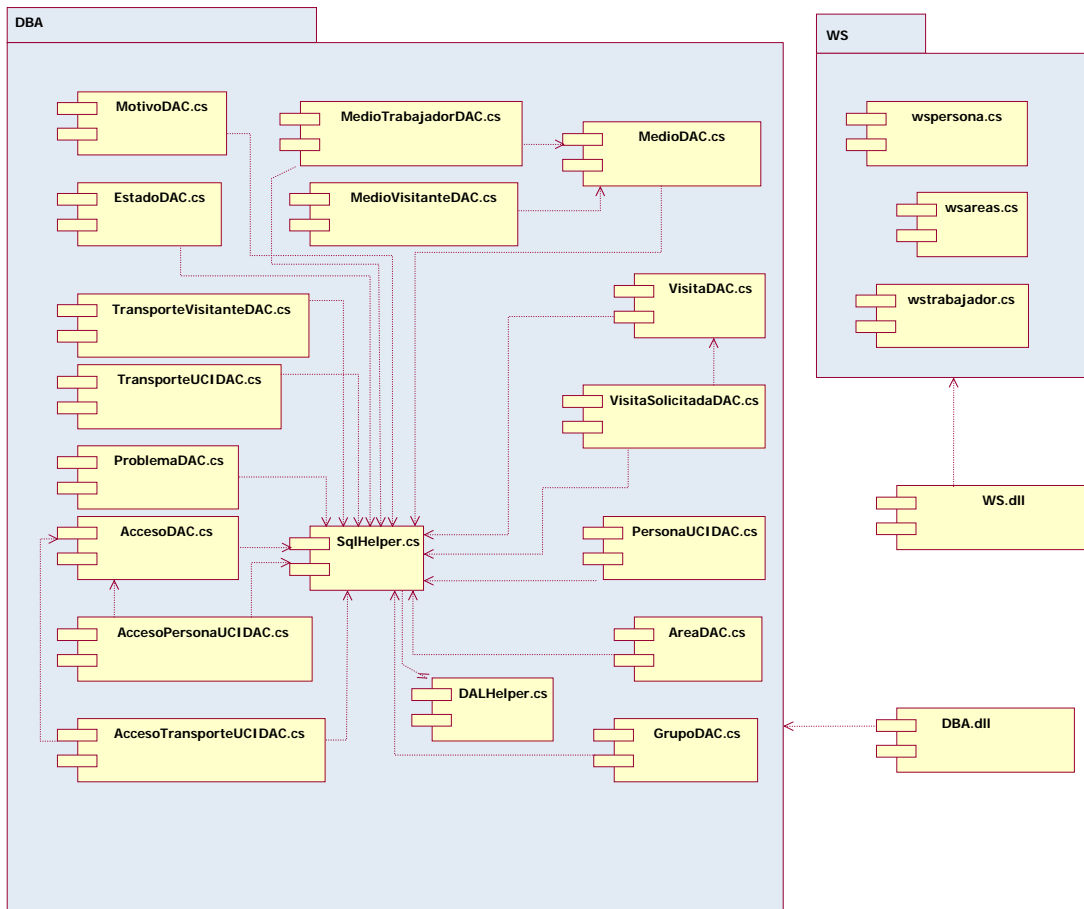


Figura 20. Diagrama de Componentes

### Paquetes Entidades y Procesamiento

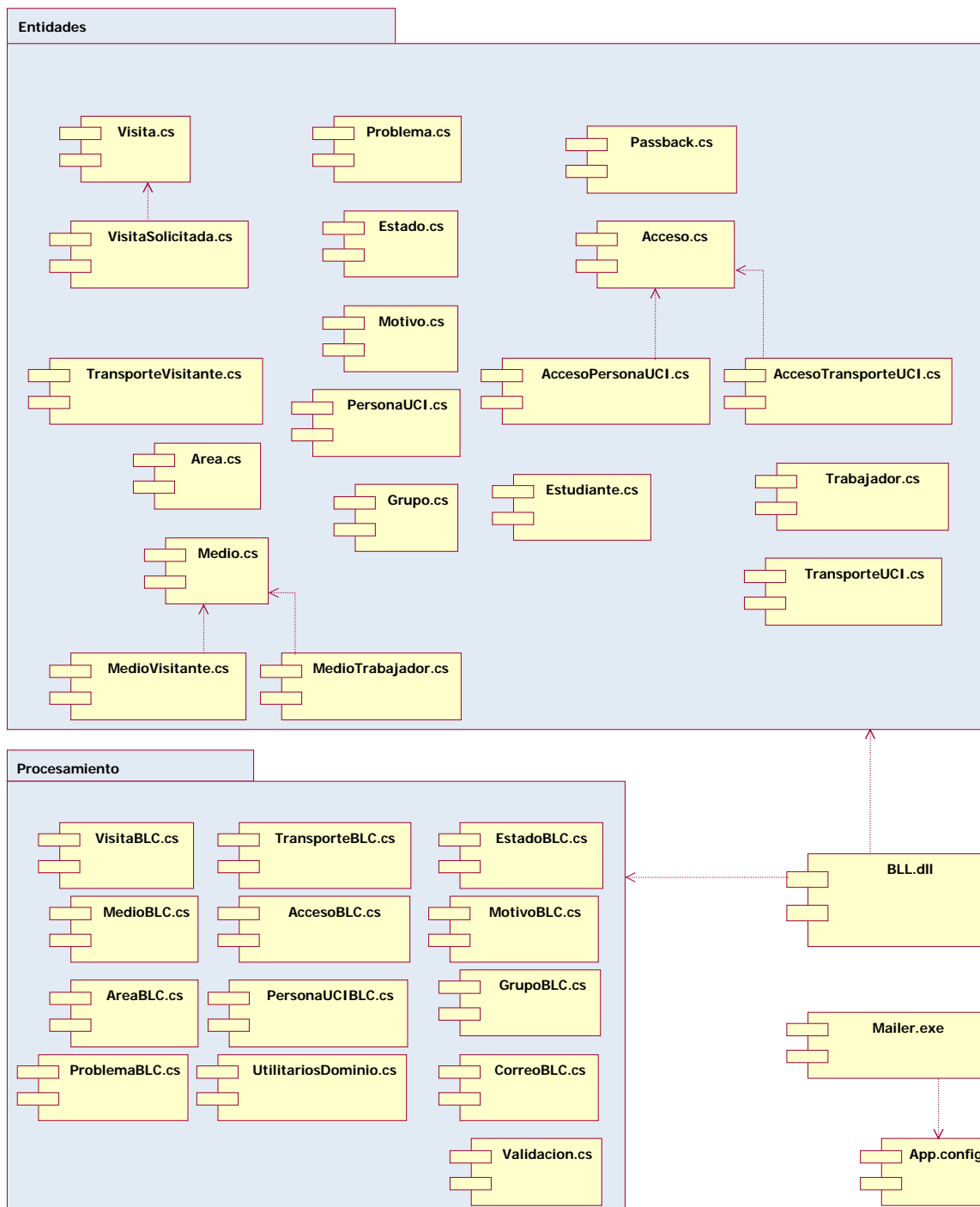


Figura 21. Diagrama de Componentes. (cont.)

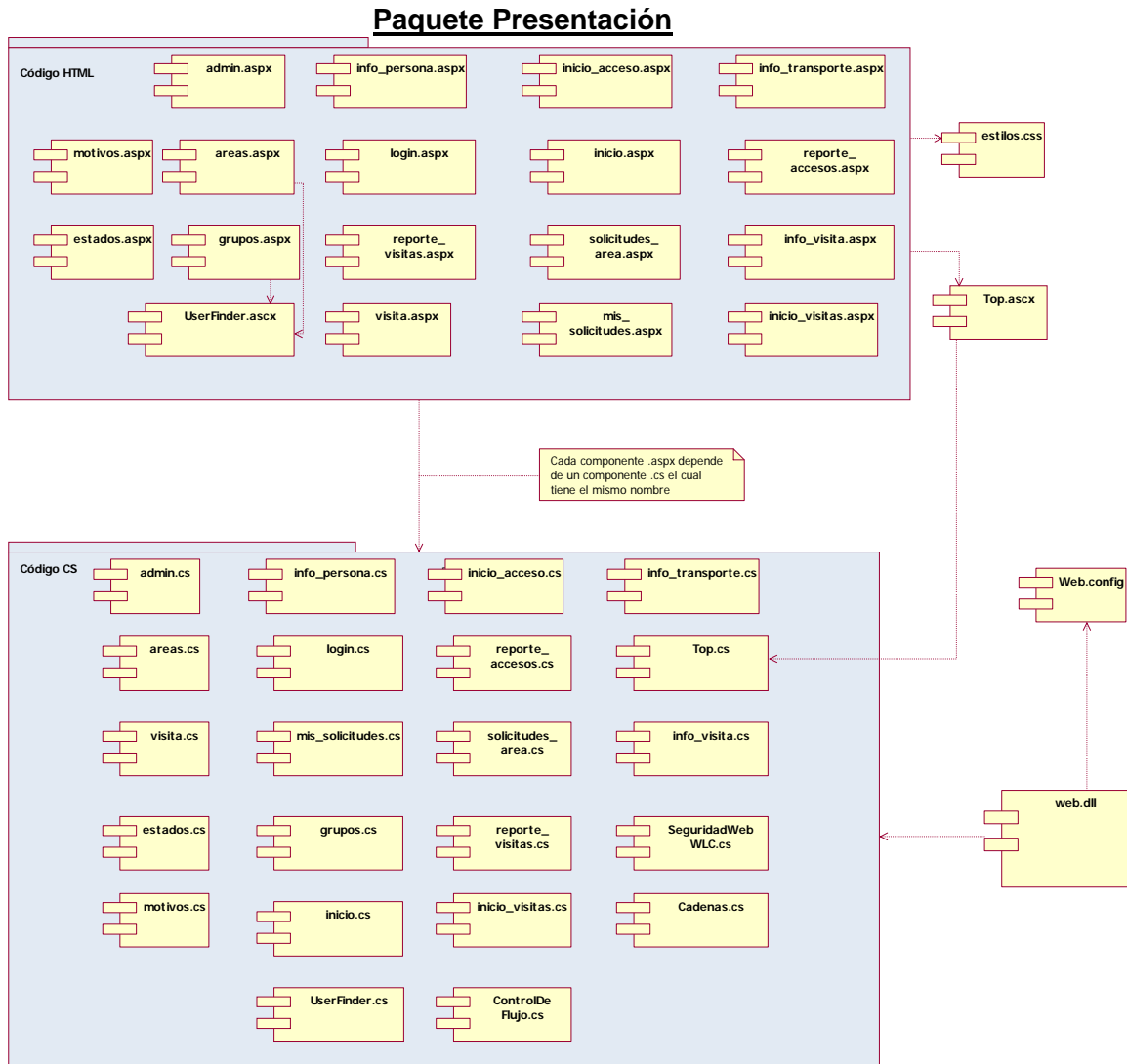


Figura 22. Diagrama de Componentes. (cont.)

#### 4.7.1 Explicación de los componentes

Para una mayor claridad en las explicaciones de cada componente, se seguirá el mismo orden en que se han presentado los paquetes. Con el objetivo de lograr una mayor granularidad, limpieza y rapidez al encontrar las clases, se implementó cada una en un fichero separado, con el mismo nombre, por tanto, cada componente tiene el mismo nombre de la clase que contiene, y las interfaces que este expone, son los mismos métodos públicos de las clases, si desea información sobre el contenido e interfaces de los componentes, remítase al diagrama de clases del paquete correspondiente. Los componentes que cumplen esta regla son los de los siguientes paquetes:

- **Paquete DBA**

En este paquete se encuentran los componentes que contienen las implementaciones de clases que se ocupan de la persistencia en base de datos de las ciertas entidades del sistema.

- **Paquete WS**

En este paquete se encuentran los componentes que contienen las implementaciones de clases que se ocupan de la recuperación de datos accediendo a WebServices de la Intranet.

- **Paquete Entidades**

En este paquete se encuentran los componentes que contienen las implementaciones de las clases que representan las entidades del sistema.

- **Paquete Procesamiento**

En este paquete se encuentran los componentes que contienen las implementaciones de clases controladoras de entidades.

- **Paquete Código CS**

Este paquete contiene el código fuente de las clases utilizadas por las páginas ASPX, los componentes se llaman igual que las páginas que dependen de ellos.

Existen otros componentes que no cumplen con la regla de nomenclatura antes mencionada, estos son:

Componente	Propósito	Contenido	Interfaces
DBA.dll	Librería necesaria para el acceso a datos desde una BD.	Implementación de las clases del paquete DBA. <i>Ver Diagrama de Componentes de dicho paquete.</i>	<i>Ver Diagrama de Clases del paquete DBA.</i>
WS.dll	Librería necesaria para la recuperación de datos desde WebServices de la Intranet.	Implementación de las clases del paquete WS. <i>Ver Diagrama de Componentes de dicho paquete.</i>	<i>Ver Diagrama de Clases del paquete WS.</i>
BLL.dll	Librería necesaria para controlar la lógica de negocio del sistema.	Implementación de las clases de los paquetes Entidades y Procesamiento. <i>Ver Diagrama de Componentes de dichos paquetes.</i>	<i>Ver Diagrama de Clases de los paquetes Entidades y Procesamiento.</i>
Mailer.exe	Ejecutable del módulo de envío de correo electrónico.	-	<i>Ver Diagrama de Clases del paquete UtilitariosProc.</i>
App.Config	Archivo de configuración para la aplicación del envío de correo electrónico (Mailer.exe).	Este componente es un XML con las direcciones de los reportes a enviar, destinatarios, etc.	-
Web.dll	Librería necesaria para el funcionamiento de las páginas sitio Web.	Contiene la implementación de las clases del paquete Presentación.	<i>Ver Diagrama de Clases del paquete Presentación.</i>
Web.config	Archivo de configuración del sitio web.	Este componente es un XML con valores globales necesarios en diversas páginas.	-
Top.ascx	UserControl, es un tipo de componente de .NET que es reutilizado en todas las páginas.	Contiene la imagen y elementos de navegación comunes a todas las páginas.	<i>Ver CU Configurar Sistema, aunque está disponible en todos los casos de uso.</i>
UserFinder.ascx	UserControl, es un tipo de componente de .NET que es reutilizado algunas páginas. Su propósito es brindar un control reutilizable para la búsqueda de personas de la UCI.	Un botón que despliega una página la cual permite hacer búsquedas de personas del dominio.	<i>Ver CU Configurar Sistema</i>
Estilos.css	Hoja de estilos del sitio.	Contiene los estilos definidos para el sitio Web.	-

Páginas Web

Componente	Propósito	Contenido
admin.aspx	Página principal de administración.	<i>Ver CU Administración.</i>
motivos.aspx	Página para la configuración de motivos de visitas.	<i>Ver CU Administración.</i>
estados.aspx	Página para la configuración de estados de visitas.	<i>Ver CU Administración.</i>
areas.aspx	Página para configurar los directivos de las áreas.	<i>Ver CU Administración.</i>
grupos.aspx	Página para configurar los grupos del sistema.	<i>Ver CU Administración.</i>
info_persona.aspx	Página que brinda información de una persona que pertenezca a la UCI.	<i>Ver CU Obtener Información Persona.</i>
info_transporte.aspx	Página que brinda información de un vehículo que pertenezca a la UCI.	<i>Ver CU Obtener Información Transporte.</i>
login.aspx	Página para autenticar usuarios.	<i>Ver CU Autenticar Usuario</i>
reporte_visitas.aspx	Página para brindar reporte de visitas	<i>Ver CU Mostrar reporte de Acceso de Personas.</i>
visita.aspx	Página que brinda las operaciones de CRUD de visitas (ver Glosario).	<i>Ver CU Manipular Solicitudes.</i>
inicio_acceso.aspx	Página de inicio para la sección de Control de Accesos.	<i>Contiene enlaces a las distintas páginas relacionadas con Accesos.</i>
inicio_visitas.aspx	Página de inicio para la sección de Visitas	<i>Contiene los enlaces a las distintas páginas relacionadas con las visitas.</i>
inicio.aspx	Página de inicio del sitio.	<i>Contiene una bienvenida y breve explicación del sitio, con enlaces a las distintas áreas, según el nivel del usuario.</i>
solicitudes_area.aspx	Muestra las solicitudes efectuadas, de determinada área.	<i>Ver CU Aprobar o Denegar Solicitud.</i>
mis_solicitudes.aspx	Muestra las solicitudes efectuadas por un usuario del sistema	<i>Ver CU Manipular solicitudes.</i>
reporte_acceso.aspx	Muestra un reporte de los accesos en un rango de fecha.	<i>Ver CU Mostrar Reporte de Accesos de Personas.</i>
info_visita.aspx	Muestra la información de una visita.	<i>Ver CU Aprobar o Denegar Solicitud.</i>

## 4.8 Conclusiones

En este capítulo se ha llevado a cabo la descripción de las clases y demás elementos necesarios para la implementación. Se obtuvo el diagrama de clases del sistema, separado por paquetes. Se definieron, a partir del mismo, cuáles son las clases que serán persistentes, luego, a partir de esto, se construyó el modelo de datos. Se expusieron las pautas seguidas para el diseño de la interfaz, y se explicó cómo está estructurada la aplicación físicamente, mediante los modelos de despliegue y de componentes.



Capítulo  
5

## ESTUDIO DE FACTIBILIDAD

### 5.1 Introducción

Es importante evaluar la factibilidad de un proyecto antes de su elaboración, para conocer si es conveniente llevarlo a cabo. La viabilidad y el análisis de riesgo están relacionados de muchas maneras. Si el riesgo del proyecto es alto, la viabilidad de producir software de calidad se reduce. En el presente capítulo se hace un estudio de factibilidad, beneficios y costo del sistema propuesto.

### 5.2 Planificación

Nombre de la entrada externa	Cantidad de Ficheros	Cantidad de Elementos de datos	Clasificación (Simple, Media y complejo)
Registrar entrada o salida de visitante	3	20	Complejo
Registrar entrada o salida de trabajador	3	7	Simple
Adicionar estado de visita	1	1	Simple
Eliminar solicitud	5	24	Complejo
Eliminar estado de visita	1	2	Simple

Adicionar motivo de visita	1	1	Simple
Eliminar motivo de visita	1	2	Simple
Adicionar grupo del sistema	1	2	Simple
Eliminar grupo del sistema	1	2	Simple
Adicionar usuario a grupo	1	3	Simple
Eliminar usuario de grupo	1	3	Simple
Adicionar transporte	1	4	Simple
Adicionar chofer	1	2	Simple
Adicionar responsable	1	2	Simple
Eliminar chofer	1	2	Simple
Eliminar responsable	1	2	Simple
Eliminar transporte	2	5	Medio
Adicionar medio	2	7	Medio
Eliminar Medio	2	7	Medio

**Tabla1. Entradas externas**

Nombre de la salida externa	Cantidad de Ficheros	Cantidad de Elementos de datos	Clasificación (Simple, Media y compleja)
Listado de solicitudes de una persona	1	2	Simple
Listados de solicitudes de un área	1	2	Simple
Detalles de solicitud	3	14	Medio
Listado de estados de visitas	1	1	Simple
Listado de motivos de visitas	1	1	Simple
Listado de grupos del sistema	1	2	Simple
Listado de usuarios del sistema	1	3	Simple
Envío de reporte	2	5	Simple
Listado de transportes	3	6	Complejo

Listado de medios	2	7	Medio
Listado de accesos de trabajadores	2	5	Medio
Listado de visitantes	1	11	Simple

Tabla 2. Salidas Externas.

Nombre de la petición	Cantidad de Ficheros	Cantidad de Elementos de datos	Clasificación (Simple, Media y compleja)
Obtener datos de persona	1	4	Simple
Obtener datos de transporte	1	4	Simple
Validar solapín	1	4	Simple
Buscar solicitud	2	14	Simple

Tabla 3. Peticiones.

Nombre del fichero interno	Cantidad de records	Cantidad de Elementos de datos	Clasificación (Simple, Media y compleja)
Area_JefeArea	1	2	Simple
PersonaUCI	1	3	Simple
MedioTrabajador	1	2	Simple
Medio	1	6	Simple
Problema	1	3	Simple
Grupo_PersonaUCI	1	2	Simple
Grupo	1	2	Simple
MedioVisitante	1	2	Simple
Visita	1	11	Simple
nMotivo	1	2	Simple
nEstado	1	2	Simple
TransporteVisitante	1	4	Simple
VisitaSolicitada	1	5	Simple
ResponsableTransport	1	2	Simple
TransporteUCI	1	4	Simple

Responsable_TransporteUCI	1	2	Simple
ChoferUCI	1	2	Simple
AccesoTransporte	1	2	Simple
Acceso	1	3	Simple
Passback	1	2	Simple
AccesoTrabajador	1	3	Simple
nTipoAcceso	1	2	Simple

Tabla 4. Ficheros Internos.

Elementos	Simples		Medios		Complejos		Subtotal de puntos de función
	No	X Peso	No	X Peso	No	X Peso	
Ficheros lógicos internos	22	7	0	10	0	15	154
Entradas externas	14	3	3	4	2	6	66
Salidas externas	8	4	2	5	1	7	49
Peticiones	4	3	0	4	0	6	12
<b>Total</b>							<b>281</b>

Tabla 5. Interfaces Externas.

### 5.3 Costos

Características	Valor
Puntos de función desajustados	281
Lenguaje	C# (90 %) SQL (10%)
Instrucciones fuentes por puntos de función	(59) (39)
Instrucciones fuentes por lenguaje (miles de instrucciones)	(7,3341) (1,0959)
Instrucciones fuentes (miles de instrucciones)	<b>8,43</b>

Tabla 6. Cantidad de Instrucciones Fuentes.

Factores	Valor	Justificación
PREC	2.48	Se han desarrollado proyectos similares en el pasado.
FLEX	2.03	Debe haber buen cumplimiento de los requerimientos del sistema y de las especificaciones de interfaz externa.
TEAM	0	El equipo que va desarrollar el software es altamente cooperativo.
RESL	2.83	La herramienta de desarrollo seleccionada posee gran tolerancia ante fallos y capacidad de resolución de riesgos que pudieran ocurrir. Se estima que la reducción del riesgo es entorno a un 75%.
PMAT	6.24	El proceso tiene nivel 1 de madurez (alto).

Tabla 7. Factores de escala

Cálculo de:	Valor	Justificación
Esfuerzo	26,4	$\Pi EM_i \approx 0.96$ $E \approx 1,05$ $\sum SF_i = 14,58$ PREC = 2,48 FLEX = 2,03 RESL = 2,83 TEAM = 0,0 PMAT = 6,24
Tiempo de desarrollo	<b>10 meses</b>	$PM \approx 26.4$ $F \approx 0.30$
Cantidad de personas	<b>3</b>	Se cuenta con 3 personas para la realización del sistema.
Costo	<b>17820 pesos</b>	$C = CHM * PM$ $CHM = 3 * 225 = 675$ $PM \approx 26.4$
Salario medio	225	El salario mínimo es de 225 pesos.
RCPX	1,33	RELY = NOMINAL DOCU = ALTO CPLX = ALTO DATA = NOMINAL

RUSE	1,00	Existe concordancia entre la documentación y las necesidades del ciclo de vida.
PDIF	1,00	El .NET es una plataforma estable, no hay restricciones fuertes de memoria o tiempo nominal
PREX	1.00	Los desarrolladores tienen buena experiencia en la plataforma y herramientas de desarrollo. Lleva 1 año.
FCIL	0,87	Se hace un uso notable de herramientas CASE para documentar implementar el sistema.
SCED	1,00	Las exigencias para el cumplimiento de cronograma son normales.
PERS	0,83	ACAP = ALTO PCAP = ALTO PCON = ALTO

**Tabla 8. Cálculos Finales de La Estimación de Costos y Esfuerzos.**

#### 5.4 Beneficios tangibles e intangibles

El Sistema de Control de Acceso a la UCI no es un software con fines comerciales, aunque puede ampliarse para convertirlo en una solución general, capaz de aplicarse a cualquier empresa o institución.

Su principal objetivo es ayudar al control de la seguridad interna de la Universidad de Ciencias Informáticas.

Por tanto, los beneficios inmediatos son mayormente intangibles:

1. Ahorro de tiempo en la búsqueda de información de cualquier persona o transporte en la UCI
2. Control de quién entra y sale del centro, con fecha y hora.

3. Posibilidad de ver reportes de accesos o visitas, con facilidad de búsqueda.
4. Hacer más eficiente el proceso de solicitar visitas de personas externas, con mayor control sobre los datos de la visita.
5. Disponibilidad de los datos para estadísticas y toma de decisiones.

### 5.5 Análisis de costos y beneficios

El desarrollo de este sistema no supone grandes gastos de recursos, ni tampoco de tiempo; la base de datos que contiene la información, puede ser alojada en los servidores existentes en la Universidad, ya que los mismos tienen buenas prestaciones y acceso rápido. La tecnología utilizada para el desarrollo del sistema es .NET, que es gratis. El sistema se ha diseñado pensando en una eventual migración al proyecto Mono por lo que un cambio de plataforma para la implantación del mismo es viable y factible, y no hay que incurrir en muchos cambios; debido a la estructuración en capas de los procesos del negocio que se diseñaron.

El gasto radica mayormente en los lectores de código de barras necesarios para controlar el acceso a trabajadores y transportes, se compraron inicialmente 4 lectores, cuyo precio total asciende a \$2,848.08, gasto que se justifica por las ventajas de seguridad que brindará al sistema, al prevenir la entrada de personas no autorizadas al centro, ayudando a proteger los valiosos recursos invertidos en la Universidad.

Se investigaron otras soluciones de control de acceso disponibles en el mundo, y se llegó a la conclusión de que no se ajustan a las características del entorno de la Universidad, además, son demasiado caras.

El sistema puede ser extendido para uso general, obteniéndose un producto comercializable que puede ser fuente de ingresos.

## 5.6 Conclusiones

En este capítulo se efectuó el estudio de factibilidad correspondiente al desarrollo del proyecto. Este permitió llegar a la conclusión que resultará factible implementar la aplicación, ya que aunque el existe cierto costo total, los beneficios sociales que se alcanzarán son considerables.



## CONCLUSIONES

Con este trabajo se propone una solución integral al problema del control de acceso en la Universidad de Ciencias Informáticas, ya que se hacen comparaciones de las tecnologías existentes para ser utilizadas en los distintos casos donde se quiera controlar el acceso o verificar identidad. Se presenta una aplicación capaz de llevar un registro de los accesos de trabajadores, transportes y visitantes, y que brinda reportes sobre los mismos. Esta aplicación, debidamente utilizada, se puede convertir en una poderosa herramienta para el control de la seguridad en la Universidad y la toma de decisiones. Además, brinda un banco de datos sobre entradas y salidas, que puede ser reutilizado por otras futuras aplicaciones de la intranet para controlar otros procesos.

El sistema se desarrolló siguiendo la metodología RUP, y se utilizaron representaciones UML para la modelación de todas las fases del proyecto.

El sistema resultante está provisto de un ambiente cómodo, fácil de entender, que cumple los estándares de diseño y utiliza técnicas modernas de programación orientada a objetos.

Por todo lo anterior se concluye que los objetivos propuestos para el presente proyecto han sido cumplidos satisfactoriamente. Se incluyen una serie de recomendaciones que deben tenerse en cuenta para el trabajo futuro.

## RECOMENDACIONES

Los objetivos generales de este trabajo han sido logrados, pero a lo largo de su desarrollo, han ido surgiendo ideas que podrían implementarse en un futuro, de forma que se logre una la aplicación más útil y efectiva, para lo cual se recomienda:

1. Integrar más las aplicaciones de la intranet con el sistema, para beneficiarse de la información recopilada por este.
2. Extender el sistema de manera que pueda ser utilizado no sólo en la Universidad, sino en cualquier empresa que requiera de un control de accesos y visitas, convirtiéndose en una fuente de ingresos.
3. Utilizar el módulo de envío de correo electrónico en otros sistemas de la intranet, ya que es un módulo independiente y configurable, pudiendo tener más aplicaciones.
4. Continuar el desarrollo de este sistema, adicionándole nuevas funcionalidades, adecuándolo más a las demandas de la creciente y dinámica intranet de la Universidad y haciéndolo más útil y provechoso.

## GLOSARIO DE TÉRMINOS

- **Camel Case:** Estilo de escritura de código, se basa en capitalizar las iniciales de las todas las palabras que forman el nombre de un método o variables, menos la primera.
- **CRUD:** Siglas en Inglés de las operaciones básicas de creación, lectura, modificación y borrado que se hacen sobre clases persistentes. (Create, Read, Update, Delete).
- **CSS:** Siglas en Inglés de Cascading Style Sheets (Hojas de Estilo en Cascada).
- **Passback:** Evento que ocurre cuando se detectan dos entradas o salidas consecutivas de una misma credencial.
- **SOAP:** Es un protocolo elaborado para facilitar la llamada remota a funciones a través de Internet, permitiendo que dos programas se comuniquen de una manera muy similar técnicamente a la invocación de páginas Web.
- **UML:** Lenguaje Unificado de Modelado (UML, por sus siglas en inglés, Unified Modelling Language) es el lenguaje de modelado de sistemas de software más conocido en la actualidad.
- **URL:** Acrónimo de Universal Resource Locator (localizador universal de recursos), método de identificación de documentos o lugares en Internet, que se utiliza principalmente en World Wide Web (WWW). Un URL es una cadena de caracteres que identifica el tipo de documento, la computadora, el directorio y los subdirectorios en donde se encuentra el documento y su nombre.
- **Web Services:** Aplicación que realiza un cometido y que puede formar parte de otros servicios para formar un servicio más completo. La comunicación hacia y desde el Webservice se realiza con XML. Permite una llamada a una funcionalidad localizada en un servidor remoto.
- **XML:** Es el acrónimo de eXtensible Markup Language (lenguaje de marcado extensible) desarrollado por el World Wide Web Consortium (W3C).

## REFERENCIAS BIBLIOGRÁFICAS

- [1] Orama Fernández, Miguel de Jesús: Sistema para el Control de Visita. Trabajo de Diploma para optar por el título de Ingeniero Informático, Instituto Superior Politécnico “José Antonio Echeverría”, Universidad de las Ciencias Informáticas, Ciudad de la Habana, junio 2004.
- [2] Fernández Hernández, Yumilka Bárbara: Sistema para el Control de Visitas. Trabajo de Diploma para optar por el título de Ingeniero Informático, Instituto Superior Politécnico “José Antonio Echeverría”, Universidad de las Ciencias Informáticas, Ciudad de la Habana, junio 2004.
- [3] Booch, G., Rumbaugh, J., Jacobson, I. *“El Lenguaje Unificado de Modelado”*. Addison-Wesley. 1999.
- [4] Características de Visual Studio .NET. Microsoft Corp. 2005.  
<http://www.microsoft.com/latam/vstudio/producto/caracteristicas.asp>
- [5] Introducción a .NET. Monografías. 2004.  
<http://www.monografias.com/trabajos11/winnet/winnet.shtml>
- [6] Preguntas sobre códigos de barras. Barmax Corp. 2005  
<http://www.barmax.com/faq.asp>
- [7] .NET. Wikipedia. 2005.  
[http://es.wikipedia.org/wiki/.NET\\_de\\_Microsoft](http://es.wikipedia.org/wiki/.NET_de_Microsoft)
- [8] Express Edition Beta Products. Microsoft Corp. 2005.  
<http://lab.msdn.microsoft.com/express/>

## BIBLIOGRAFÍA

1. Rumbaugh, J.; Jacobson, I. y Booch, G.; “El Lenguaje Unificado de Modelado. Manual de Referencia. 2000. Addison-Wesley.
2. Von Hallen, B.; “Building a Business Rules”.  
<http://www.kpiusa.com/ReadingRoom/ReadingRoom.htm>
3. Agualló, P.; “Desarrollo Cliente / servidor: ubicación de las reglas de negocio”.  
<http://www.ctv.es/USERS/pagullo/arti/esbr/esbr.htm>
4. S. Giandin, Roxana. “Relaciones entre Casos de Uso en el Unified Modeling Language”. Revista colombiana de computación. Vol. 1, núm. 1.
5. ¿Qué es ASP.NET?  
<http://www.zonagratis.com/microsoft/asp/aspnet.htm> (01-06-2005).
6. Cómo Investigar. (2-6-2005)  
<http://www.monografias.com/trabajos13/artinves/artinves.shtml>
7. Control de Acceso basado en la PC (3-6-2005)  
<http://www.jantek.com/Spanish/jaspcbased.htm>
8. Araujo, J. Whittle, J., Toval Álvarez, J. A. France, R. Integration and Transformation of UML Models. ECOOP Workshops 2005.
9. Object Management Group. Unified Modeling Language Specification. v1.5. Marzo 2005.
10. Arlow, J. Neustadt, I. UML and the Unified Process: Practical Object-Oriented Analysis and Design. ISBN 0201770601. Addison-Wesley.
11. Miller, J. Mukerji, J. MDA Guide Version 1.0.1 Technical Report, OMG. 2003.
12. Historia de los sistemas de bases de datos  
<http://www3.uji.es/~mmarques/f47/apun/node6.html>. (14/02/2004)
13. Funciones de los sistemas de gestión de bases de datos  
<http://www3.uji.es/~mmarques/f47/apun/node39.html>. (14/02/2004)
14. Arquitectura Cliente/Servidor  
<http://www.inei.gob.pe/cpi-mapa/bancopub/libfree/lib616/cap0301.HTM>  
(11/5/2005)

15. Craig Larman “UML y Patrones. Introducción al Análisis y Diseño Orientado a Objetos.” Prentice Hall. 1999.
  16. Practical UML. A Hands-On Introduction for Developers.  
[http://info.borland.com/techpubs/together/together\\_guides/umlonlinecourse/#component-and-deployment-diagrams](http://info.borland.com/techpubs/together/together_guides/umlonlinecourse/#component-and-deployment-diagrams) (14/6/2005)
  17. Agile Models Distilled: Potential Artifacts for Agile Modeling.  
<http://www.agilemodeling.com/artifacts/> (14/6/2005)
  18. Requirement Analysis  
[http://www.popkin.com/solutions/requirements\\_analysis.htm](http://www.popkin.com/solutions/requirements_analysis.htm) (14/6/2005)
- UML basics: An introduction to the Unified Modeling Language.  
<http://www-128.ibm.com/developerworks/rational/library/769.html#footnotes> (14/5/2005)