

Universidad de las Ciencias Informáticas

FACULTAD 1



Título: “Análisis y Diseño del sistema de Control de Acceso de la Universidad de las Ciencias Informáticas”.

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

Autores: Maidel Ojeda Cruz.

Maité Suárez Oliveros.

Tutor: Ing. Julio Cesar Herrera Soria.

Junio 2008

DECLARACIÓN DE AUTORÍA

DECLARACIÓN DE AUTORÍA

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

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

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

Firma de la Autora

Firma de la Autora

Firma del Tutor

OPINIÓN DEL TUTOR DEL TRABAJO DE DIPLOMA

OPINIÓN DEL TUTOR DEL TRABAJO DE DIPLOMA

Título: “Análisis y Diseño del sistema de Control de Acceso de la Universidad de las Ciencias Informáticas”.

Autoras: Maidel Ojeda Cruz.

Maité Suárez Oliveros.

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

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 _____.

Julio Cesar Herrera Soria

Firma

Fecha

Datos de Contacto

Tutor: Julio Cesar Herrera Soria.

Graduado de ingeniería en informática en el curso 2006 - 2007 en la Universidad de Holguín, se encuentra en estos momentos en proceso de adiestramiento.

Agradecimientos

A la Universidad de las Ciencias Informáticas, por habernos formado como profesionales.

A nuestro tutor por su apoyo.

Maité

A mis compañeros de cinco años de estudio y esfuerzo, por brindarme su amistad y apoyo, en especial a Dole.

A mi abuela Yoya que tanta fuerza me dió.

A mis tíos Mercy, Tita y Juanjo por estar siempre presentes en los momentos que los he necesitado.

A Robert que de no ser por él nada hubiese sido posible.

A todas aquellas personas que permitieron que este, mi sueño se hiciera realidad.

Maidel

A mi mamá y mi papá por brindarme su amor, su cariño, dedicación y sobre todo la confianza que tuvieron en mí.

A mi hermano por darme su apoyo.

A toda mi familia.

A mis compañeros de aula con los que tantos momentos inolvidables compartí.

A todos los que de una forma u otra han contribuido en mi formación profesional y personal.

A todos, muchas gracias.

Dedicatoria

Maité

A mi mamá y mi papá por su dedicación y por haber sabido guiarme en tantos años de estudio.

A mis abuelos Mimi y Abu por darme el apoyo y la seguridad que he necesitado y por su amor eterno.

A mi Bebé por estar a mi lado en los momentos más importantes y por tanta paciencia y cariño.

A mis hermanas y mi primo para servir como ejemplo y guía.

Este es mi regalo.

Maidel

A mi familia, especialmente a mis padres.

Resumen

El Control de Acceso en la Universidad de las Ciencias Informáticas (UCI) consiste en controlar las entradas y salidas del personal, vehículos y visitantes. En la Universidad existe un sistema de control de acceso de personas que no se está utilizando, ya que no cumple con las políticas establecidas por la misma, y no permite controlar los accesos del personal, vehículos y visitantes de forma integrada. Actualmente, este flujo de información se realiza de forma manual, lo que provoca la entrada de personas ajenas al centro y demoras significativas en el acceso al mismo.

Por lo expresado anteriormente el presente trabajo, tiene como objetivo realizar el análisis y diseño de un sistema informático que gestione el control de acceso en la UCI. Para este fin se siguieron los pasos que propone la metodología de desarrollo Rational Unified Process (RUP), con vista a llevar a cabo los flujos de trabajo, Modelación del Negocio, Requerimientos, Análisis y Diseño. Se utilizó el Lenguaje Unificado de Modelado (UML), y Visual Paradigm para el modelado visual.

Con el desarrollo de este trabajo se pretende sentar las bases para la implementación de un sistema, que permita elevar la eficiencia en los procesos de control de acceso, reducir el esfuerzo y aumentar la seguridad interna de la UCI.

Palabras Claves: Control de Acceso, Personal, Vehículos, Visitantes, Procesos, Sistema.

Tabla de Contenidos

Introducción..... 1

Capítulo 1: Fundamentación Teórica 5

1.1 Introducción.....5

1.2 Sistemas de Control de Acceso.....5

1.3 Tendencias Actuales de los sistemas de Control de Acceso8

1.4 Tecnología utilizada en la UCI: código de barras.....11

1.5 Lectores de código de barra12

1.6 Metodología de Desarrollo de Software13

 1.6.1 Caracterizar RUP14

 1.6.2 La vida de un producto software según RUP.....15

1.7 Lenguaje Unificado de Modelado (UML).....15

1.8 Herramienta CASE15

1.9 Lenguaje de Programación.....17

1.10 Base de Datos: DB4o17

1.11 Aplicaciones de Escritorio.....18

1.12 Conclusiones19

Capítulo 2: Características del Sistema 20

2.1 Introducción.....20

2.2 Flujo Actual de Procesos involucrados en el campo de acción.....20

2.3 Objeto de automatización21

2.4 Información que se maneja21

2.5 Reglas del negocio a considerar.....21

2.6 Modelo de negocio22

 2.6.1 Descripción de los Actores y Trabajadores del Negocio.....22

TABLA DE CONTENIDOS

2.6.2 Descripción de los Casos de Uso del Negocio	23
2.6.3 Diagrama de actividades para casos de uso del negocio	26
2.6.4 Diagrama de Casos de Uso del Negocio.....	26
2.6.5 Diagrama de objetos del Negocio	27
2.7 Requerimientos	28
2.7.1 Requerimientos funcionales	28
2.7.2 Requerimientos no funcionales	29
2.8 Definición de los Casos de Uso del Sistema.....	31
2.8.1 Actores del sistema.....	31
2.8.2 Diagrama de Casos de Uso del Sistema.....	32
2.8.3 Descripción de los casos de uso del sistema	32
2.9 Propuesta de sistema.....	50
2.10 Análisis de la Factibilidad	51
2.11 Conclusiones.....	57
Capítulo 3: Análisis y Diseño del sistema	58
3.1 Introducción.....	58
3.2 Modelo de Análisis	58
3.2.1 Diagramas de Clases del Análisis	58
3.3 Modelo de Diseño.....	60
3.3.1 Diagrama de clases de diseño	60
3.3.1.1 Diagramas de Secuencia.....	65
3.3.2 Patrones de Diseño.....	66
3.4 Descripción de la arquitectura	67
3.5 Descripción de Clases	67
3.5.1 Clases Interfaz	67

TABLA DE CONTENIDOS

3.5.2 Clases Controladoras.....	75
3.5.3 Clases entidad	85
3.5.4 Clases de Acceso a datos.....	90
3.6 Conclusiones.....	96
Conclusiones	97
Recomendaciones	98
Bibliografía Referenciada.....	99
Bibliografía Consultada.....	99
Glosario	102

Índice de Tablas

Tabla 1 Actores del negocio.	22
Tabla 2 Trabajadores del negocio.....	23
Tabla 3 Descripción Caso de Uso del Negocio: Caso de Uso: “Entrar Persona”.....	24
Tabla 4 Descripción Caso de Uso del Negocio: “Salir Persona”.....	24
Tabla5 Descripción Caso de Uso del Negocio: “Entrar Vehículo”.....	24
Tabla6 Descripción Caso de Uso del Negocio: “Salir Vehículo”.....	25
Tabla7 Descripción Caso de Uso del Negocio: “Entrar Visitante”.....	25
Tabla8 Descripción Caso de Uso del Negocio: “Salir Visitante”.....	25
Tabla9 Descripción Caso de Uso del Negocio: “Informar Medios”.....	26
Tabla10 Actores del sistema.....	31
Tabla11 Descripción del caso de uso “Controlar acceso puntual de personas de la UCI”.....	36
Tabla12 Descripción del caso de uso “Controlar accesos de vehículos”.....	39
Tabla13 Descripción del caso de uso “Controlar accesos de visitantes”.....	43
Tabla14 Descripción del caso de uso “Actualizar Lector”.....	45
Tabla15 Descripción del caso de uso “Actualizar BD Local”.....	46
Tabla 16 Descripción del caso de uso “Configurar Sistema”.....	50
Tabla17 Factor de peso de los actores sin ajustar.....	51
Tabla18 Factor de peso de los casos de uso sin ajustar.....	52
Tabla19 Factor de complejidad técnica.....	53
Tabla20 Factor de ambiente.....	54
Tabla21 Esfuerzo del proyecto.....	56
Tabla22 Clase Interfaz “CI_Controlar Persona”.....	68
Tabla23 Clase Interfaz “CI_Controlar Vehículo”.....	69
Tabla24 Clase Interfaz “CI_Controlar Visitantes”.....	71

ÍNDICE DE TABLAS

Tabla25 Clase Interfaz “CI_Configurar Sistema”	73
Tabla26 Clase Interfaz “CI_Actualizar Lector”	74
Tabla27 Clase Interfaz “CI_Actualizar_Base_Datos_Local”	74
Tabla28 Clase Controladora “CC_ControlarPersona_Imple”	75
Tabla29 Clase Controladora “CC_ControlAccesoPersona_Imple”	76
Tabla30 Clase Controladora “CC_ControlPersona_Imple”	77
Tabla31 Clase Controladora “CC_ControlarVehiculo_Imple”	77
Tabla32 Clase Controladora “CC_ControlAccesoVehiculo_Imple”	78
Tabla33 Clase Controladora “CC_ControlVehiculo_Imple”	79
Tabla34 Clase Controladora “CC_ControlarVisitantes_Imple”	79
Tabla35 Clase Controladora “CC_AccesoVisitantesControl_Imple”	80
Tabla36 Clase Controladora “CC_ConfigurarSistema”	81
Tabla37 Clase Controladora “CC_ControlSincronizacion_Imple”	82
Tabla38 Clase Controladora “CC_ActualizarLector_Imple”	83
Tabla 39 Clase Controladora “CC_ControlOptimusS_Imple”	83
Tabla40 Clase Controladora “CC_Actualizar_Base_Datos_Local_Imple”	84
Tabla41 Clase Controladora “CC_PassBack_Imple”	84
Tabla42 Clase Entidad “Persona”	85
Tabla43 Clase Entidad “Vehículo”	85
Tabla44 Clase Entidad “Visitante”	86
Tabla45 Clase Entidad “ConfigurarSistema”	88
Tabla46 Clase Entidad “PassBack”	88
Tabla47 Clase Entidad “AccesoPersona”	88
Tabla48 Clase Entidad “Acceso Vehiculo”	89
Tabla49 Clase Entidad “AccesoVisitante”	89

ÍNDICE DE TABLAS

Tabla50 Clase Entidad "CausaAcceso"	89
Tabla 51 Clase de Acceso a Datos "ControlarAccesoPersona_DAO_Imple"	90
Tabla 52 Clase de Acceso a Datos: "ControlarPersona_DAO_Imple"	91
Tabla 53 Clase de acceso a datos: "ControlAccesoVehiculo_DAO_Imple"	92
Tabla 54 Clase de acceso a datos: "Control_Vehiculo_DAO_Imple"	92
Tabla 55 Clase de acceso a datos: "Controlar Visitante_DAO_Imple"	93
Tabla 56 Clase de acceso a datos: "AccesoVisitanteControl_DAO_Imple"	94
Tabla 57 Clase de acceso a datos: "Control_Configuracion_DAO_Imple"	95
Tabla 58 Clase de acceso a datos: "CC_Control_Sincronizacion_DAO_Imple"	96
Tabla 59 Clase de acceso a datos: "Passback_DAO_Imple"	96

Índice de Figuras

Figura 1 Optimus Colector Metrologic SP-5500 Optimus S Bluetoothya.	13
Figura 2 Diagrama de Casos de Uso del Negocio.	27
Figura 3 Diagrama de objetos del Negocio.	27
Figura 4 Diagrama de Casos de Uso del Sistema.	32
Figura5 DCA “Controlar acceso puntual de personas de la UCI”	58
Figura6 DCA “Controlar acceso de vehículos”	59
Figura7 DCA “Controlar acceso de visitantes”	59
Figura8 DCA “Configurar Sistema”	59
Figura9 DCA “Actualizar lector”	59
Figura10 DCA “Actualizar BD local”	60
Figura11 DCD: “Controlar Acceso puntual de personas de la UCI”	61
Figura 12 DCD: “Controlar Acceso de Vehículos”	62
Figura 13 DCD: “Controlar Acceso de Visitantes”	63
Figura 14 DCD: “Configurar Sistema”	63
Figura 15 DCD: “Actualizar Base de Datos Local”	64
Figura 16 DCD: “Actualizar Lector”	65

Introducción

La Universidad de las Ciencias Informáticas es un centro con características atípicas al resto de las universidades del país, esta cuenta con más de 10000 estudiantes internos, asimismo existen más de 3000 profesores, tanto internos como externos, los cuales pueden salir y entrar de la Universidad en cualquier momento. Además en la misma trabajan una gran cantidad de personas externas, que realizan otras labores de servicios, y que acceden diariamente.

La Universidad cuenta con una gran cantidad de transportes que se destinan para las actividades complementarias a las actividades docentes, también acceden al centro otros transportes externos, que por algún motivo tienen necesidad de ello, ya sea por razones laborales, o de visita protocolar.

Debido a las características especiales de la Universidad, en la cual existe una gran cantidad de recursos tecnológicos e información confidencial, que debe estar protegida ante el personal no autorizado, se necesita tener un control de todas las personas que acceden a la misma.

Para el control de acceso a la Universidad no solo se tiene en cuenta el personal de esta, dígase estudiantes, profesores y trabajadores, sino también los visitantes que están autorizados para el acceso, así como los vehículos.

Los sistemas de control de acceso permiten a las diferentes entidades mantener un control estricto de personas y vehículos que acceden a las mismas. Registrando los horarios y permitiendo obtener los reportes necesarios para mantener controlados los accesos en un determinado período.

En la Universidad existe un sistema de control de acceso de personas que no se está utilizando, ya que no cumple con las políticas establecidas por la misma, y no permite controlar los accesos del personal, vehículos y visitantes de forma integrada.

Actualmente por el gran flujo de personal existente y la realización del trabajo de forma manual, trae consigo que esta labor sea muy engorrosa, dando lugar a la entrada de personas ajenas al centro, gasto de materiales, y demora en las entradas y salidas.

Debido a lo expuesto anteriormente el **problema científico** queda formulado de la siguiente manera: En la UCI no existe un sistema que controle de forma integrada los accesos del personal, vehículos y visitantes, y que se ajuste a las políticas establecidas.

El **objeto de estudio** lo constituye el control de acceso del personal, vehículos y visitantes a la UCI.

El **campo de acción** lo constituyen los procesos de control de acceso del personal, vehículos y visitantes en las puertas de acceso a la UCI.

La **idea a defender** que se plantea es: La realización del análisis y diseño de un sistema informático, que gestione las entradas y salidas del personal, vehículos y visitantes a la UCI, permitirá la futura implementación del mismo.

El **objetivo general** es: Realizar el análisis y diseño de un sistema informático, que gestione el control de acceso a la UCI. Para darle cumplimiento al objetivo anteriormente planteado se desarrollaron las siguientes tareas:

- Estudiar las funcionalidades de sistemas de Control de Acceso.
- Estudiar la metodología, lenguajes y herramienta a utilizar.
- Realizar la modelación del Negocio y del Sistema.
- Desarrollar el análisis y diseño de los procesos.

Diseño Metodológico:

❖ Métodos de Investigación Científica:

- Los **métodos teóricos** utilizados para cumplir con las tareas son:
 - **Histórico Lógico:** Se usó para estudiar de forma analítica la trayectoria histórica de los fenómenos, evolución y desarrollo de los procesos de control de acceso.
 - **Modelación:** Se modelaron los diagramas para facilitar la implementación del sistema, como parte de la propuesta de solución.

- **Analítico-Sintético:** Utilizado para analizar la bibliografía y realizar una síntesis de la misma.
- Los **métodos empíricos** utilizados para obtener información sobre el objeto de estudio son:
 - **Entrevistas:** Mediante las entrevistas se logró conocer y comprender los problemas existentes, y en cuales se basó la investigación, asimismo sirvió para determinar las características de la propuesta de solución, basadas en las necesidades del cliente.

Se espera desarrollar una propuesta que dé solución a las problemáticas existentes en la UCI, mediante el análisis y diseño de un sistema que permita elevar la eficiencia en los procesos de control de acceso, mediante el control de forma integrada del personal, vehículos y visitantes, para reducir el esfuerzo que implica la realización de estas actividades, por parte del personal encargado de ello, y sobre todo aumentar la seguridad interna en la UCI.

La tesis está estructurada en tres capítulos, a continuación se describe el contenido de estos:

- **Capítulo 1. Fundamentación Teórica:**

En este capítulo se tratarán los principales conceptos relacionados con el proceso de control de acceso. Se hace un estudio acerca de sistemas de control de acceso, tanto a nivel nacional como internacional, se describen tendencias actuales de estos, y se reflejan cuáles son los lenguajes, herramienta y tecnologías utilizadas para el desarrollo de la propuesta de solución.

- **Capítulo 2. Características del Sistema:**

En este capítulo se modela el negocio estudiado, y se determinan las características del sistema a desarrollar. Se modelan los artefactos relacionados con los flujos de trabajo de Modelado del Negocio y Requerimientos, y se realiza un estudio de la factibilidad.

- **Capítulo 3. Análisis y Diseño:**

En este capítulo se muestra el análisis y diseño del sistema, se representan los diagramas de clases de análisis y diseño pertenecientes a cada caso de uso. Además se abordan conceptos básicos para una mejor explicación del tema.

Capítulo 1: Fundamentación Teórica

1.1 Introducción

En este capítulo se tratarán los principales conceptos relacionados con el proceso de control de acceso. Se hace un estudio acerca de sistemas de control de acceso, tanto a nivel nacional como internacional, se describen tendencias actuales de estos, y se reflejan cuáles son los lenguajes, herramienta y tecnologías utilizadas para el desarrollo de la propuesta de solución.

1.2 Sistemas de Control de Acceso

Existen innumerables sistemas de control de acceso que le dan seguridad y protección a diversas instalaciones, desafortunadamente se vive en tiempos donde los riesgos incrementan, por lo que surge la necesidad de utilizar estos sistemas. Son numerosos los sectores y las aplicaciones en los que se tiene en cuenta el control de acceso, tanto en entornos domésticos como casas, edificios, control de acceso comercial e industrial, control de acceso para hoteles, balnearios, centros sanitarios, universidades, centros deportivos, puertos y otras áreas reservadas o de seguridad. A continuación se nombrarán algunos de estos sistemas, los que han sido desarrollados e implantados en otros países, así como en la propia Universidad.

Sistemas Internacionales

➤ **Sistema de Control de Acceso del Puerto Vilagarcía, Pontevedra, España.**

El Sistema de Control de Acceso del puerto Vilagarcía, consta de 8 accesos al recinto portuario con: barreras para vehículos, molinetes para peatones, lectores de tarjetas para la identificación, interfonos para la comunicación, y cámaras para la vigilancia; también un centro de control y vigilancia en las instalaciones de la Autoridad Portuaria en el muelle, y un centro de vigilancia en la caseta de la entrada al Muelle Comercial.

Los usuarios habituales del puerto poseen tarjetas digitales, que les permiten la entrada y salida automatizadas en las diferentes áreas del puerto, a través de las vías que les correspondan. Estas tarjetas digitales son entregadas por la Autoridad Portuaria a todos los

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

usuarios que lo soliciten, y que cumplan con las condiciones necesarias para cada tipo de tarjeta.

➤ **Sistema de Control de Acceso y presencia laboral con tarjeta chip en Universidades.**

Aplicación realizada en las Universidades de Zaragoza, Huesca y Teruel, consistente en 140 terminales autónomas con contacto chip: terminal off-line Kimaldi Kreta2 y lector chip Kimaldi SC6000 (lectura de chip TIBC y WG10).

El terminal Kreta2 trabaja de forma completamente autónoma, lo que permite almacenar los diferentes marcajes y eventos en su memoria, para posteriormente enviarlos a la computadora. Esto permite una capacidad de almacenaje de 1.000 tarjetas autorizadas, 4.500 marcajes y 3.000 excepciones. En esta aplicación se ha ampliado la capacidad de memoria del Kreta2 a 5.000 usuarios.

Dispone de diferentes tipos de conexión (RS-232, TCP/IP o CAN) y su configuración se puede realizar desde la PC o host. Esta aplicación, es utilizada para el control de acceso de los estudiantes y presencia laboral del personal de la Universidad. Permite su acceso a diferentes áreas o zonas restringidas (laboratorios, despachos, sala de conferencias, bibliotecas...).

Todos los lectores están conectados mediante conexión TCP/IP, y centralizada en el Centro de Cálculo (Zaragoza) a través de la intranet de la Universidad.

➤ **Venta de entradas y control de acceso a la Copa América Venezuela 2007.**

La Copa América es un evento de grandes dimensiones que se celebra entre todos los equipos de fútbol de Sudamérica, además de México y Estados Unidos. Para el campeonato de la Copa América 2007, celebrado en Venezuela, fue integrado un sistema de venta entradas y control de accesos para grandes recintos, con tarjetas electrónicas NdCAN Max en los tornos de control de acceso. Las tarjetas electrónicas NdCAN Max son muy útiles en controles de acceso muy concurridos, ya que son totalmente integrables y se adaptan fácilmente a las necesidades generadas por cada ocasión.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

En este sistema existe la integración de los torniquetes de acceso al recinto, de una tarjeta electrónica NdCAN Max, un lector de código de barras y un display de información. Todo ello funcionando on-line con el servidor de control de acceso, gracias a un software de comunicación, que consigue que el sistema de control de accesos esté perfectamente integrado con el sistema de venta de entradas.

El funcionamiento es de la siguiente manera: el usuario llega al turno con su ticket de entrada y pasa el código de barras por el lector, el lector envía la información al host y este la compara en la base de datos, una vez comprobado, envía la información de retorno, si es positiva accionará el relé que abrirá el turno de control de acceso, en caso contrario enviará un mensaje al display informando de la incidencia que se produjo.

Sistemas existentes en la UCI

En la Universidad existen sistemas vinculados al Control de Acceso, que están encaminados a resolver diversos problemas relacionados con esta esfera.

Sistema de Control de Acceso a la UCI:

El Sistema de Control de Acceso a la UCI fue creado en la Universidad en el año 2006, este controla los accesos de las personas donde intervienen procesos como: controlar el acceso (entrada y salida) de personas que residen o trabajan en la UCI, verificando su existencia y si esta tiene autorización a entrar o salir, almacenando cada acceso con datos de la persona, la hora y el tipo de acceso. Utiliza la plataforma .Net y SQLServer como gestor de base de datos.

Sistema de Comedores:

El Sistema de Control de Acceso a los comedores en la UCI, brinda información sobre por cuál complejo y puerta debe pasar una persona, además de permitir visualizar la cantidad de comensales que han pasado hasta el momento. Siendo de carácter obligatorio que cada persona porte su solapín, para de esta forma controlar que no ocurran accesos inadecuados. El sistema puede trabajar sin red, y cuando esta llega, sincroniza los accesos que ha tenido hasta el momento.

➤ Sistema de Acreditación:

Mediante este sistema se controla todo el personal existente en la Universidad, guardando sus datos personales, para esto se tiene en cuenta que la persona puede ser un estudiante o un trabajador. Este sistema consiste en la realización de un grupo de actividades, que se deben tener en cuenta a la hora de acreditar al personal que está presente en la Universidad, o que desea ingresar a la misma. Se le asigna una credencial con un código de barra único para cada individuo.

1.3 Tendencias Actuales de los sistemas de Control de Acceso

El propósito de describir algunas de las tendencias actuales de los sistemas de control de acceso es precisamente exponer el desarrollo que han experimentado estas tecnologías en los últimos tiempos, utilizándose en todo el mundo en el desarrollo de diversos software destinados para este fin. A continuación se relacionan algunas de ellas, así como sus características principales.

➤ Lector de huella dactilar de sobremesa con conexión a PC.

El lector de huella digital Nitgen Hamster es un periférico para la seguridad del ordenador y seguridad informática en general, aplicable a distintos campos donde la autenticación personal por reconocimiento dactilar sea necesaria. Está equipado con un módulo de lectura de huella dactilar basado en la tecnología única de biometría Nitgen.

Es un sistema de altas prestaciones, mucha precisión y larga durabilidad. Lector que se conecta directamente al ordenador, y de forma muy segura se reemplaza la contraseña que es vulnerable al fraude y difícil de recordar.

El Hamster ha sido diseñado por la tecnología más optimizada de reconocimiento de huella dactilar, y es muy robusto a posibles impactos físicos externos, aspectos ambientales y corriente estática. [2]

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

➤ **Identificación por radiofrecuencia, smart card o banda magnética.**

Funcionamiento off-line con múltiples opciones de conectividad y configuración mediante lectores con memoria para 7.500 usuarios.

Potente terminal basado en un procesador de 32 bits, top de la gama de productos de control de acceso y control de asistencia, tiene funcionamiento autónomo y memoria no volátil para almacenar la base de datos. En esta se guardan los permisos de los usuarios, histórico de marcajes, horarios y turnos con festivos y excepciones, mensajes personalizados para cada usuario y gestión de incidencias de acceso. Los marcajes de acceso (o fichajes), activación de alarma y otros eventos, son enviados a tiempo real y almacenados en la memoria con información del momento en que se han producido para ser recuperados cuando se desee.

Identificación por radiofrecuencia (RFID de 125 kHz y 13, 56 MHz), tarjeta chip, banda magnética o introducción de contraseña. El terminal admite dos lectores para entrada y dos para salida, lo que permite combinar varios tipos de identificación del usuario.

La conectividad con el host se puede realizar vía conexión IP, CAN, RS-232 completamente integrada y opcionalmente conexión Wi-Fi mediante conversor. El sistema permite crear una extensa red de terminales de control de acceso y control de asistencia. [2]

➤ **Solución para gestionar el control de acceso a universidades o cualquier centro de enseñanza.**

Aplicación múltiple de control de accesos y alarmas mediante radiofrecuencia. El sistema de seguridad y control visual lo proporcionan los mismos terminales de acceso que son capaces de activar puntualmente las cámaras.

El sistema incorpora un potente programa que, a banda del control específico de todos los tipos de registros para las entradas y/o salidas, gestiona el acceso por régimen jerárquico, por franjas horarias dentro de un calendario anual, el registro del histórico de los accesos no autorizados y los intentos de uso fraudulento.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Tanto la actualización de la base de datos como su mantenimiento, son ágiles y de fácil manejo, ya que no exigen conocimientos técnicos específicos.

El control de alarmas de las puertas de salida de emergencia del edificio principal, permiten visualizar por pantalla desde el centro de gestión cualquier incidencia que se produzca en cualquier salida de emergencia, así como el estado en que puede quedar (abierta o cerrada), y todos los registros de incidencias, históricos, activación por calendario y franjas horarias.

➤ **Servicio Autónomo de Identificación, Migración y Extranjería (SAIME).**

Este sistema es la encargada de la Identificación nacional, el control migratorio y de extranjeros de la República Bolivariana de Venezuela. Dentro de sus alcances está desarrollar las aplicaciones necesarias para la automatización de los procesos relativos a la Cedulación, Pasaporte, Migración y Extranjería.

Dispositivos usados en el sistema SAIME:

➤ **Captadores de Firma**

Este tipo de dispositivo permite la captura de firmas, simulando el proceso de firmado manual, a través de una tabla sensible al contacto, que captura los puntos que se generan al deslizar una especie de lápiz proveído con el equipo, sobre la superficie o área definida para su uso.

Se manejan dos variantes principales para estos equipos: los que presentan una pantalla LCD (Liquid Crystal Display) que muestra el trazo del usuario mientras se dibuja sobre ella, tal es el caso del Edad-Ink de la empresa estadounidense InterLink Electronics, y los que no presentan esta pantalla, solamente son sensibles al contacto. Estos equipos son utilizados en el proceso de captura de imágenes para el enrolamiento del ciudadano en los sistemas de pasaporte y cédula, así como en la definición de la firma autorizada para los documentos emitidos en los distintos procesos.

➤ **Lectores de página completa**

Los lectores de página completa, son dispositivos diseñados para la lectura de documentos de viaje, según se definen en el estándar de ICAO (International Civil Aviation Organization).

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Estos equipos cuentan con lectores ópticos para el reconocimiento de caracteres OCR, que se encuentran en determinadas áreas de lectura, como la MRZ (Machine Readable Zone), y la VIZ (Visible Inspection Zone), o zonas de lectura mecánica y zona de inspección visual de sus siglas en inglés.

Estos dispositivos también cuentan con láser y distintos tipos de luces para la captura de imágenes completas del documento, en colores del espectro visible, infrarrojo y ultravioleta, estos últimos para destacar características de seguridad que presentan cada diseño de documento, en dependencia del país de emisión, con el objetivo de realizar algún chequeo especial.

En el caso del sistema SAIME hasta el momento solo se ha hecho uso de dos de estos lectores de página completa, el Eagle CM2500 de la empresa Canadian Bank Note Company, para el control migratorio en los aeropuertos y el Visotec Expert 300 de la empresa alemana Bundes Druckerei, para la verificación de la calidad de los pasaportes que se emiten en el proceso de producción del CPID (Centro de Personalización de Documentos de Identificación).

1.4 Tecnología utilizada en la UCI: código de barras

Para el control de acceso se utiliza la identificación de la persona mediante su credencial y un código de barras impreso en este. El código de barras se lee mediante lectores, en este caso se utiliza el lector "Optimus SP5500". Una vez que este lee el código de barras, verifica que sea válida esa credencial, es decir que esa persona pertenece a la Universidad.

El código de barras es un código basado en la representación mediante un conjunto de líneas paralelas verticales, de distinto grosor y espaciado, que en su conjunto contienen una determinada información.

Esta tecnología 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.

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. [1]

1.5 Lectores de código de barras

Un lector de código de barras es en términos sencillos un dispositivo de entrada. Esto significa que su función es ingresar datos a la computadora. A continuación se describen algunos lectores y sus características, dentro de los cuales se encuentra el que posee la Universidad.

➤ **Lector p460:**

Es un Scanner versátil con memoria y arquitectura programable, dispositivo resistente y de alto rendimiento, diseñado para reducir al mínimo la fatiga en las aplicaciones de lectura de datos intensiva. Esta herramienta de recogida de datos por lotes (“batch”) le permite recoger datos mientras se desplaza libremente por los locales hasta un máximo de 12 horas. Para descargar datos, se inserta en la cuna conectada al host. También funciona en modo con cable cuando se conecta por este medio a un host. Además se pueden revisar y eliminar registros existentes de forma directa y rápida, gracias al teclado y a la pantalla montados en la parte superior.

➤ **Lector OptimusR:**

Es un potente y resistente terminal de recogida de datos. Ofrece desde 2 Mb hasta 8Mb de memoria Ram para la recolección de datos y 2 Mb de memoria Flash para programación, un lector láser, baterías Lion que ofrecen hasta 200 horas de trabajo, teclado de 24 teclas de goma, para programación o introducción de datos y una pantalla de cristal líquido de 128x64 puntos. Diseño ergonómico con especial atención a su protección antigolpes. Dispone de una amplia variedad de herramientas que posibilitan su programación en C++, Visual Basic o Windows Generator. Interfaz: USB, RS232, Infrarrojos (IRDA) o WireLess (802.11b). Códigos: Auto discriminante de todos los códigos de barras standard.

➤ Lector utilizado en la UCI (OptimusS):

El Optimus Colector Metrologic SP-5500 Optimus S Bluetoothya (Figura 1) es un lector portátil o fuera de línea con luz de propagación para el chequeo de tipo láser y este no es afectado por la iluminación a la hora de hacer la lectura.

Es un novedoso terminal de recogida de datos de reducidas dimensiones. Ofrece 2 Mb de memoria RAM para la recolección de datos y 1 Mb de memoria Flash para programación, un lector láser, baterías Lion que ofrecen hasta 100 horas de trabajo, memoria para datos de 2 MB SRAM, teclado de 21 teclas de goma, para programación o introducción de datos y una pantalla de cristal líquido de 100x64 puntos del tipo FSTN de led retroiluminado. Dispone de una amplia variedad de herramientas que posibilitan su programación en C++, Visual Basic o Windows Generator. [1]



Figura 1 Optimus Colector Metrologic SP-5500 Optimus S Bluetoothya.

1.6 Metodología de Desarrollo de Software

Las metodologías de desarrollo de software son un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos software. Estas metodologías, indican paso a paso todas las actividades a realizar para lograr el producto informático deseado, indicando qué personas deben participar en el desarrollo de las actividades y qué papel deben de tener. Además detallan la información que se debe producir como resultado de una actividad y la información necesaria para comenzarla. La metodología a aplicar es la que a continuación se describe.

➤ **Rational Unified Process (RUP).**

El RUP (Rational Unified Process) es un proceso de desarrollo de software creado por James Rumbaugh, Grady Booch e Ivar Jacobson para la Rational Corporation.

RUP es un proceso de desarrollo de software, o sea, conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema de software. Sin embargo, RUP 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. RUP está basado en componentes, lo cual quiere decir que el sistema software en construcción está formado por componentes de software interconectados a través de interfaces bien definidas y utiliza el Lenguaje Unificado de Modelado (Unified Modeling Language, UML).

1.6.1 Caracterizar RUP

El RUP se caracteriza por ser dirigido por casos de uso, centrado en la arquitectura e iterativo e incremental.

- ✓ **Dirigido por casos de uso:** RUP utiliza los casos de uso tanto para especificar los requisitos funcionales del sistema, como para guiar todos los demás pasos de su desarrollo, dígase diseño, implementación y prueba.
- ✓ **Centrado en la Arquitectura:** La arquitectura representa la forma del sistema, la cual va madurando en su interacción con los casos de uso hasta llegar a un equilibrio entre funcionalidad y características técnicas.
- ✓ **Iterativo e Incremental:** La alta complejidad de los sistemas actuales hace que sea factible dividir el proceso de desarrollo en varios mini-proyectos o versiones del producto donde a cada uno de estos se le denomina iteración y pueden o no representar un incremento en el grado de terminación del producto completo.

1.6.2 La vida de un producto software según RUP

La vida de un sistema transcurre a través de ciclos de desarrollo, en cada ciclo se repite el proceso unificado de desarrollo. Cada ciclo consta de cuatro fases: Inicio, elaboración, construcción y transición y finalizan con un hito donde se debe tomar una decisión importante:

- ✓ **Inicio:** Definir el alcance del proyecto.
- ✓ **Elaboración:** Construir una versión ejecutable de la arquitectura de la aplicación.
- ✓ **Construcción:** Completar el esqueleto de la aplicación con la funcionalidad.
- ✓ **Transición:** Obtener el release del proyecto.

Se utiliza RUP ya que las fases de desarrollo que propone, el cumplimiento de hitos en cada una de las fases y el trabajo por roles garantiza que se desarrolle el proyecto de manera efectiva y eficiente. Es la metodología que se estudia en la UCI, y se tiene bastante información sobre ella.

1.7 Lenguaje Unificado de Modelado (UML)

Como lenguaje para el modelado se utilizará UML (Lenguaje Unificado de Modelado), es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad, permite visualizar, especificar, construir y documentar un sistema de software.

UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocios y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables.

Es importante saber que UML no es una guía para realizar el análisis y diseño orientado a objetos, es decir, no es un proceso. UML es un lenguaje que permite la modelación de sistemas con tecnología orientada a objetos.

1.8 Herramienta CASE

Existen varias Herramientas CASE (Computer-Aided Systems Engineering) que dan asistencia a analistas, ingenieros de software y desarrolladores durante el ciclo de vida de

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

desarrollo de un software. La herramienta utilizada en el proyecto es la que se describe a continuación.

➤ **Visual Paradigm 6.0 Enterprise Edition.**

Se utilizará Visual Paradigm porque es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software. El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor coste.

Visual Paradigm ofrece:

- ✓ Entorno de creación de diagramas para UML 2.0.
- ✓ Diseño centrado en casos de uso y enfocado al negocio que generan un software de mayor calidad.
- ✓ Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.
- ✓ Capacidades de ingeniería directa e inversa.
- ✓ Modelo y código que permanece sincronizado en todo el ciclo de desarrollo.
- ✓ Disponibilidad de múltiples versiones, para cada necesidad.
- ✓ Disponibilidad de integrarse en los principales IDEs (Integrated Development Environment).
- ✓ Disponibilidad en múltiples plataformas.
- ✓ Ofrecen un mecanismo general para la organización de los modelos/subsistemas/capas agrupando elementos de modelado.
- ✓ Versión gratuita (licencia para Community Edition). [3]

Esta herramienta soporta anotaciones de JAVA y UML, se integra con Eclipse, Borland, JBuilder, Oracle y otras para soportar las fases de implementación en el desarrollo de software.

1.9 Lenguaje de Programación

➤ Java.

Como lenguaje de programación se utilizará Java, este fue desarrollado por la compañía Sun Microsystems. Es un lenguaje de programación orientado a objeto que muchas de sus sintaxis son tomadas de C y C++, pero tiene un modelo de objeto más simple y elimina herramientas de bajo nivel, que por lo general suelen inducir errores como la manipulación directa de punteros o memoria. Es un lenguaje muy extendido y cada vez con más importancia ya que es independiente de la plataforma, no importa el sistema operativo con el que se esté trabajando. Esto se debe al desarrollo de una máquina virtual para cada sistema que sirve de puente entre el sistema operativo y el programa de Java.

Sun Microsystems liberó la mayor parte de sus tecnologías Java bajo la licencia GNU GPL, de acuerdo con las especificaciones del Java Community Process, de tal forma que prácticamente todo el Java de Sun es ahora software libre.

En la actualidad Java se utiliza debido a sus grandes posibilidades y en gran medida a que la mayoría de las cosas que se hacen en cualquier otro lenguaje de programación se puede hacer en Java y muchas veces con grandes ventajas.

Una de las ventajas fundamentales de usar Java es que por ser software libre no es necesario comprar licencias de uso, que generalmente son muy caras, y permite la libre distribución del código sin violar los principios del software libre.

1.10 Base de Datos: DB4o

DB4O es un novedoso motor de base de datos orientada a objetos. Sus siglas se corresponden con la expresión "DataBase 4 (for) Objects", que a su vez es el nombre de la compañía que lo desarrolla: db4objects, Inc.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Las claves innovadoras de este producto son su alto rendimiento (sobre todo en modo embebido) y el modelo de desarrollo que proporciona a las aplicaciones para su capa de acceso a datos, el cual propugna un abandono completo del paradigma relacional de las bases de datos tradicionales.

Deja de existir un lenguaje SQL de consultas/modificaciones para pasar a crearse sistemas de consulta por métodos delegados y actualización/creación/borrado automático de entidades mediante código compilable.

Se elimina la necesidad de representar el modelo de datos de la aplicación en dos tipos de esquemas: modelo de objetos y modelo relacional. Ahora el esquema de datos del dominio viene representado por la implementación que se realice del diagrama de clases. Si se quiere desarrollar software libre con esta librería, su uso no conlleva ningún coste por licencia, Db4o es "open source".

Tiene algunas características interesantes:

- No hay diferencia de impedancia: los objetos se almacenan tal y como son.
- Manejo automático del esquema de base de datos.
- No hay que cambiar las clases para poder almacenarlas.
- Uniones de datos automatizadas.

1.11 Aplicaciones de Escritorio

Sin duda las aplicaciones de escritorio tienen una gran utilidad. Estas pueden estar representadas por sistemas de oficinas, aplicaciones gráficas, juegos. Algunas de las tecnologías principales usadas son: VBasic, C#, C++, y Java.

Entre sus ventajas se encuentran la rapidez de acceso a los datos, además normalmente no es necesaria una conexión a Internet ya que funciona localmente o en la intranet en la que está trabajando. Es más fácil el acceso a recursos locales como disco duro, impresora, memoria de vídeo, sonido, ratón, teclado, etc.; y también con un mayor acceso a aplicaciones

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

locales en caso de que sea necesario compartir datos entre aplicaciones. Siendo una mejor opción cuando es necesario realizar numerosos cálculos o procesos de CPU muy intensos.

No todas las aplicaciones son más factibles hacerlas Web, hay cosas que se necesita que se ejecuten estrictamente en su máquina para aprovechar el poder que tiene al alcance.

1.12 Conclusiones

En este capítulo después de haber realizado un estudio acerca de los sistemas existentes a nivel internacional, y de determinar los principales elementos que los caracterizan, se definieron los posibles aspectos a tener en cuenta para desarrollar la propuesta de solución, dentro de los cuales se encuentra mantener la tecnología que se aplica en la Universidad (código de barras), ya que es mucho más económica, sin dejar de reconocer las ventajas que pudiera traer para la Universidad la utilización de un equipamiento que pueda hacer mucho más eficiente los procesos que se realizan. Además se describieron tendencias y tecnologías, así como las principales características de la metodología, lenguajes y herramienta.

Capítulo 2: Características del Sistema

2.1 Introducción

En este capítulo se tratarán las características del sistema a desarrollar. Se verá como se realiza el flujo actual de procesos en la Universidad. Además se modela el negocio propuesto, identificándose los actores, trabajadores y los casos de uso correspondientes y se representa el modelo de objetos. Se plantean los requerimientos funcionales y no funcionales de la aplicación a desarrollar y se describe la solución propuesta en términos de casos de uso del sistema.

2.2 Flujo Actual de Procesos involucrados en el campo de acción

Los procesos que ocurren actualmente en el control de acceso del personal, vehículos y visitantes se describen de la siguiente manera:

+ Control de Acceso de Personas de la UCI.

En este proceso la persona ya sea estudiante o trabajador de la UCI, solicita la entrada o salida de la misma. Se le verifica la credencial mirando su foto, en caso que no la posea se hace una búsqueda en el directorio para comprobar si la persona pertenece o no. Si está saliendo se le chequean las pertenencias, en caso de alguna violación se retiene.

+ Control de Acceso de Vehículos.

En este proceso el chofer solicita la entrada o salida a la Universidad. Se verifica que el vehículo sea de la UCI, se registra en una planilla los datos del vehículo (chapa), y la hora de cada uno de los accesos. En caso de no pertenecer se le deniega la entrada. Si el vehículo está saliendo se le hace una revisión, en caso de alguna violación se retiene.

+ Control de Acceso de Visitantes.

En este proceso el visitante que es una persona que no pertenece a la UCI, se le debe entregar un permiso para poder acceder a la misma y se realiza de la siguiente manera: solicita la entrada a la UCI, se verifica si está autorizado y se le confecciona un pase de visitante. En caso de no ser autorizado se le deniega la entrada. A la hora de salida se le chequean las pertenencias, en caso de alguna violación se le retiene.

2.3 Objeto de automatización

Se desean automatizar todos los procesos necesarios para un óptimo funcionamiento en el control de acceso. Los principales procesos que requieren de automatización son:

- Controlar acceso puntual de personas de la UCI.
- Controlar accesos de vehículos.
- Controlar accesos de visitantes.
- Actualizar Lector.
- Actualizar Base de Datos Local.
- Configurar Sistema.

2.4 Información que se maneja

Existen documentos que se recogen y archivan en el control de acceso, están los documentos de pase de visitante y el libro de vehículos.

- **Pase de visitante:** Recoge los datos a tener en cuenta para permitirle la entrada del visitante (datos personales, fecha de entrada/salida, lugar a visitar y dirigente que lo autoriza).
- **Registro de vehículos:** Recoge los datos del vehículos (chapa, fecha de entrada/salida) en cada uno de los accesos.

2.5 Reglas del negocio a considerar

- ✓ Ninguna persona o vehículo de la UCI que no posea credencial puede entrar a la Universidad si no es autorizado por un directivo.
- ✓ Ninguna persona puede extraer un medio básico de la Universidad si no tiene un autorizo.
- ✓ Todas las personas para salir de la Universidad deben mostrar la credencial y si es estudiante mostrar pase de autorización por directivo.
- ✓ Todos los visitantes deben entregar documentos personales y si viene en vehículo los documentos de este para poder acceder a la Universidad.
- ✓ Todos los visitantes tienen que poseer un autorizo para poder entrar a la Universidad.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

- ✓ El agente SEPCOM debe chequear las pertenencias de todas las personas que salgan de la Universidad.
- ✓ El agente SEPCOM debe chequear los vehículos que salgan de la Universidad.

2.6 Modelo de negocio

El Modelo de Negocio es el primer flujo de trabajo que propone la metodología RUP. Es importante destacar que es una técnica para la especificación de los requisitos más importantes del sistema, que da soporte al negocio, con lo cual se refuerza la idea de que sea el propio negocio lo que determine los requisitos.

El objetivo de modelar el negocio fue comprender la estructura y la dinámica de la organización para la cual se implementaría el sistema, comprender sus problemas actuales y además asegurar que los consumidores, usuarios finales y desarrolladores tuviesen un entendimiento común de la misma para lograr obtener los requerimientos del sistema.

2.6.1 Descripción de los Actores y Trabajadores del Negocio

Actores del Negocio.

Un actor del negocio es cualquier individuo, grupo, organización, máquina o sistema de información externo que interactúa con el negocio. El término actor significa el rol que algo o alguien juega cuando interactúa con el negocio para beneficiarse de sus resultados.

Nombre del Actor	Descripción
Persona	Es el que solicita la entrada o salida a la Universidad.
Chofer	Es el que solicita la entrada o salida a la Universidad.
Visitante	Es el que solicita la entrada o salida a la Universidad.

Tabla 1 Actores del negocio.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

Trabajadores del Negocio.

Un trabajador define el comportamiento y las responsabilidades de un individuo que actúa en el negocio realizando una o varias actividades, interactuando con otros trabajadores del negocio y manipulando entidades del negocio.

Nombre del Trabajador	Descripción
Agente SEPCOM	Agentes de Seguridad y Protección en la UCI. Son los encargados de velar por la Seguridad interna de la Universidad, chequear credenciales de las personas que entran y salen diariamente, y velar porque no se extraiga ningún medio indebido de esta.
Técnico SP	Técnico de Seguridad y Protección. Son los encargados de realizar el autorizo de entrada a visitas que lleguen a la Universidad, verificando si existe un autorizo previo. Responsable del control de los medios básicos y controla el trabajo de los responsables de seguridad.
Directivo	Persona que trabaja en la UCI con cargo en la Dirección de la Universidad, pertenece a determinada área, puede informar las personas que residen en ella, autoriza a personas a visitar la misma y autoriza la entrada de medios.

Tabla 2 Trabajadores del negocio.

2.6.2 Descripción de los Casos de Uso del Negocio

Caso de Uso: "Entrar Persona".

Caso de Uso:	Entrar Persona
Actores:	Persona (Inicia)
Trabajadores	Agente SEPCOM, Técnico SP, Directivo.
Resumen:	El caso de uso se inicia cuando la persona solicita la entrada a la Universidad, el Agente SEPCOM le pide la credencial, la verifica y autoriza

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

	la entrada de la persona.
--	---------------------------

Tabla 3 Descripción Caso de Uso del Negocio: “Entrar Persona”.

Caso de Uso: “Salir Persona”.

Caso de Uso:	Salir Persona
Actores:	Persona(Inicia)
Trabajadores	Agente SEPCOM, Técnico SP, Directivo.
Resumen:	El caso de uso se inicia cuando la persona solicita la salida de la Universidad, el Agente SEPCOM le pide la credencial, la verifica, revisa pertenencias, si es estudiante le solicita el pase también y autoriza la salida de la persona.

Tabla 4 Descripción Caso de Uso del Negocio: “Salir Persona”.

Caso de Uso: “Entrar Vehículo”.

Caso de Uso:	Entrar Vehículo
Actores:	Chofer(Inicia)
Trabajadores	Agente SEPCOM, Técnico SP, Directivo.
Resumen:	El caso de uso se inicia cuando el chofer solicita la entrada a la Universidad, el Agente SEPCOM le pide la credencial; verifica la credencial y registra datos del vehículo y autoriza la entrada.

Tabla5 Descripción Caso de Uso del Negocio: “Entrar Vehículo”.

Caso de Uso: “Salir Vehículo”.

Caso de Uso:	Salir Vehículo
Actores:	Chofer(Inicia)
Trabajadores	Agente SEPCOM, Técnico SP, Directivo.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

Resumen:	El caso de uso se inicia cuando el chofer solicita la salida de la Universidad, el Agente SEPCOM le pide la credencial; verifica la credencial y registra datos del vehículo, revisa el vehículo y autoriza la salida.
-----------------	--

Tabla6 Descripción Caso de Uso del Negocio: “Salir Vehículo”.

Caso de Uso: “Entrar Visitante”.

Caso de Uso:	Entrar Visitante
Actores:	Visitante(Inicia)
Trabajadores	Agente SEPCOM, Técnico SP, Directivo.
Resumen:	El caso de uso se inicia cuando el visitante solicita la entrada a la Universidad, el Técnico SP le pide el pase y lo registra y si el visitante no posee el pase, el Directivo autoriza y el Técnico SP confecciona autorizo de entrada.

Tabla7 Descripción Caso de Uso del Negocio: “Entrar Visitante”.

Caso de Uso: “Salir Visitante”.

Caso de Uso:	Salir Visitante
Actores:	Visitante(Inicia)
Trabajadores	Agente SEPCOM, Técnico SP, Directivo.
Resumen:	El caso de uso se inicia cuando el visitante solicita la salida de la Universidad, el Técnico SP le pide el pase y lo verifica, revisa las pertenencias y autoriza la salida.

Tabla8 Descripción Caso de Uso del Negocio: “Salir Visitante”.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

Caso de Uso: “Informar Medios”.

Caso de Uso:	Informar Medios
Actores:	Actor
Trabajadores	Agente SEPCOM, Técnico SP, Directivo.
Resumen:	El caso de uso se inicia cuando el Agente SEPCOM solicita el autorizo del medio o los datos para autorizar la entrada del mismo.

Tabla9 Descripción Caso de Uso del Negocio: “Informar Medios”.

2.6.3 Diagrama de actividades para casos de uso del negocio

Un diagrama de actividades es un tipo especial de diagrama de estados que muestra el flujo de actividades dentro del negocio. Donde participan los actores/trabajadores del negocio, las actividades a realizar por ellos y las relaciones entre estas. Debido al gran tamaño que adoptan los diagramas de actividades, no es posible ejemplificarlos en este documento, ya que se verá afectada la legibilidad de los mismos.

2.6.4 Diagrama de Casos de Uso del Negocio

Representa gráficamente los procesos del negocio y su interacción con los actores del negocio. A continuación se muestra la figura correspondiente a este diagrama:

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

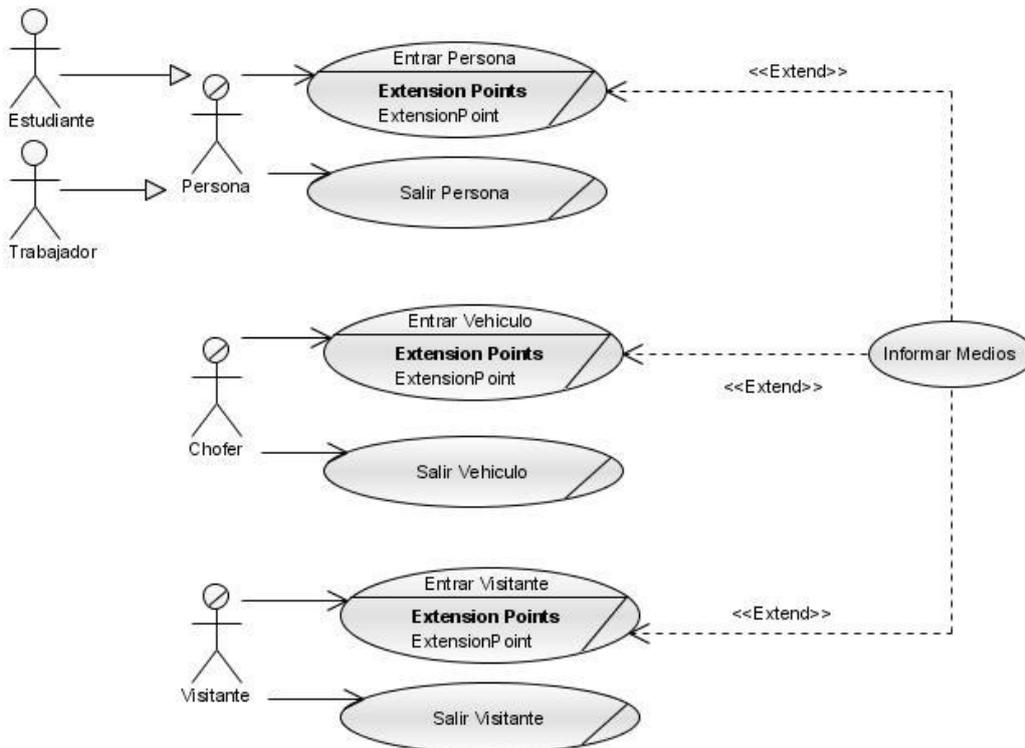


Figura 2 Diagrama de Casos de Uso del Negocio.

2.6.5 Diagrama de objetos del Negocio

Un modelo de objetos del negocio es un modelo interno a un negocio. Describe cómo colaboran los trabajadores y las entidades del negocio dentro del flujo de trabajo del proceso de negocio.

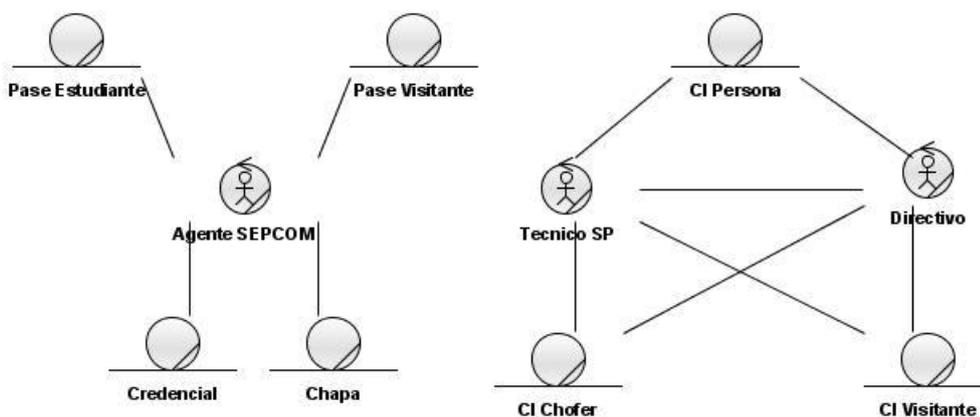


Figura 3 Diagrama de objetos del Negocio.

2.7 Requerimientos

Un requerimiento es una condición o capacidad necesaria para que un usuario resuelva un problema o alcance un objetivo. A continuación se describen los requerimientos funcionales y no funcionales del sistema.

2.7.1 Requerimientos funcionales

Los requerimientos funcionales son aquellos requisitos que, desde el punto de vista de las necesidades del usuario, debe cumplir el sistema. Para cumplir con los objetivos propuestos se plantean las siguientes funcionalidades:

RF-1 Autenticar usuario.

RF-1.1 Verificar Usuario y contraseña.

RF-2 Controlar acceso puntual de personas de la UCI.

RF-2.1 Verificar la validez de las credenciales de personas de la UCI.

RF-2.2 Registrar las entradas de personas de la UCI.

RF-2.3 Registrar las salidas de personas de la UCI.

RF-2.4 Eliminar las entradas de personas.

RF-2.5 Eliminar las salidas de personas.

RF-3 Controlar accesos de vehículos.

RF-3.1 Verificar la validez de los vehículos.

RF-3.2 Registrar las entradas de los vehículos.

RF-3.3 Registrar las salidas de los vehículos.

RF-3.4 Eliminar las entradas de vehículos.

RF-3.5 Eliminar las salidas de vehículos.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

RF-4 Controlar accesos de visitantes.

RF-4.1 Registrar visitante.

RF-4.2 Registrar las entradas de visitantes.

RF-4.3 Registrar las salidas de visitantes.

RF-4.4 Eliminar las entradas de visitantes.

RF-4.5 Eliminar las salidas de visitantes.

RF-5 Verificar PassBack.

RF- 5.1 Verificar PassBack.

RF- 5.2 Registrar motivo de PassBack.

RF-6 Mostrar Medios Autorizados.

RF -7 Mostrar Avisos.

RF-8. Configurar el sistema.

RF-8.1 Configurar tiempos de sincronización de las BD.

RF-8.2 Visualizar datos de configuración del sistema.

RF-9. Actualizar Lector.

RF-10. Actualizar Base de Datos Local.

2.7.2 Requerimientos no funcionales

Los requerimientos no funcionales se consideran los atributos del sistema, propiedades o cualidades que debe tener el producto.

- **Requerimientos de apariencia o interfaz externa:** El sistema debe tener un ambiente amigable y entendible para los usuarios finales, de forma tal que no les sea muy complicado utilizar el software.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

- **Requerimientos de usabilidad:** El sistema podrá ser usado por personas autorizadas, la navegabilidad no debe ser muy compleja, todas las funcionalidades deben ser rápidamente accesibles por el usuario.
- **Requerimientos de rendimiento:** El tiempo de respuesta de una petición al servidor debe ser rápido ya que el sistema operará con grandes cantidades de datos.
- **Requerimientos de soporte:** Se le debe dar mantenimiento periódicamente a los servidores de bases de datos controlando la integridad de la información.
- **Requerimientos de portabilidad:** Debe tener facilidad para adaptarlo a diferentes ambientes. Independencia de la plataforma.
- **Requerimientos de seguridad y privacidad.**
 - ✓ Identificar al usuario antes de que pueda realizar cualquier acción sobre la configuración del sistema.
 - ✓ Garantizar que la información sea vista solo por personas autorizadas.
 - ✓ Verificación sobre acciones irreversibles (eliminaciones).
 - ✓ Garantía de que el sistema funcione correctamente aún cuando no haya conectividad.
- **Requerimientos de confiabilidad:** Se debe chequear la integridad de los datos.
- **Requerimientos 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.
- **Requerimientos de Ayuda y documentación en línea:** Se debe brindar un sistema de ayuda que explique las diferentes funcionalidades con que cuenta el sistema, además los manuales de usuario.
- **Restricciones en el diseño y la implementación:** Para el análisis y el diseño del sistema debe ser utilizada la metodología RUP, usando el lenguaje de modelado UML y como herramienta para llevarlo a cabo el Visual Paradigm.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

- **Requerimientos de software:** Se debe disponer en el servidor con Windows XP, Windows 2000 Server o 2000 Advanced Server. Se utilizará como lenguaje de programación: Java y como Base de Datos: DB4o.
- **Requerimientos de hardware:** Requiere estar instalada en una PC Pentium, 256 Mb de RAM (como mínimo), Micro 2.0 o superior, 280 Mb de disco duro, una tarjeta de red de 100Mbps y el procesador de 133 MHz o superior.

2.8 Definición de los Casos de Uso del Sistema

Un caso de uso no es más que una secuencia de actividades que realiza un sistema y que da como resultado un valor para el actor. Estos han alcanzado un uso universal debido a dos razones básicas, la primera de ellas es que proporcionan un medio intuitivo y sistemático de capturar los requisitos anteriormente mencionados, centrándose en lo que quiere obtener el cliente, y la segunda es que dirigen todo el proceso apreciando que el análisis, diseño y prueba se realizan partiendo de los casos de uso.

Es de vital importancia realizar una buena selección de los casos de uso debido que el proceso de desarrollo está guiado por ellos, lo que se traduce en que, una serie de flujos de trabajo se inicia a partir de los mismos.

2.8.1 Actores del sistema

Los actores del sistema pueden representar el rol que juega una o varias personas, un equipo o un sistema automatizado, son parte del sistema, y pueden intercambiar información con él o ser recipientes pasivos de información. En este caso los actores que interactúan con el sistema se definen en la siguiente tabla.

Actor	Descripción
Técnico SP	Es el encargado de gestionar todo el acceso del personal, vehículos, visitantes y medios en la UCI.

Tabla10 Actores del sistema.

2.8.2 Diagrama de Casos de Uso del Sistema

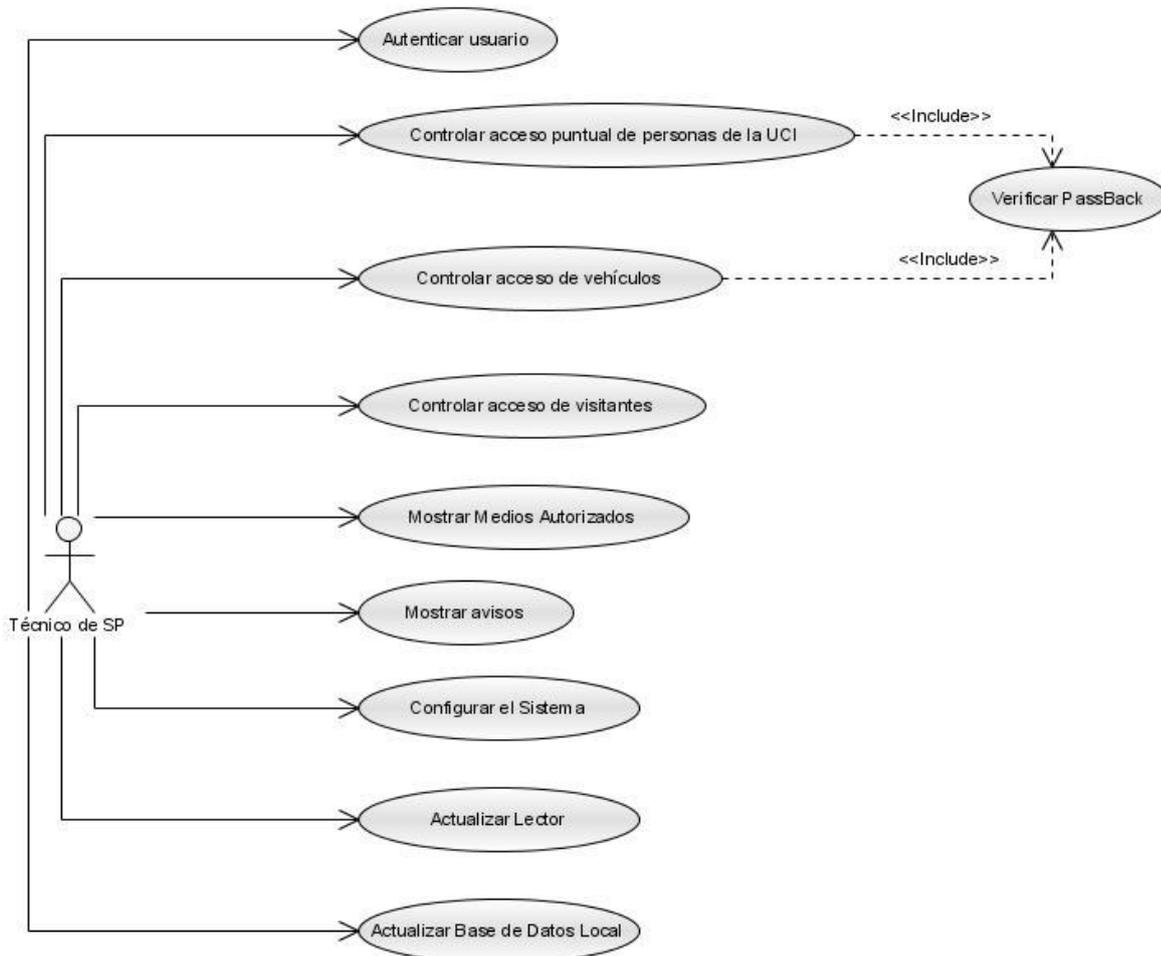


Figura 4 Diagrama de Casos de Uso del Sistema.

2.8.3 Descripción de los casos de uso del sistema

Dentro de los casos de uso del sistema se encuentran autenticar usuario, verificar passback, mostrar medios autorizados, mostrar avisos, controlar acceso puntual de personas de la UCI, controlar acceso de vehículos, controlar acceso de visitantes, configurar sistema, actualizar lector y actualizar base de datos local, a continuación se describen los principales procesos.

Caso de Uso: “Controlar acceso puntual de personas de la UCI”.

Caso de Uso:	Controlar acceso puntual de personas de la UCI
Actores:	Técnico de Seguridad y protección (inicia)

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

Resumen:	El caso de uso se inicia cuando el Técnico de SP solicita controlar el acceso puntual de una persona perteneciente a la UCI, verifica la validez de los datos y registra la entrada o salida.
Precondiciones:	El Técnico de SP debe estar autenticado en el sistema.
Referencias	RF 2, RF 2.1, RF 2.2, RF 2.3, RF 2.4, RF 2.5
Prioridad	Alta
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1- El caso de uso se inicia cuando el Técnico SP selecciona en la ventana principal la opción "Persona" para controlar el acceso puntual.	2- Muestra la interfaz con la opción "Auto Registrar", que contiene las opciones de "Entrada", "Salida" (desactivadas) y "Deshacer". Y la opción "Buscar" desactivada que contiene las opciones "Entrada", "Salida", desactivadas y "Deshacer" y los campos: <ul style="list-style-type: none"> • Solapín. • Cl. Y los datos vacíos: <ul style="list-style-type: none"> • Solapín. • Nombre. • Tipo. • Cl. • Estado. • Fecha de vencimiento (Eventual). • Foto.
3- Selecciona la opción deseada.	4- Ejecuta una de las siguientes acciones: <ul style="list-style-type: none"> a) Auto-Registrar. Ver Sección "Auto-Registrar". b) Buscar. Ver Sección "Buscar". c) Deshacer. Ver Sección "Deshacer".
Secciones del Flujo Básico	
Acción del Actor	Respuesta del Sistema

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

Sección “Auto-Registrar”	
<p>2- Selecciona el tipo de acceso.</p> <p>3- Introduce el código de barras.</p>	<p>1- Activa las opciones “Entrada” y “Salida”</p> <p>4- Captura el código de barras.</p> <p>5 Comprueba que el dato capturado (código de barras) pertenece a una persona de la UCI. En caso contrario ver FA 1.</p> <p>6- Verifica PassBack. Ver CU “Verificar PassBack”.</p> <p>7- Actualiza el estado de la persona.</p> <p>8- Muestra los datos.</p> <ul style="list-style-type: none"> • Solapín. • Nombre. • Tipo. • CI. • Estado. • Fecha de vencimiento (Eventual). • Foto. <p>9- Guarda la última fecha en que se realizó el acceso y el identificador.</p>
Sección “Buscar”	
<p>1- Introduce el dato por el cual desea realizar la búsqueda (solapín o CI).</p> <p>2- Selecciona la opción “Buscar”.</p>	<p>3- Comprueba que el dato introducido pertenece a una persona de la UCI. En caso contrario ver FA 1.</p> <p>4- Muestra los datos.</p> <ul style="list-style-type: none"> • Solapín. • Nombre. • Tipo. • CI. • Estado. • Fecha de vencimiento (Eventual). • Foto. <p>5- Activa las opciones “Entrada” y “Salida”.</p>
<p>6- Selecciona el tipo de acceso.</p>	<p>7- Verifica PassBack. Ver CU “Verificar</p>

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

	<p>PassBack”.</p> <p>8- Actualiza el estado de la persona.</p> <p>9- Actualiza el campo estado de la interfaz.</p> <p>10- Guarda la última fecha en que se realizó el acceso y el identificador.</p>
Sección “Deshacer”	
<p>2- Selecciona la opción “Aceptar”. En caso contrario ver FA3.</p>	<p>1- Muestra un mensaje de verificación, indicando si desea cambiar el último acceso con las opciones “Aceptar” y “Cancelar”. En caso contrario ver FA2.</p> <p>3- Elimina el último acceso registrado.</p> <p>4- Va al paso 4 del flujo normal de eventos.</p>
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
FA1	
	<p>1-Muestra un mensaje de error indicando que la persona no se encontró, finalizando así el caso de uso.</p>
FA2	
	<p>1- Muestra un mensaje indicando que no hay ningún acceso registrado.</p>
FA3	
<p>1- Selecciona la opción “Cancelar”.</p>	<p>3- Va al paso 4 del flujo normal de eventos.</p>
Prototipo de Interfaz	

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA



Poscondiciones	Se registró el acceso de personas de la UCI.
-----------------------	--

Tabla11 Descripción del caso de uso “Controlar acceso puntual de personas de la UCI”.

Caso de Uso: “Controlar acceso de vehículo”.

Caso de Uso:	Controlar accesos de vehículos.	
Actores:	Técnico de Seguridad y protección (inicia)	
Resumen:	El caso de uso se inicia cuando el Técnico de SP solicita controlar el acceso de vehículos, verifica la validez de los datos y registra la entrada o salida.	
Precondiciones:	El Técnico de SP debe estar autenticado en el sistema.	
Referencias	RF 3, RF 3.1, RF 3.2, RF 3.3, RF 3.4, RF 3.5	
Prioridad	Alta	
Flujo Normal de Eventos		
	Acción del Actor	Respuesta del Sistema

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

<p>1- El caso de uso inicia cuando el Técnico SP selecciona en la ventana principal la opción "Vehículo" para controlar el acceso.</p>	<p>2- Muestra la interfaz que contiene un campo: chapa; la opción "Buscar", y las opciones "Entrada", "Salida" (desactivadas) y "Deshacer".</p> <p>Y los datos vacíos:</p> <ul style="list-style-type: none"> • Chapa. • Color. • Marca. • Foto del vehículo. • Foto del chofer. • Nombre del chofer.
<p>3- Selecciona la opción deseada.</p>	<p>4- Ejecuta una de las siguientes acciones:</p> <p>a) Buscar. Ver Sección "Buscar".</p> <p>b) Deshacer. Ver Sección "Deshacer".</p>
Secciones del Flujo Básico	
Acción del Actor	Respuesta del Sistema
Sección "Buscar".	
<p>1- Introduce el dato por el cual desea realizar la búsqueda (chapa).</p> <p>2- Selecciona la opción "Buscar".</p>	<p>3- Comprueba que la chapa pertenece a un vehículo. En caso contrario ver FA1</p> <p>4- Muestra los datos.</p> <ul style="list-style-type: none"> • Chapa. • Color. • Marca. • Foto del vehículo. • Foto del chofer. • Nombre del chofer. <p>5- Activa las opciones "Entrada y Salida".</p>
<p>6- Selecciona el tipo de acceso.</p>	<p>7- Verifica PassBack. Ir al CU "Verificar PassBack".</p> <p>8- Actualiza el estado del vehículo.</p> <p>9- Actualiza el campo estado de la interfaz.</p> <p>10- Guarda la última fecha en que se realizó el acceso y el identificador.</p>

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

Sección “Deshacer”	
2- Selecciona la opción “Aceptar”. En caso contrario ver FA3.	1- Muestra un mensaje de verificación, indicando si desea cambiar el último acceso con las opciones “Aceptar” y “Cancelar”. En caso contrario ver FA2. 3- Elimina el último acceso registrado. 4- Va al paso 4 del flujo normal de eventos.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
FA1	
	1-Muestra un mensaje de error indicando que el vehículo no se encontró, finalizando así el caso de uso.
FA2	
	1- Muestra un mensaje indicando que no hay ningún acceso registrado.
FA3	
1- Selecciona la opción “Cancelar”.	3- Va al paso 4 del flujo normal de eventos.
Prototipo de Interfaz	

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

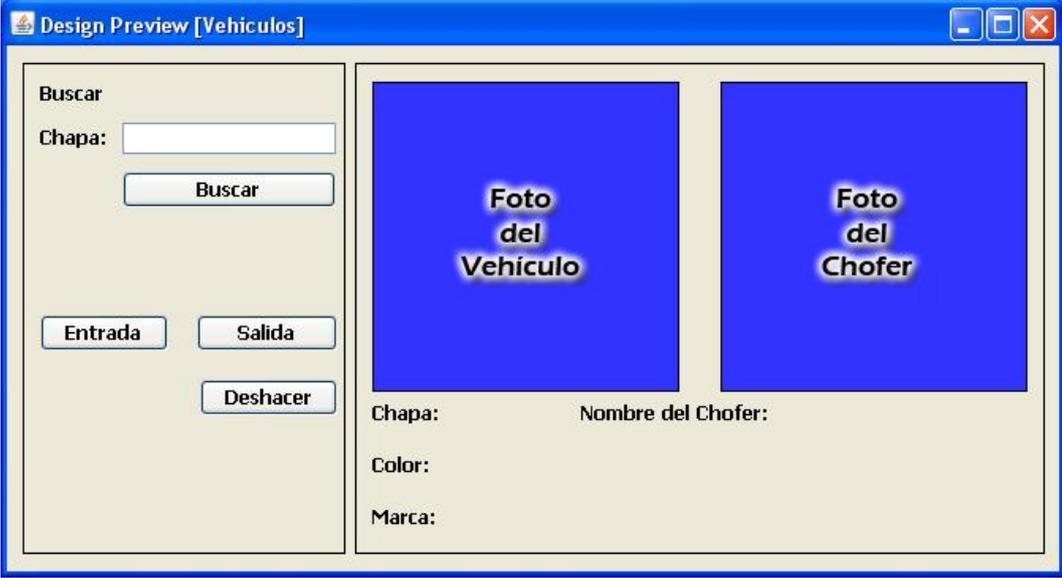
	
Poscondiciones	Se registró el acceso de vehículos.

Tabla12 Descripción del caso de uso “Controlar accesos de vehículos”.

Caso de Uso: “Controlar accesos de visitantes”.

Caso de Uso:	Controlar accesos de visitantes.	
Actores:	Técnico de Seguridad y protección (inicia)	
Resumen:	El caso de uso se inicia cuando el Técnico de SP solicita controlar el acceso de visitantes, en caso de que no esté autorizado se introducen los datos del mismo; y se registra la entrada o salida.	
Precondiciones:	El Técnico de SP debe estar autenticado en el sistema.	
Referencias	RF 4, RF 4.1, RF 4.2, RF 4.3, RF 4.4, RF 4.5	
Prioridad	Alta	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1- El caso de uso inicia cuando el Técnico SP selecciona en la ventana principal la opción “Visitantes” para controlar el acceso.	2- Muestra la interfaz que contiene las opciones “Buscar” con el campo CI (vacío) y “Registrar Visitantes” con las opciones “Aceptar” y “Cancelar” y las opciones	

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

	“Entrada”, “Salida” (desactivadas) y “Deshacer”.
3- Selecciona la opción deseada.	4-Ejecuta una de las siguientes acciones: a) Buscar. Ver Sección “Buscar”. b) Registrar Visitante. Ver Sección “Registrar Visitante”. c) Deshacer. Ver Sección “Deshacer”.
Secciones del Flujo Básico	
Acción del Actor	Respuesta del Sistema
Sección “Buscar”.	
1- Introduce el dato por el cual desea realiza la búsqueda (CI). 2- Selecciona la opción “Buscar”.	3- Comprueba que el CI pertenece a un visitante autorizado. En caso contrario ver FA1. 4- Muestra los siguientes datos: <ul style="list-style-type: none"> • Nombre y Apellidos. • No CI. • Organismo. • Lugar a visitar. • Dirigente que autoriza. • Paquetes. • Fecha de entrada. • Fecha de salida. • Nombre del Chofer • CI del Chofer • Licencia de Conducción. • Circulación del carro. • Marca • Chapa del vehículo. • Tipo de Vehículo. 5- Activa las opciones “Entrada” y “Salida”.
6- Selecciona el tipo de acceso.	7- Guarda la última fecha en que se realizó el acceso y el identificador.
Sección “Registrar Visitante”.	
	1- Muestra los siguientes campos (vacíos):

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

<p>2- Llena los datos del visitante. 3- Selecciona la opción "Posee Vehículo". En caso contrario ver FA2.</p>	<ul style="list-style-type: none"> • Nombre y Apellidos. • No CI. • Organismo. • Lugar a visitar. • Dirigente que autoriza. • Paquetes. • Fecha de entrada. • Fecha de salida. <p>Y la opción "Posee Vehículo".</p> <p>4- Muestra los campos:</p> <ul style="list-style-type: none"> • Nombre del Chofer • CI del Chofer • Licencia de Conducción. • Circulación del carro. • Marca • Chapa del vehículo. • Tipo de Vehículo.
<p>5- Introduce los datos. 6- Selecciona la opción "Aceptar". En caso contrario ver FA 3.</p>	<p>7- Guarda los datos del visitante. 8- Activa las opciones "Entrada" y "Salida".</p>
<p>9- Selecciona el tipo de acceso.</p>	<p>10- Guarda la última fecha en que se realizó el acceso y el identificador.</p>
Sección "Deshacer"	
<p>2- Selecciona la opción "Aceptar". En caso contrario ver FA3.</p>	<p>1- Muestra un mensaje de verificación, indicando si desea cambiar el último acceso con las opciones "Aceptar" y "Cancelar". En caso contrario ver FA4. 3- Elimina el último acceso registrado. 4- Va al paso 4 del flujo normal de eventos.</p>
Flujos Alternos	
Acción del Actor	Respuesta del Sistema

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

FA1	
	1-Muestra un mensaje de error indicando que el visitante no se encontró, finalizando así el caso de uso.
FA2	
	1- Va al paso 7 de la Sección “Registrar Visitante”.
FA3	
	1- Va al paso 4 del flujo normal de eventos.
FA4	
	1- Muestra un mensaje indicando que no hay ningún acceso registrado.

Prototipo de Interfaz

Controlar Accesos de Visitantes

Opciones:

Buscar Visitante

CI del Visitante:

Registrar Visitantes

Datos del Visitante

- Nombre y Apellidos
- No CI
- Organismo.
- Lugar a visitar
- Dirigente que autoriza
- Fecha de entrada
- Fecha de salida
- Paquetes

Datos si presenta Chofer

- Nombre del Chofer
- CI del Chofer
- Licencia de Conducción
- Circulación del carro
- Marca
- Chapa del vehículo
- Tipo de Vehículo.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

Poscondiciones	Se registró el acceso de visitantes.

Tabla13 Descripción del caso de uso “Controlar accesos de visitantes”.

Caso de Uso: “Actualizar Lector”.

Caso de Uso:	Actualizar Lector
Actores:	Técnico de Seguridad y protección (inicia)
Resumen:	El caso de uso inicia cuando el Técnico SP solicita actualizar el lector, donde son introducidos los códigos de barra actualizados. Finalizando así el caso de uso.
Precondiciones:	El Scanner debe ser conectado a un terminal de PC que contenga la BD local de personas, vehículos, y que contenga la aplicación. Debe existir una aplicación encargada de subir información al lector portátil.
Referencias	RF 9
Prioridad	Alta.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1- El caso de uso inicia cuando el Técnico SP selecciona en la ventana principal la opción “Actualizar Lector”.	2- Muestra la interfaz que contiene las opciones “Actualizar Optimus S” y “Cancelar”. Y dos listados (personas o

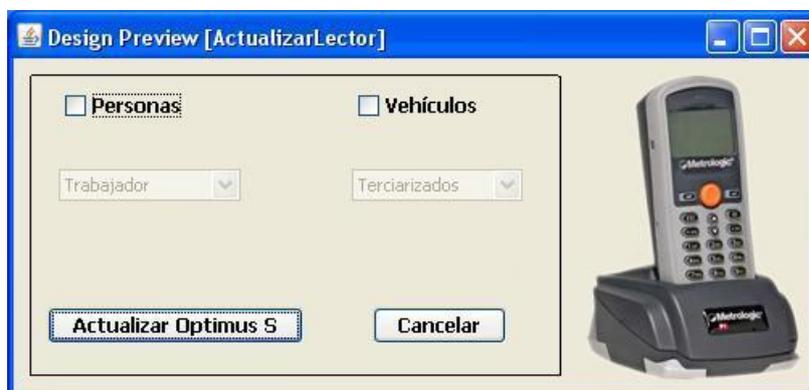
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

	vehículos) para seleccionar el dato por el cual será actualizado el lector.
3- Selecciona la opción "Actualizar Optimus S". En caso contrario ver FA1.	4- Muestra una interfaz que contiene: <ul style="list-style-type: none"> • Una opción para pasar al Optimus los códigos de barra. (Permite buscar el fichero que contiene los códigos de barra) • Tipo de plataforma (escoger) • Puerto COM (escoger) • Velocidad de Trasmisión (escoger) Y las opciones "Aceptar" y "Cancelar".
5- Selecciona las opciones con los valores correctos para actualizar en lector. 6- Selecciona la opción "Aceptar". En caso contrario ver FA1.	7- Actualiza el lector. Finalizando así el caso de uso.

Flujos Alternos

Acción del Actor	Respuesta del Sistema
FA1	
1- Selecciona la opción "Cancelar".	2- Va a la ventana principal.

Prototipo de Interfaz



CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

	
Poscondiciones	Quedan registrados los códigos de barra de personas o vehículos en el lector Optimus S.

Tabla14 Descripción del caso de uso “Actualizar Lector”.

Caso de Uso: “Actualizar Base de Datos Local”.

Caso de Uso:	Actualizar Base de Datos Local.	
Actores:	Técnico de Seguridad y protección (inicia)	
Resumen:	El caso de uso inicia cuando el Técnico SP solicita actualizar la Base de Datos local, donde se registran las entradas o salidas según la causa. Finalizando así el caso de uso.	
Precondiciones:	El Scanner debe ser conectado a un terminal de PC que contenga la BD local de personas, vehículos, y que contenga la aplicación. Debe existir una aplicación encargada de bajar información del lector portátil.	
Referencias	RF 11	
Prioridad	Alta	
Flujo Normal de Eventos		
	Acción del Actor	Respuesta del Sistema
	1- El caso de uso inicia cuando el Técnico SP	2- Muestra la interfaz que contiene las

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

selecciona en la ventana principal la opción “Actualizar Base de Datos local”.	opciones “Entrada” y “Salida”. Y un listado para seleccionar la causa del acceso.
3- Selecciona o introduce la causa del acceso. 4- Selecciona el tipo de acceso.	5- Guarda en la Base de Datos el tipo de acceso y la fecha en que ocurrió. Finalizando así el caso de uso.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
Prototipo de Interfaz	
	
Poscondiciones	Queda registrada la entrada o salida de la persona o vehículo junto con la fecha y hora en la BD local.

Tabla15 Descripción del caso de uso “Actualizar BD Local”.

Caso de Uso: “Configurar Sistema”.

Caso de Uso:	Configurar el Sistema.
Actores:	Técnico de Seguridad y protección (inicia)
Resumen:	El caso de uso inicia cuando el Técnico de SP accede a la configuración y se muestra una ventana para actualizar o cambiar los parámetros de configuración como tiempo para actualizar las BDs, nombre de la puerta, etc. Finalizando así el caso de uso.
Precondiciones:	El Técnico de SP debe estar autenticado en el sistema.
Referencias	RF 8, RF 8.1, RF 8.2

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

Prioridad	Alta
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1- El caso de uso inicia cuando el Técnico SP selecciona en la ventana principal la opción “Configurar Sistema”.	<p>2- Muestra una interfaz con las opciones:</p> <ul style="list-style-type: none"> • Sincronización de Accesos de Personas. • Sincronización de Accesos de Vehículos. • Sincronización de Accesos de Visitantes. • Sincronización del listado local de Personas. • Sincronización del listado local de Vehículos. • Sincronización del listado local de Visitantes. • Sincronización automática. <p>Con los campos para introducir los tiempos de las sincronizaciones. Y muestra la última fecha en que se hizo la sincronización, Nombre de puerta.</p> <p>Y las opciones “Aceptar” y “Cancelar”.</p>
3- Selecciona la opción deseada.	<p>4-Ejecuta una de las siguientes acciones:</p> <p>a) Sincronización de Accesos de Personas. Ver Sección “Sincronización de Accesos de Personas”.</p> <p>b) Sincronización de Accesos de Vehículos. Ver Sección “Sincronización de Accesos de Vehículos”.</p> <p>c) Sincronización de Accesos de Visitantes. Ver Sección “Sincronización de Accesos de Visitantes”.</p> <p>d) Sincronización del listado local de Personas. Ver Sección “Sincronización del listado local de Personas”.</p> <p>e) Sincronización del listado local de Vehículos. Ver Sección “Sincronización del listado local de Vehículos”.</p>

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

	<p>f) Sincronización del listado local de Visitantes. Ver Sección “Sincronización del listado local de Visitantes”.</p> <p>g) Sincronización automática. Ver Sección “Sincronización Automática”.</p>
Sección “Sincronización de Accesos de Personas”	
Acción del Actor	Respuesta del Sistema
2- Selecciona la opción “Aceptar”. En caso contrario ver FA1.	1- Sube los accesos de las persona de la BD local a la BD remota.
Sección “Sincronización de Accesos de Vehículos”	
Acción del Actor	Respuesta del Sistema
2- Selecciona la opción “Aceptar”. En caso contrario ver FA1.	1- Sube los accesos de los vehículos de la BD local a la BD remota.
Sección “Sincronización de Accesos de Visitantes”	
2- Selecciona la opción “Aceptar”. En caso contrario ver FA1.	1- Sube los accesos de los visitantes de la BD local a la BD remota.
Sección “Sincronización del listado local de Personas”	
Acción del Actor	Respuesta del Sistema
2- Selecciona la opción “Aceptar”. En caso contrario ver FA1.	1-Baja el listado de personas de la BD remota a la BD local.
Sección “Sincronización del listado local de Vehículos”	
Acción del Actor	Respuesta del Sistema
2- Selecciona la opción “Aceptar”. En caso contrario	1- Baja el listado de vehículos de la BD remota a la BD local.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

ver FA1.	
Sección “Sincronización del listado local de Visitantes”	
2- Selecciona la opción “Aceptar”. En caso contrario ver FA1.	1- Baja el listado de visitantes de la BD remota a la BD local.
Sección “Sincronización automática”	
1- Introduce el tiempo en el que desea realizar las actualizaciones automáticas. 2- Selecciona la opción “Aceptar”. En caso contrario ver FA1.	3- Guarda en la BD local los tiempos de sincronización introducidos. Finalizando así el caso de uso.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
FA1	
1- Selecciona la opción “Cancelar”.	1- Va a la ventana principal.
Prototipo de Interfaz	

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

Poscondiciones	Queda configurado el sistema.

Tabla 16 Descripción del caso de uso “Configurar Sistema”.

2.9 Propuesta de sistema

Para solucionar los problemas existentes en la Universidad se propone realizar el análisis y diseño de un sistema informático, con el propósito de elevar la calidad de todos los procesos que se desarrollen en este sentido en la UCI. Estos procesos permiten controlar las entradas y salidas del personal, vehículos y visitantes a la Universidad, la autenticación del usuario, se verifica si ocurrió passback en alguno de los accesos, muestra avisos emitidos por los directivos y los medios autorizados. Además permite la actualización del lector, de la base de

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

datos local y la configuración del sistema. Este sistema con respecto al existente contiene mayor número de procesos y brinda nuevos servicios.

2.10 Análisis de la Factibilidad

La planificación basada en casos de usos es un método de estimación del tiempo de desarrollo de un proyecto mediante la asignación de "pesos" a un cierto número de factores que lo afectan, para finalmente, contabilizar el tiempo total estimado para el proyecto a partir de esos factores. A continuación, se detallan los pasos a seguir para la aplicación de éste método.

Se calcula a partir de la siguiente ecuación:

$$UUCP=UAW+UUCW$$

Donde:

UUCP: Puntos de casos de uso sin ajustar.

UAW: Factor de peso de los actores sin ajustar.

UUCW: Factor de peso de los casos de uso sin ajustar.

Este valor se calcula mediante un análisis de la cantidad de Actores presentes en el sistema y la complejidad de cada uno de ellos como se muestra a continuación.

Tipo de Actor	Descripción	Factor de Peso	Actores	Total
Simple	Otro sistema que interactúa con el sistema a desarrollar mediante una interfaz de programación.	1	0	0
Medio	Otro sistema que interactúa con el sistema a desarrollar mediante un protocolo o una interfaz basada en texto.	2	0	0
Complejo	Una persona que interactúa con el sistema mediante una interfaz gráfica.	3	1	3

Tabla17 Factor de peso de los actores sin ajustar.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

$$UAW = \sum \text{cant actores} * \text{peso}$$

$$UAW = 3$$

El valor del Factor de Peso de los Casos de Uso sin ajustar se calcula mediante un análisis de la cantidad de Casos de Uso presentes en el sistema y la complejidad de cada uno de ellos. La complejidad de los Casos de Uso se establece teniendo en cuenta la cantidad de transacciones efectuadas en el mismo, donde una transacción es una secuencia de actividades completa. Los criterios se muestran en la siguiente tabla:

Tipo de CU	Descripción	Factor de Peso	Cantidad de CU	Total
Simple	El caso de uso tiene de 1 a 3 transacciones.	5	7	35
Medio	El caso de uso tiene de 4 a 7 transacciones.	10	2	20
Complejo	El caso de uso tiene más de 8 transacciones.	15	1	15
Total			10	70

Tabla18 Factor de peso de los casos de uso sin ajustar.

$$UUCW = \sum \text{cantCU} * \text{Peso}$$

$$UUCW = 70$$

$$UUCP = 3 + 70$$

$$UUCP = 73$$

Paso 2. Cálculo de los Puntos de casos de uso ajustados.

El factor de complejidad técnica (TCF) se calcula mediante la cuantificación de un conjunto de factores que determinan la complejidad técnica del sistema. Cada factor se cuantifica en un valor desde 0 (aporte irrelevante) hasta 5 (aporte muy relevante).

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

Factor	Descripción	Peso	Valor asignado	Total	Comentario
T1	Sistema distribuido	2	5	10	El sistema será utilizado desde muchas máquinas a la vez.
T2	Tiempo de respuesta	1	3	3	La velocidad es limitada por la conectividad.
T3	Eficiencia del usuario final	1	3	3	Requiere de conocimiento para tratar con la tecnología y el sistema.
T4	Funcionamiento Interno complejo	1	0	0	No existen cálculos complejos.
T5	El código debe ser reutilizable	1	4	4	El código será reutilizable.
T6	Facilidad de instalación	0.5	4	2	No conlleva a una instalación compleja.
T7	Facilidad de uso	0.5	5	2.5	Gran facilidad de uso.
T8	Portabilidad	2	3	6	No se requiere que el sistema sea portable.
T9	Facilidad de cambio	1	2	2	Costo moderado de mantenimiento.
T10	Concurrencia	1	3	3	Existe concurrencia.
T11	Incluye objetivos especiales de seguridad	1	4	4	Sistema de autenticación por dominio.
T12	Provee acceso directo a terceras partes.	1	0	0	No presenta terceras partes.
T13	Se requieren facilidades especiales de entrenamiento de usuarios	1	4	4	Presenta facilidad de uso pero por su importancia se adiestrarán.
Total				43.5	

Tabla19 Factor de complejidad técnica.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

$$TCF = 0.6 + 0.01 * \Sigma(\text{peso} * \text{valor asignado})$$

$$TCF = 0.6 + 0.01 * 43.5$$

$$TCF = 0.6 + 0.435$$

$$TCF = 1.035$$

El factor de ambiente (EF) está relacionado con las habilidades y entrenamiento del grupo de desarrollo que realiza el sistema. Cada factor se cuantifica con un valor desde 0 (aporte irrelevante) hasta 5 (aporte muy relevante).

Factor	Descripción	Peso	Valor Asignado	Total	Comentario
E1	Familiaridad con el modelo de proyecto utilizado.	1.5	4	6	El grupo está familiarizado con el modelo.
E2	Experiencia en la aplicación	0.5	5	2.5	La mayoría del grupo ha trabajado tiempo en esta aplicación.
E3	Experiencia en la orientación a objetivos.	1	4	4	La mayoría del grupo programa en objetos.
E4	Capacidad del analista líder.	0.5	5	2.5	El líder es un Ingeniero en Informática.
E5	Motivación.	1	5	5	Existe motivación del grupo.
E6	Estabilidad de requerimientos	2	3	6	Existe probabilidad de cambios.
E7	Personal Part–Time	-1	1	-1	Todo el grupo trabaja a tiempo completo.
E8	Dificultad del lenguaje de programación	-1	2	-2	Se usará lenguaje Java.
Total				23	

Tabla20 Factor de ambiente.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

$$EF = 1.4 - 0.03 * \Sigma(\text{peso} * \text{valor asignado})$$

$$EF = 1.4 - 0,03 * 23$$

$$EF = 1.4 - 0.69$$

$$EF = 0.71$$

$$UCP = UUCP * TCF * EF$$

$$UCP = 73 * 1.035 * 0.71$$

$$UCP = 53.64405$$

Paso 3. Estimación de esfuerzo a través de los puntos de casos de uso.

$$E = UCP * CF$$

Donde:

E: Esfuerzo estimado en horas hombres.

UCP: Punto de casos de usos ajustados.

CF: Factor de conversión.

Para obtener el factor de conversión (CF) se cuentan cuántos valores de los que afectan el factor ambiente (E1...E6) están por debajo de la media (3), y los que están por arriba de la media para los restantes (E7, E8).

- Si el total es 2 o menos se utiliza el factor de conversión 20 Horas-Hombre / Punto de Casos de uso.
- Si el total es 3 ó 4 se utiliza el factor de conversión 28 Horas-Hombre / Punto de Casos de uso.
- Si el total es mayor o igual que 5 se recomienda efectuar cambios en el proyecto ya que se considera que el riesgo de fracaso del mismo es demasiado alto.

En este caso se puede decir que:

CF = 20 Horas-Hombre / Punto de Casos de uso.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

$$E = 53.64405 * 20$$

$$E = 1072.881$$

Paso 4. Calcular esfuerzo de todo el proyecto.

Actividad	Porcentaje %	Horas-Hombres
Análisis	10	268.22025
Diseño	20	536.4405
Implementación	40	1072.881
Pruebas	15	402.330375
Sobrecarga (otras actividades)	15	402.330375
Total	100	2682.2025

Tabla21 Esfuerzo del proyecto.

Como el desarrollo de la investigación comprende las fases de análisis y diseño, el esfuerzo en estas etapas sería **ET = 804.66075 horas-hombre** y se estima que cada mes tiene 4 semanas y cada semana 5 días laborales, por tanto se trabajaría 20 días al mes y si se trabaja 6 horas diarias como promedio se trabajarían en un mes 120 horas laborables, eso daría un **ET = 6.71 mes-hombre.**

$$\text{Si: Tiempo} = ET / CH$$

$$\text{Tiempo} = 6.71 / 2$$

$$\text{Tiempo} = 3.36$$

Esto quiere decir que con 2 hombres trabajando en el proyecto el mismo se desarrolla en aproximadamente 3 meses.

Costo del Proyecto.

Se asume como salario promedio mensual \$100.00

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

CH: Cantidad de hombres.

Tiempo: Tiempo total del proyecto.

CH = 2 hombres

CHM = 2 * Salario Promedio

CHM = 200.00 \$/mes

Costo = CHM * ET / CH

Costo = 200.00 * 6.71 / 2

Costo = \$671

De los resultados obtenidos se interpreta que con 2 hombres trabajando en el proyecto, el mismo se desarrolla en aproximadamente 3 meses y su costo total se estima que sea \$671.

2.11 Conclusiones

En este capítulo, después de efectuarse un estudio sobre la situación actual del control de acceso en la Universidad se realizó la modelación del negocio, definiéndose actores, trabajadores y procesos. Se hizo una investigación para determinar los requerimientos funcionales y no funcionales con los que debe contar el sistema. También se hace referencia en este capítulo a la realización de los casos de uso, y se realizó una estimación del costo y la factibilidad del proyecto.

Capítulo 3: Análisis y Diseño del sistema

3.1 Introducción

En este capítulo se muestra el análisis y diseño del sistema, se representan los diagramas de clases de análisis y diseño pertenecientes a cada caso de uso. Además se abordan conceptos básicos para una mejor explicación del tema.

3.2 Modelo de Análisis

En el modelo de análisis se identifican las clases que describen la realización de los casos de uso, los atributos y las relaciones entre ellas. Con esta información se construye el Diagrama de clases del Análisis. Las clases de análisis se centran en los requerimientos funcionales y se clasifican en Interfaz, de Control o Entidad.

- ✓ Clase Interfaz: Modelan la interacción entre el sistema y sus actores.
- ✓ Clase Control: Representan coordinación, secuencia, transacciones, y control de otros objetos y a menudo encapsula a un caso de uso en concreto.
- ✓ Clase Entidad: Estas clases modelan información que posee una larga vida y que a menudo es persistente y fenómenos, conceptos y sucesos que ocurren en el mundo real.

3.2.1 Diagramas de Clases del Análisis

Diagrama de Clases del Análisis Caso de Uso: “Controlar acceso puntual de personas de la UCI”.

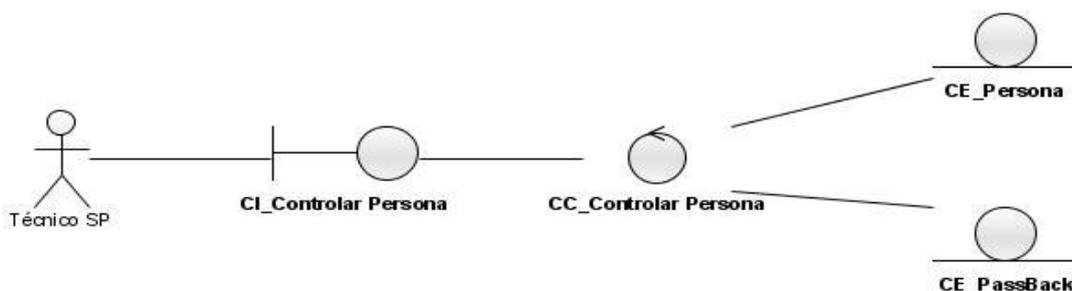


Figura5 DCA “Controlar acceso puntual de personas de la UCI”.

Diagrama de Clases del Análisis Caso de Uso: “Controlar acceso de vehículos”.

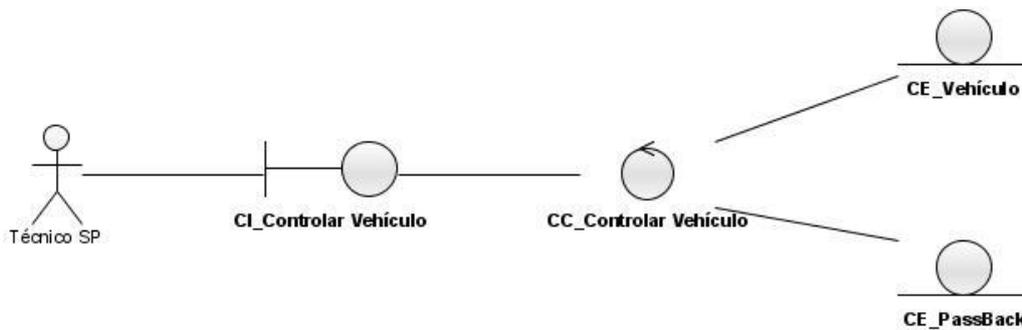


Figura6 DCA “Controlar acceso de vehículos”.

Diagrama de Clases del Análisis Caso de Uso: “Controlar acceso de visitantes”.

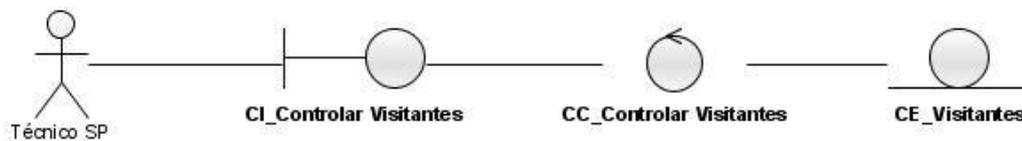


Figura7 DCA “Controlar acceso de visitantes”.

Diagrama de Clases del Análisis Caso de Uso: “Configurar sistema”.



Figura8 DCA “Configurar Sistema”.

Diagrama de Clases del Análisis Caso de Uso: “Actualizar lector”.

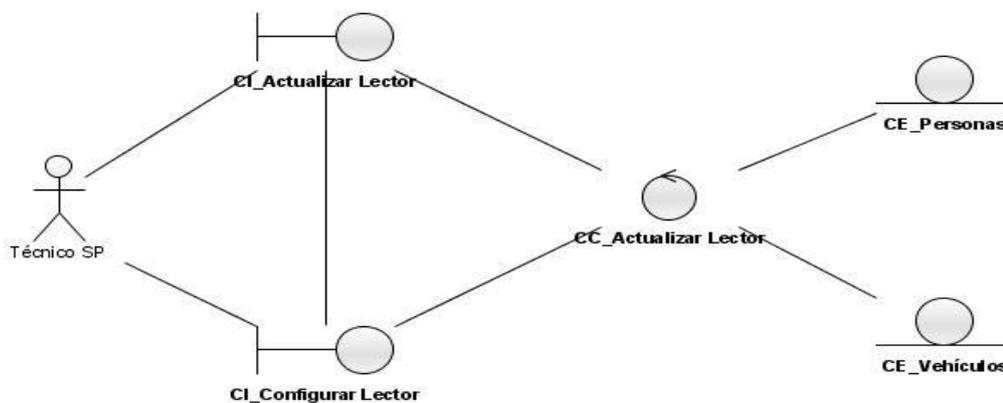


Figura9 DCA “Actualizar lector”.

Diagrama de Clases del Análisis Caso de Uso: “Actualizar BD local”.

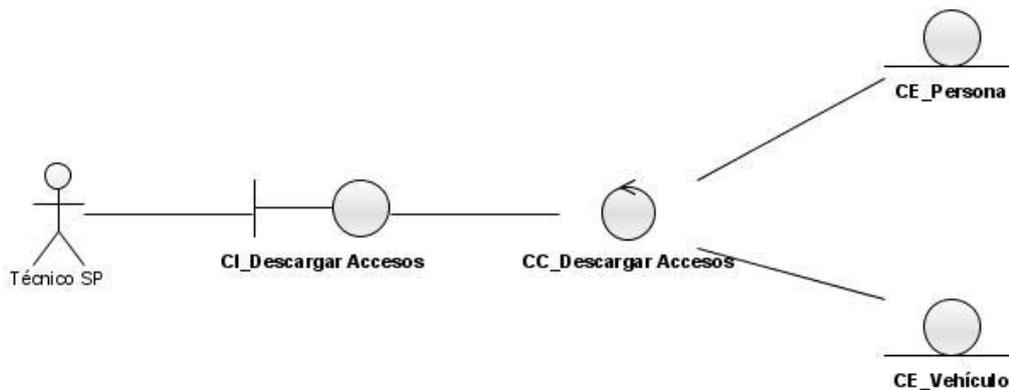


Figura10 DCA “Actualizar BD local”.

3.3 Modelo de Diseño

El modelo de diseño describe la realización física de los casos de uso centrándose en como los requisitos funcionales y no funcionales, junto a otras restricciones relacionadas con el entorno de implementación, tienen impacto en el sistema a considerar.

3.3.1 Diagrama de clases de diseño

Una clase de diseño es una construcción similar en la implementación del sistema. Los diagramas de clases de diseño exponen un conjunto de interfaces, colaboraciones y sus relaciones. Se utilizan para modelar la vista de diseño estática de un sistema. Los diagramas de clases de diseño del sistema están separados en casos de usos, es decir, cada caso de uso cuenta con un diagrama de clases, para hacer más fácil la comprensión.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Diagrama de Clases del Diseño Caso de Uso “Controlar Acceso puntual de personas de la UCI”.

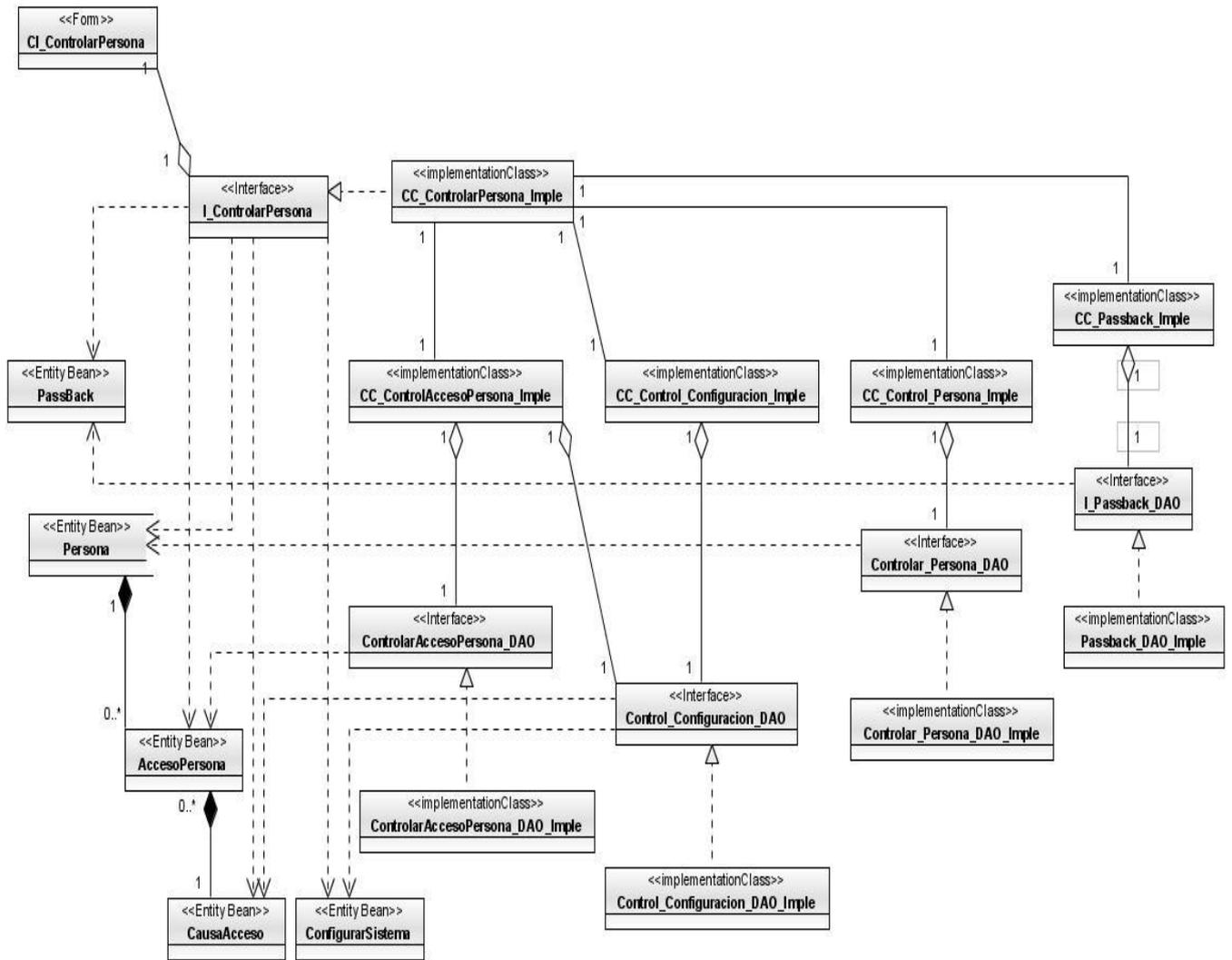


Figura11 DCD: “Controlar Acceso puntual de personas de la UCI”.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Diagrama de Clases del Diseño Caso de Uso “Controlar Acceso de Vehículos”.

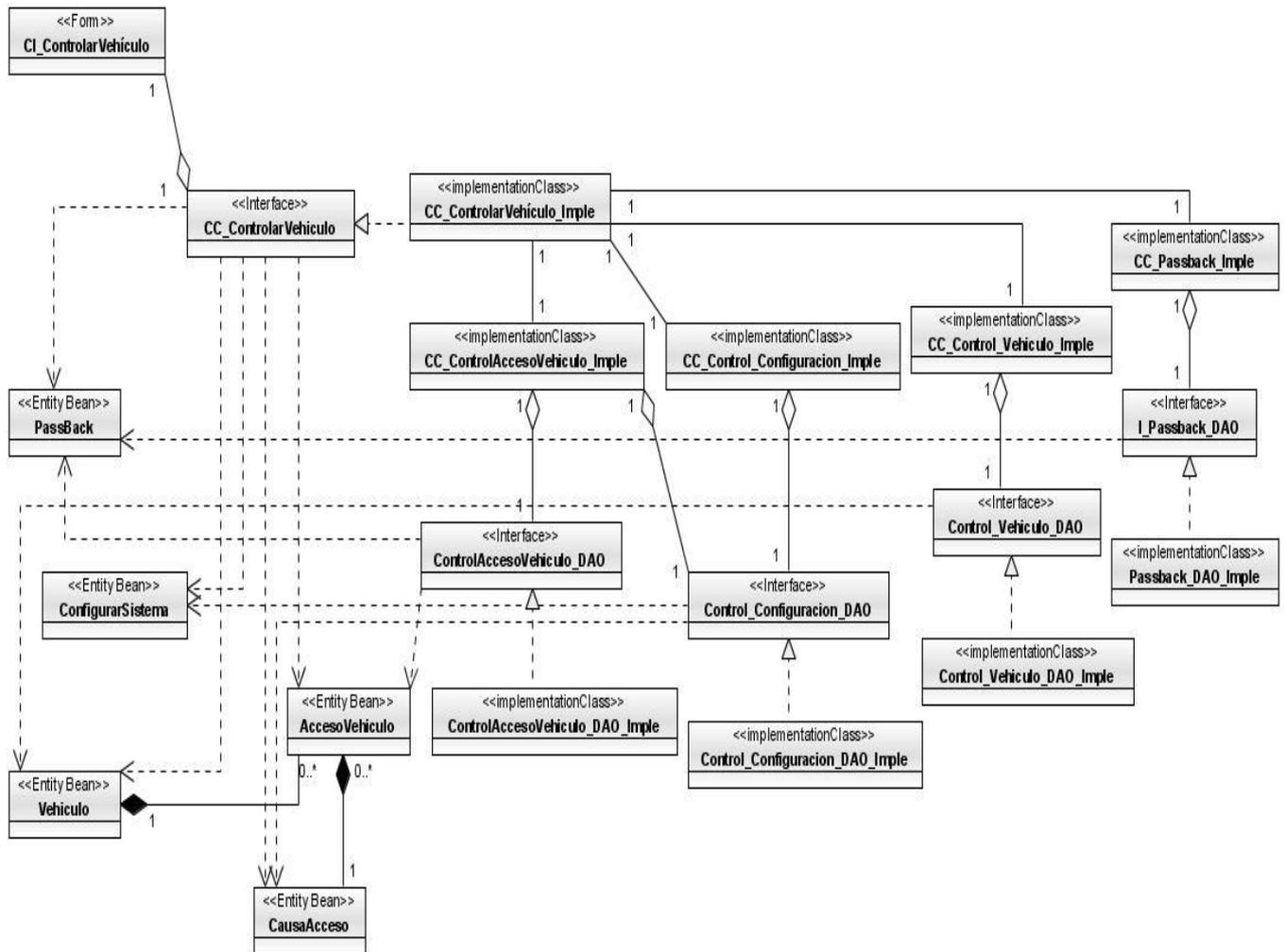


Figura 12 DCD: “Controlar Acceso de Vehículos”.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Diagrama de Clases del Diseño Caso de Uso “Controlar Acceso de Visitantes”.

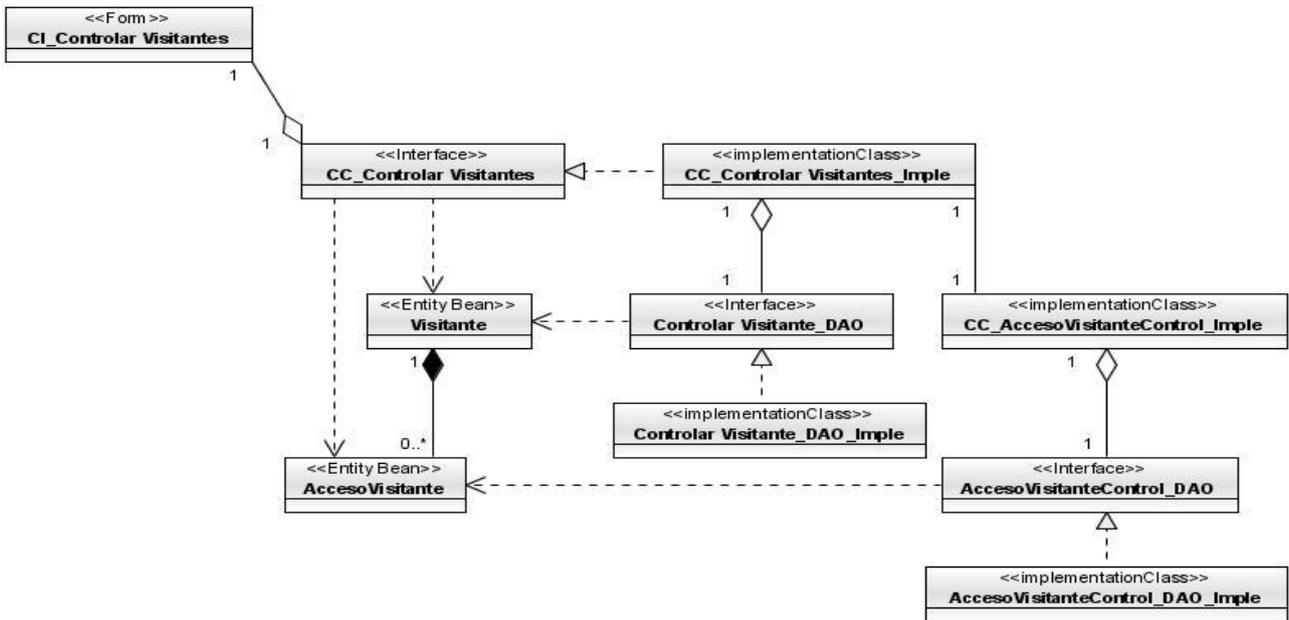


Figura 13 DCD: “Controlar Acceso de Visitantes”.

Diagrama de Clases del Diseño Caso de Uso “Configurar Sistema”.

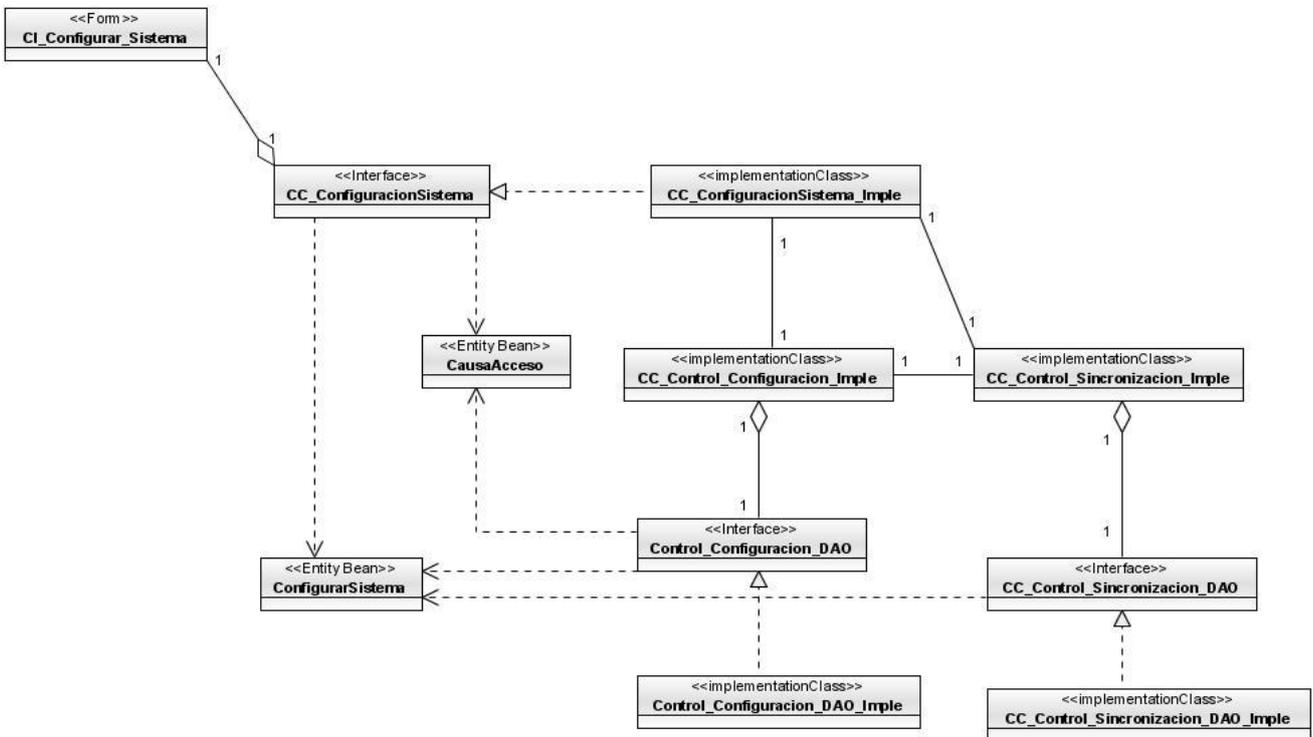


Figura 14 DCD: “Configurar Sistema”.

Diagrama de Clases del Diseño Caso de Uso “Actualizar Lector”.

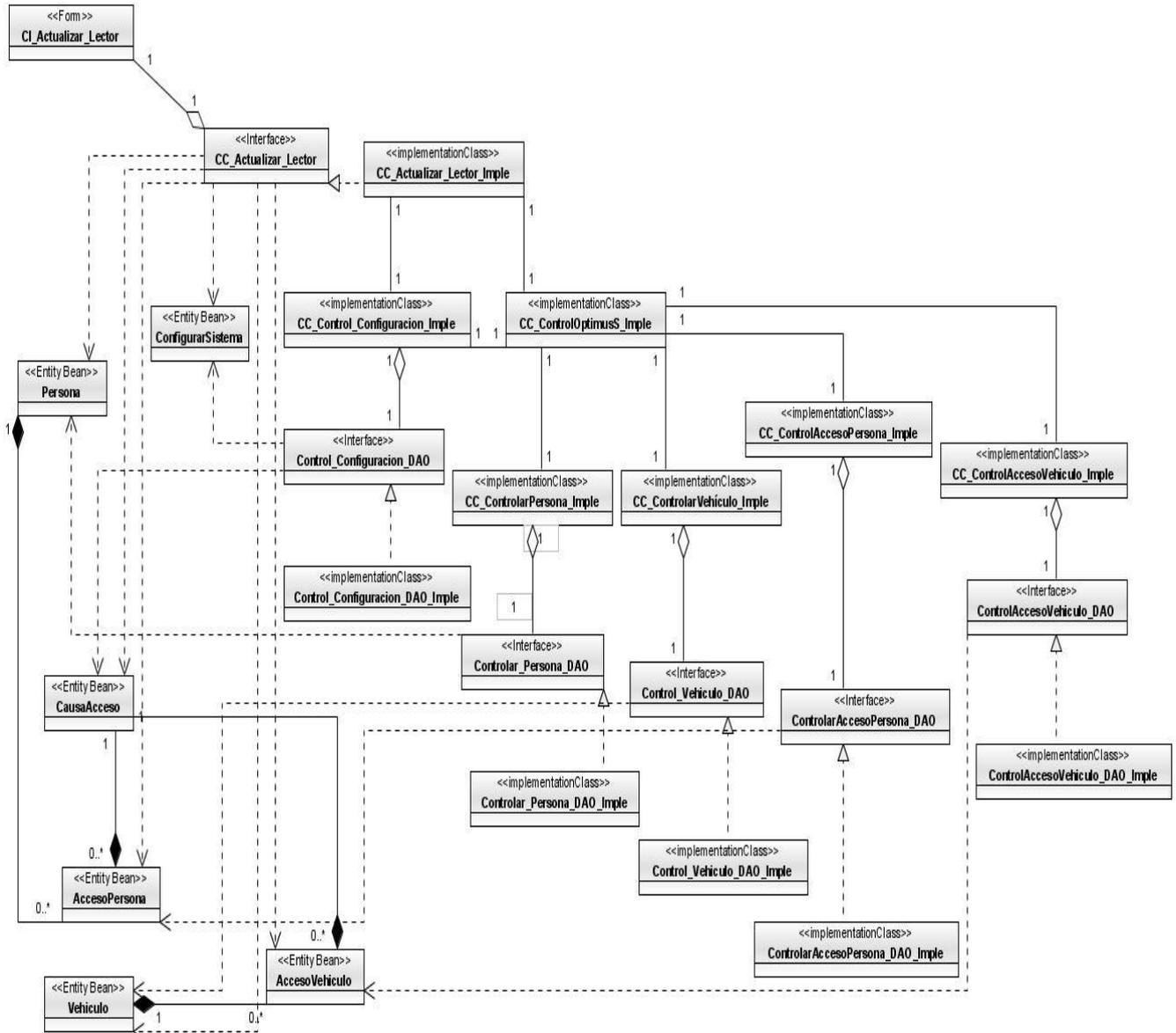


Figura 16 DCD: “Actualizar Lector”.

3.3.1.1 Diagramas de Secuencia

Un diagrama de secuencia es un diagrama de interacción que destaca la ordenación temporal de los mensajes. Debido al gran tamaño que adoptan los diagramas de secuencia, no es posible ejemplificarlos en este documento, ya que se verá afectada la legibilidad de los mismos.

3.3.2 Patrones de Diseño

Patrones GRASP

Los patrones GRASP describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones. GRASP es un acrónimo que significa General Responsibility Assignment Software Patterns (patrones generales de software para asignar responsabilidades). Estos fueron los patrones que se aplicaron para realizar el diseño de la aplicación que propone la investigación.

- **Experto:** Es un patrón que se usa más que cualquier otro al asignar responsabilidades; es un principio básico que suele utilizarse en el diseño orientado a objetos. Con él no se pretende designar una idea oscura ni extraña; expresa simplemente la "intuición" de que los objetos hacen cosas relacionadas con la información que poseen.
- **Creador:** El patrón Creador guía la asignación de responsabilidades relacionadas con la creación de objetos, tarea muy frecuente en los sistemas orientados a objetos. El propósito fundamental de este patrón es encontrar un creador que se debe conectar con el objeto producido en cualquier evento. Al escogerlo como creador, se da soporte al bajo acoplamiento.
- **Alta Cohesión:** Asignar una responsabilidad de modo que la cohesión siga siendo alta. Una clase con baja cohesión hace muchas cosas no afines o un trabajo excesivo. Una clase con mucha cohesión es útil porque es bastante fácil darle mantenimiento, entenderla y reutilizarla. La ventaja que significa una gran funcionalidad también soporta un aumento de la capacidad de reutilización.
- **Bajo Acoplamiento:** Asignar una responsabilidad para mantener bajo acoplamiento (una clase con bajo (o débil) acoplamiento no depende de muchas otras; una clase con alto (o fuerte) acoplamiento recurre a muchas otras.). Este patrón estimula asignar una responsabilidad de modo que su colocación no incremente el acoplamiento tanto que produzca los resultados negativos propios de un alto acoplamiento.

- **Controlador:** Asignar la responsabilidad del manejo de un mensaje de los eventos de un sistema a una clase. La mayor parte de los sistemas reciben eventos de entrada externa, En cualquiera de los casos que puedan existir, si se recurre a un diseño orientado a objetos, hay que elegir los controladores que manejen esos eventos de entrada.

3.4 Descripción de la arquitectura

El modelado del diseño se basa en una arquitectura de tres niveles o capas, donde cada nivel hace referencia a los componentes de las capas inferiores. La capa de presentación reúne las interfaces de usuario. La capa de negocio recoge la lógica de la aplicación y constituye básicamente el código al que recurre la capa de presentación para recuperar los datos deseados. La tercera capa, denominada capa de persistencia, contiene las fuentes de información necesarias para el funcionamiento de la aplicación. Para esta arquitectura se utiliza el patrón de diseño DAO (Objeto de acceso a datos), que a continuación se describe.

Patrón DAO (Objeto de acceso a datos): Es un componente de software que suministra una interfaz común entre la aplicación y uno o más dispositivos de almacenamiento de datos, tales como una Base de datos o un archivo.

3.5 Descripción de Clases

3.5.1 Clases Interfaz

Clase Interfaz: "CI_Controlar Persona".

Nombre: CI_Controlar Persona	
Tipo de clase: interfaz	
Atributo	Tipo
cbox_Autoregistrar	jcheckbox
rbtn_Entrada	jradiobutton
rbtn_Salida	jradiobutton
tbox_Solapín	jtextbox

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

tbox_Ci	jtextbox
btn_Entrada	jbutton
btn_Salida	jbutton
btn_Deshacer	jbutton
btn_Buscar	jbutton
Para cada responsabilidad:	
Nombre:	public void jButtonEntrada_Click
Descripción:	Guarda datos del acceso.
Nombre:	public void jButtonSalida_Click
Descripción:	Guarda datos del acceso.
Nombre:	public void jButtonDeshacer_Click
Descripción:	Elimina el último acceso.
Nombre:	public void jButtonBuscar_Click
Descripción:	Busca la persona por Ci o solapín.
Nombre:	public ValidarDatos
Descripción:	Valida los datos (campos vacíos).
Nombre:	public MostrarImagen
Descripción:	Muestra la foto de la persona perteneciente a la UCI.
Nombre:	public MostrarPersona
Descripción:	Muestra los datos de la persona perteneciente a la UCI.

Tabla22 Clase Interfaz “CI_Controlar Persona”.

Clase Interfaz: “CI_Controlar Vehículo”.

Nombre: CI_Controlar Vehículo
Tipo de clase: interfaz

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Atributo	Tipo
tbox_Chapa	jtextbox
btn_Buscar	jbutton
btn_Entrada	jbutton
btn_Salida	jbutton
btn_Deshacer	jbutton
Para cada responsabilidad:	
Nombre:	public void jButtonBuscar_Click
Descripción:	Busca el vehículo por chapa.
Nombre:	public void jButtonEntrada_Click
Descripción:	Guarda datos del acceso.
Nombre:	public void jButtonSalida_Click
Descripción:	Guarda datos del acceso.
Nombre:	public void jButtonDeshacer_Click
Descripción:	Elimina el último acceso.
Nombre:	public ValidarDatos
Descripción:	Valida los datos (campos vacíos).
Nombre:	public MostrarImagenChofer
Descripción:	Muestra la foto del chofer del vehículo
Nombre:	public MostrarImagenVehiculo
Descripción:	Muestra la foto del vehículo.
Nombre:	public MostrarVehiculo
Descripción:	Muestra los datos del vehículo.

Tabla23 Clase Interfaz “CI_Controlar Vehículo”.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Clase Interfaz: “CI_Controlar Visitantes”.

Nombre: CI_Controlar Visitantes	
Tipo de clase: interfaz	
Atributo	Tipo
cbox_Entrar_Visitante	jcheckbox
tbox_Nom_Ape	jtextbox
tbox_Ci	jtextbox
tbox_Organismo	jtextbox
tbox_Lugar_Visitar	jtextbox
tbox_Dirig_Autoriza	jtextbox
combox_Paquetes	jcombobox
tbox_Fecha_Entrada	jtextbox
tbox_Fecha_Salida	jtextbox
tbox_Nom_Chofer	jtextbox
tbox_Ci_Chofer	jtextbox
tbox_Lic_Cond	jtextbox
tbox_Circulacion_Veh	jtextbox
tbox_Marca	jtextbox
tbox_Chapa	jtextbox
tbox_Tipo_Veh	jtextbox
combox_Autorizado	jcombobox
btn_Aceptar	jbutton
btn_Cancelar	jbutton
Para cada responsabilidad:	
Nombre:	public void jButtonBuscar_Click

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Descripción:	Busca datos del visitante.
Nombre:	public void jButtonEntrada_Click
Descripción:	Guarda datos del acceso.
Nombre:	public void jButtonSalida_Click
Descripción:	Guarda datos del acceso.
Nombre:	public void jButtonDeshacer_Click
Descripción:	Elimina el último acceso.
Nombre:	public ValidarDatos
Descripción:	Valida los datos (campos vacíos).
Nombre:	public MostrarVisitante
Descripción:	Muestra los datos del visitante

Tabla24 Clase Interfaz “CI_Controlar Visitantes”.

Clase Interfaz: “CI_Configurar Sistema”.

Nombre: CI_Configurar Sistema	
Tipo de clase: interfaz	
Atributo	Tipo
tbox_Subir_Acc_Persona	jtextbox
tbox_Subir_Acc_Vehiculo	jtextbox
tbox_Subir_Acc_Visitante	jtextbox
tbox_Bajar_List_Persona	jtextbox
tbox_Bajar_List_Vehiculo	jtextbox
tbox_Bajar_List_Visitante	jtextbox
btn_Sinc_Acceso_Persona	jbutton
btn_Sinc_Acceso_Vehiculo	jbutton

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

btn_Sinc_Acceso_Visitante	jbutton
btn_Sinc_list_Persona	jbutton
btn_Sinc_list_Vehiculo	jbutton
btn_Sinc_list_Visitante	jbutton
combox_PuertasNoActivas	jcombobox
rbtn_Si_Sinc_Persona	jradiobutton
rbtn_No_Sinc_Persona	jradiobutton
rbtn_Si_Sinc_Vehiculo	jradiobutton
rbtn_No_Sinc_Vehiculo	jradiobutton
rbtn_Si_Sinc_Visitante	jradiobutton
rbtn_No_Sinc_Visitante	jradiobutton
btn_Aceptar	jbutton
btn_Cancelar	jbutton
Para cada responsabilidad:	
Nombre:	public void jbutton Sinc_Acceso_Persona_Click
Descripción:	Subir accesos de personas a la BD remota.
Nombre:	public void jbutton Sinc_Acceso_Vehiculo_Click
Descripción:	Subir accesos de vehiculos a la BD remota.
Nombre:	public void jbutton Sinc_Acceso_Visitante_Click
Descripción:	Subir accesos de visitantes a la BD remota.
Nombre:	public void jbutton Sinc_list_Persona_Click
Descripción:	Actualiza BD local.
Nombre:	protected void jbutton Sinc_list_Vehiculo_Click
Descripción:	Actualiza BD local.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Nombre:	protected void jButton Sinc_list_Visitantes_Click
Descripción:	Actualiza BD local.
Nombre:	public void jButtonAceptar_Click
Descripción:	Guarda en la BD local los datos introducidos.
Nombre:	public void jButtonCancelar_Click
Descripción:	Cierra la interfaz.
Nombre:	public ValidarDatos
Descripción:	Valida los datos (campos vacíos).
Nombre:	public MostrarDatos
Descripción:	Muestra los datos de los tiempos.

Tabla25 Clase Interfaz “CI_Configurar Sistema”.

Clase Interfaz: “CI_Actualizar Lector”.

Nombre: CI_Actualizar Lector	
Tipo de clase: interfaz	
Atributo	Tipo
cbox_Personas	jcheckbox
cbox_Vehiculos	jcheckbox
combox_Personas	jcombobox
combox_Vehiculos	jcombobox
btn_Actualizar_Opt	jbutton
btn_Cancelar	jbutton
Para cada responsabilidad:	
Nombre:	public void jButton Actualizar_Opt _Click
Descripción:	Guarda en el lector los códigos de barra.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Nombre:	public void jButton Cancelar _Click
Descripción:	Cierra la interfaz.
Nombre:	public ValidarSeleccion
Descripción:	Valida que al menos un tipo de credencial esté seleccionada.

Tabla26 Clase Interfaz “CI_Actualizar Lector”.

Clase Interfaz: “CI_Actualizar_Base_Datos_Local”.

Nombre: CI_Actualizar_Base_Datos_Local	
Tipo de clase: interfaz	
Atributo	Tipo
combox_ Causa_Acceso	jcombobox
cbox_Persona	jcheckbox
cbox_Vehiculo	jcheckbox
btn_ Entrada	jbutton
btn_ Salida	jbutton
Para cada responsabilidad:	
Nombre:	public void jButton Entradas_Click
Descripción:	Guarda los datos de los accesos en la BD local.
Nombre:	public void jButton Salidas_Click
Descripción:	Guarda los datos de los accesos en la BD local.
Nombre:	public ValidarVacíaSeleccion
Descripción:	Valida que al menos una causa de acceso esté seleccionada.
Nombre:	public void CargarCausaAcceso
Descripción:	Muestra todas las causas de acceso.

Tabla27 Clase Interfaz “CI_Actualizar_Base_Datos_Local”.

3.5.2 Clases Controladoras

Clase Controladora: “CC_ControlarPersona_Imple”.

Nombre: CC_ControlarPersona_Imple	
Tipo de clase: controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	BuscarPersona(Solapin : String, CodBarras : String, CI : String) : Persona
Descripción:	Devuelve la persona.
Nombre:	Auto_Registrar(CodBarra : String) : Persona
Descripción:	Auto registra la persona según la opción “Entrada” o “Salida”.
Nombre:	Verificar_PassBack(aGuid : String, TipoAcceso : String) : boolean
Descripción:	Verifica que se genere PassBack.
Nombre:	RegistrarPassback(passBack :PassBack) : void
Descripción:	Registra el motivo del PassBack.
Nombre:	SalvarAcceso(persona : Persona, Tipo Acceso : String) : void
Descripción:	Salva los accesos de las persona.
Nombre:	Deshacer(Guid : String, Ult_Fecha : Datetime) : boolean
Descripción:	Elimina el último acceso.
Nombre:	BuscarCredenciales(aTipoCredencial : String) : ArrayList
Descripción:	Busca todas las credenciales activas.
Nombre:	BuscarTiposCredencial() : ArrayList
Descripción:	Busca todos los tipos de credenciales existentes.

Tabla28 Clase Controladora “CC_ControlarPersona_Imple”.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Clase Controladora: "CC_ControlAccesoPersona_Imple".

Nombre: CC_ControlAccesoPersona_Imple	
Tipo de clase: controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	RegistrarAccesoPersona(persona : Persona, aTipo : String) : boolean
Descripción:	Registra la persona y el tipo de acceso.
Nombre:	RegistrarAccesoPersonaLocal(accesoPersona : AccesoPersona) : boolean
Descripción:	Registra el acceso de la persona.
Nombre:	EliminarAccesoPersonaLocal(aGuid : String, aFechaHora : Datetime) : boolean
Descripción:	Elimina el último acceso introducido en la BD local.

Tabla29 Clase Controladora "CC_ControlAccesoPersona_Imple".

Clase Controladora: "CC_ControlPersona_Imple".

Nombre: CC_ControlPersona_Imple	
Tipo de clase: controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	BuscarPersona(Solapin : String, CodBarras : String, CI : String) : Persona
Descripción:	Devuelve la persona.
Nombre:	BuscarCredenciales(TipoCredencial : String) : ArrayList
Descripción:	Busca todas las credenciales activas.
Nombre:	BuscarTiposCredenciales() : ArrayList

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Descripción:	Busca todos los tipos de credenciales existentes.
--------------	---

Tabla30 Clase Controladora “CC_ControlPersona_Imple”.

Clase Controladora: “CC_ControlarVehiculo_Imple”.

Nombre: CC_ControlarVehiculo_Imple	
Tipo de clase: controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	Buscar_x_Chapa(Chapa : String) :Vehiculo
Descripción:	Devuelve el vehículo con esa chapa.
Nombre:	Verificar_PassBack(aGuid : String, TipoAcceso : String) : boolean
Descripción:	Verifica que se genere PassBack.
Nombre:	RegistrarPassback(passBack : PassBack) : void
Descripción:	Registra el motivo del PassBack.
Nombre:	SalvarAcceso(vehiculo : Vehiculo, TipoAcceso : String) : void
Descripción:	Salva los accesos de los vehículos.
Nombre:	Deshacer(Guid : String, Ult_Fecha : Datetime) : boolean
Descripción:	Elimina el último acceso.
Nombre:	BuscarCredenciales(aTipoCredencial : String) : ArrayList
Descripción:	Busca todas las credenciales activas.
Nombre:	BuscarTiposCredencial() : ArrayList
Descripción:	Busca todos los tipos de credenciales existentes.

Tabla31 Clase Controladora “CC_ControlarVehiculo_Imple”.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Clase Controladora: "CC_ControlAccesoVehiculo_Imple".

Nombre: CC_ControlAccesoVehiculo_Imple	
Tipo de clase: controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	RegistrarAccesoVehiculo(vehiculo :Vehiculo, aTipo : String) : boolean
Descripción:	Registra la persona y el tipo de acceso.
Nombre:	RegistrarAccesoVehiculoLocal(accesoVehiculo : AccesoVehiculo) : boolean
Descripción:	Registra el acceso de la persona.
Nombre:	EliminarAccesoVehiculoLocal(aGuid : String, aFechaHora : Datetime) : boolean
Descripción:	Elimina el último acceso introducido en la BD local.

Tabla32 Clase Controladora "CC_ControlAccesoVehiculo_Imple".

Clase Controladora: "CC_ControlVehiculo_Imple".

Nombre: CC_ControlVehiculo_Imple	
Tipo de clase: controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	BuscarVehiculoXChapa(aChapa : String) : Vehiculo
Descripción:	Devuelve el vehículo.
Nombre:	BuscarCredencial(TipoCredencial : String) : ArrayList
Descripción:	Busca todas las credenciales activas.
Nombre:	BuscarTiposCredenciales() : ArrayList

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Descripción:	Busca todos los tipos de credenciales existentes.
--------------	---

Tabla33 Clase Controladora “CC_ ControlVehiculo_Imple”.

Clase Controladora: “CC_ ControlarVisitantes_Imple”.

Nombre: CC_ ControlarVisitantes_Imple	
Tipo de clase: controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	Buscar_x_Ci(Ci : String) : Visitante
Descripción:	Devuelve el visitante con ese CI.
Nombre:	SalvarAcceso(visitante : Acceso Visitantes.Visitante) : void
Descripción:	Salva los accesos de los visitantes.
Nombre:	Deshacer(CI : String, Ult_Fecha : Datetime) : boolean
Descripción:	Elimina el último acceso.
Nombre:	EntrarVisitante(Nombre : String, NoCI : String, Organismo : String, LugarVisitar : String, DirigenteAut : String, Paquetes : boolean, FechaE : String, FechaS : String, NomChefer : String, CiChofer : String, LicConduccion : String, Circulacio : String, Marca : String, Chapa : String, TipoVehiculo : String) : void
Descripción:	Guarda los datos introducidos en la BD.

Tabla34 Clase Controladora “CC_ ControlarVisitantes_Imple”.

Clase Controladora: “CC_ AccesoVisitantesControl_Imple”.

Nombre: CC_ AccesoVisitantesControl_Imple	
Tipo de clase: controladora	
Atributo	Tipo
Para cada responsabilidad:	

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Nombre:	RegistrarAccesoVisitante(visitante :Visitante) : boolean
Descripción:	Registra el acceso del visitante
Nombre:	EliminarAccesoVisitanteLocal(CI : String, AFechaHora : Datetime) : boolean
Descripción:	Elimina el último acceso.

Tabla35 Clase Controladora “CC_AccesoVisitantesControl_Imple”.

Clase Controladora: “CC_ConfigurarSistema”.

Nombre: CC_ConfigurarSistema	
Tipo de clase: controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	Subir_Acceso_Persona() : boolean
Descripción:	Sube los accesos de las personas de la BD local a la remota.
Nombre:	Subir_Acceso_Vehiculo() : boolean
Descripción:	Sube los accesos de los vehículos de la BD local a la remota.
Nombre:	Subir_Acceso_Visitante() : boolean
Descripción:	Sube los accesos de los visitantes de la BD local a la remota.
Nombre:	Bajar_Listado_Persona() : boolean
Descripción:	Baja los listados de personas de la BD remota a la local.
Nombre:	Bajar_Listado_Vehiculo() : boolean
Descripción:	Baja los listados de vehículos de la BD remota a la local.
Nombre:	Bajar_Listado_Visitante() : boolean
Descripción:	Baja los listados de visitantes de la BD remota a la local.
Nombre:	BuscarPuertasNoActivas() : ArrayList

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Descripción:	Busca las puertas que no están en uso.
Nombre:	Actualizar_Campos_Locales(P_Sinc_Persona : int, F_Sinc_Persona : Datetime, P_Sinc_Accesos_Persona : int, P_Sinc_Vehiculo : int, F_Sinc_Vehiculo : Datetime, P_Sinc_Acceso_Vehiculo : int, P_Sinc_Visitante : int, F_Sinc_Visitante : Datetime, P_Sinc_Acceso_Visitante : int) : void
Descripción:	Actualiza en la BD local los datos introducidos.
Nombre:	ObtenerCampo_Base_de_Dato_Local(aNombreCampo : String) : String
Descripción:	Obtiene de la BD el campo deseado.
Nombre:	ObtenerListadoDatos() : ArrayList
Descripción:	Obtiene un listado con los campos de la BD.
Nombre:	ActualizarAccesosPersonas() : boolean
Descripción:	Actualiza los accesos de personas en la BD remota.
Nombre:	ActualizarAccesosVehiculos() : boolean
Descripción:	Actualiza los accesos de vehículos en la BD remota.
Nombre:	ActualizarAccesosVisitantes() : boolean
Descripción:	Actualiza los accesos de visitantes en la BD remota.
Nombre:	ActualizarCampoLocal(aCampo : String) : boolean
Descripción	Actualiza un campo en la BD local.

Tabla36 Clase Controladora “CC_ ConfigurarSistema”.

Clase Controladora: “CC_ControlSincronizacion_Imple”.

Nombre: CC_ControlSincronizacion_Imple	
Tipo de clase: controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	ActualizarAccesosPersonas() : boolean

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Descripción:	Sube los accesos de las personas de la BD local a la remota.
Nombre:	ActualizarAccesosVehiculos() : boolean
Descripción:	Sube los accesos de los vehículos de la BD local a la remota.
Nombre:	ActualizarAccesosVisitante() : boolean
Descripción:	Sube los accesos de los visitantes de la BD local a la remota.
Nombre:	ActualizarPersonasLocales() : boolean
Descripción:	Baja el listado de personas de la BD remota a la local.
Nombre:	ActualizarVehiculosLocales() : boolean
Descripción:	Baja el listado de vehículos de la BD remota a la local.
Nombre:	ActualizarVisitantesLocales() : boolean
Descripción:	Baja el listado de visitantes de la BD remota a la local.

Tabla37 Clase Controladora “CC_ ControlSincronizacion_Imple”.

Clase Controladora: “CC_ActualizarLector_Imple”.

Nombre: CC_ActualizarLector_Imple	
Tipo de clase: controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	BuscarCredencialesPersona(Tipo : String) : ArrayList
Descripción:	Devuelve un listado de las credenciales de las personas.
Nombre:	BuscarTiposCredencialesPersona() : ArrayList
Descripción:	Devuelve un listado con los tipos de credenciales de personas.
Nombre:	BuscarCredencialesVehiculo(TipoCredencial : String) : ArrayList
Descripción:	Devuelve un listado de las credenciales de los vehículos.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Nombre:	BuscarTiposCredencialesVehiculos() : ArrayList
Descripción:	Devuelve un listado de los tipos de credenciales de vehículos.
Nombre:	SubirDatosOptimus() : void
Descripción:	Actualiza al Optimus con las credenciales de personas o vehículos.

Tabla38 Clase Controladora “CC_ActualizarLector_Imple”.

Clase Controladora: “CC_ControlOptimusS_Imple”.

Nombre: CC_ControlOptimusS_Imple	
Tipo de clase: controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	Ejecutar(aCamino : String) : void
Descripción:	Actualiza las credenciales del lector
Nombre:	DescargarAccesos() : void
Descripción:	Baja del lector hacia la BD local los accesos.
Nombre:	SubirLookup() : void
Descripción:	Envia al lector todos las credenciales.
Nombre:	GenerarLecturas(TipoAcceso : String, Causa : String) : ArrayList
Descripción:	Devuelve un listado de todos los accesos registrados en el lector.

Tabla 39 Clase Controladora “CC_ControlOptimusS_Imple”.

Clase Controladora: “CC_Actualizar_Base_Datos_Local_Imple”.

Nombre: CC_Actualizar_Base_Datos_Local_Imple	
Tipo de clase: controladora	
Atributo	Tipo

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Para cada responsabilidad:	
Nombre:	DescargarAccesos() : void
Descripción:	Baja del lector hacia la BD local los accesos.
Nombre:	GenerarLecturas(TipoAcceso : String, Causa : String) : boolean
Descripción:	Devuelve un listado de todos los accesos registrados en el lector.
Nombre:	RegistrarCausaAcceso(causaAcceso : String) : boolean
Descripción:	Registra en la BD local la causa de acceso.
Nombre:	ObtenerCausaAceso() : ArrayList
Descripción:	Devuelve un listado con las causas de acceso.
Nombre:	CargarCausaAcceso(CausasAccesos : ArrayList) : void
Descripción:	Muestra un listado con las causas de acceso.

Tabla40 Clase Controladora “CC_Actualizar_Base_Datos_Local_Imple”.

Clase Controladora: “CC_PassBack_Imple”.

Nombre: CC_PassBack_Imple	
Tipo de clase: controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	Verificar_PassBack(aGuid : String, Tipo : TipoAcceso) : boolean
Descripción:	Verifica que exista Passback.
Nombre:	RegistrarPassback(Passback :PassBack) : void
Descripción:	Guarda en la BD el motivo del PassBack.

Tabla41 Clase Controladora “CC_PassBack_Imple”.

3.5.3 Clases entidad

Clase entidad: "Persona".

Nombre: Persona	
Tipo de clase: entidad	
Atributo	Tipo
PersonasNombreCompleto	String
CodigoBarras	String
Solapin	String
NombreArea	String
NombreFoto	String
TipoCredencial	String
EstadoCredencial	String
Guid	String
FechaExpiración	Datetime

Tabla42 Clase Entidad "Persona".

Clase entidad: "Vehículo".

Nombre: Vehiculo	
Tipo de clase: entidad	
Atributo	Tipo
Chapa	String
Color	String
Marca	String
Tipo_Vehiculo	int

Tabla43 Clase Entidad "Vehículo".

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Clase entidad: "Visitante".

Nombre: Visitante	
Tipo de clase: entidad	
Atributo	Tipo
NombreApellidos	String
NoCI	String
Organismo	String
LugarVisitar	String
DirigenteQAutoriza	String
Paquetes	boolean
FechaEntrada	String
FechaSalida	String
NombreChofer	String
CIChofer	String
LicenciaConducción	String
CirculaciónCarro	String
Marca	String
Chapa	String
TipoVehículo	String

Tabla44 Clase Entidad "Visitante".

Clase entidad: "ConfigurarSistema".

Nombre: ConfigurarSistema	
Tipo de clase: entidad	
Atributo	Tipo

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

P_Sinc_Persona	int
P_Sinc_Vehiculo	int
P_Sinc_Visitante	int
P_Sinc_Accesos_Personas	int
P_Sinc_Accesos_Vehiculos	int
P_Sinc_Accesos_Visitantes	int
UF_Sinc_Accesos_Personas	Datetime
UF_Sinc_Accesos_Vehiculos	Datetime
UF_Sinc_Accesos_Visitantes	Datetime
F_Sinc_Persona	Datetime
F_Sinc_Vehiculo	Datetime
F_Sinc_Visitante	Datetime
UF_Sinc_Personas	Datetime
UF_Sinc_Vehiculos	Datetime
UF_Sinc_Visitantes	Datetime
UF_Acceso_Persona	Datetime
UF_Acceso_Vehiculo	Datetime
UF_Acceso_Visitante	Datetime
SN_Sinc_Personas	String
SN_Sinc_Vehiculos	String
SN_Sinc_Visitante	String
U_Guid_Personas	String
U_Guid_Vehiculos	String
Ult_CI : String	String

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

UF_Inicio_APP	Datetime
ID_Puerta	String
U_Apagado	String

Tabla45 Clase Entidad “ConfigurarSistema”.

Clase entidad: “PassBack”.

Nombre: PassBack	
Tipo de clase: entidad	
Atributo	Tipo
Comentario	String
Permitido	String
IdAcceso	String

Tabla46 Clase Entidad “PassBack”.

Clase entidad: “AccesoPersona”.

Nombre: AccesoPersona	
Tipo de clase: entidad	
Atributo	Tipo
Id_Acceso	int
GUID	String
FechaHora	String
Id_Tipo	int
Id_Causa	int

Tabla47 Clase Entidad “AccesoPersona”.

Clase entidad: “AccesoVehículo”.

Nombre: AccesoVehiculo

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Tipo de clase: entidad	
Atributo	Tipo
Id_Acceso	int
Chapa	String
FechaHora	String
Tipo	String
GuidChofer	String

Tabla48 Clase Entidad “Acceso Vehiculo”.

Clase entidad: “AccesoVisitante”.

Nombre: AccesoVisitante	
Tipo de clase: entidad	
Atributo	Tipo
Id_Acceso	int
FechaHora	String
NoCI	String
Id_Tipo	int
Id_Causa	int

Tabla49 Clase Entidad “AccesoVisitante”.

Clase entidad: “CausaAcceso”.

Nombre: CausaAcceso	
Tipo de clase: entidad	
Atributo	Tipo
Id_Causa	int
Causa	String

Tabla50 Clase Entidad “CausaAcceso”.

3.5.4 Clases de Acceso a datos

Clase de acceso a datos: “ControlarAccesoPersona_DAO_Imple”.

Nombre: ControlarAccesoPersona_DAO_Imple	
Tipo de clase: acceso a datos	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	EliminarAccesoPersonaLocal(aGuid : String, aFechaHora : Datetime) : boolean
Descripción:	Elimina el último acceso introducido en la BD local.
Nombre:	ObtenerAccesosPersonas() : ArrayList
Descripción:	Devuelve un listado con todos los accesos de la BD local.
Nombre:	EliminarAccesosPersonas() : void
Descripción:	Elimina todos los accesos de la BD local.
Nombre:	RegistrarAccesoPersona(ListaAcceso : ArrayList, Nombre_Cliente : String) : boolean
Descripción:	Registra todos los accesos de las personas desde la BD local hacia la remota.
Nombre:	RegistrarAccesoPersonaLocal(accesosPersona : AccesoPersona) : boolean
Descripción:	Se registran los accesos de las personas en la BD local.

Tabla 51 Clase de Acceso a Datos “ControlarAccesoPersona_DAO_Imple”.

Clase de acceso a datos: “ControlarPersona_DAO_Imple”.

Nombre: ControlarPersona_DAO_Imple	
Tipo de clase: acceso a datos	
Atributo	Tipo
Para cada responsabilidad:	

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Nombre:	BuscarPersona(aCodigoBarras : String, aNumeroUCI : String, aCI : String) : Persona
Descripción:	Devuelve la persona.
Nombre:	BuscarCredenciales(aTipoCredencial : String) : ArrayList
Descripción:	Busca todas las credenciales activas.
Nombre:	BuscarTiposCredencial() : ArrayList
Descripción:	Busca todos los tipos de credenciales existentes.
Nombre:	EliminarPersonas() : boolean
Descripción:	Elimina personas en la BD local.
Nombre:	RegistrarPersonas(aListaPersonas : ArrayList) : void
Descripción:	Registra en la BD local la lista de persona buscada en la BD remota.
Nombre:	BuscarPersonas() : ArrayList
Descripción:	Devuelve un listado con todas las personas que están en la remota.

Tabla 52 Clase de Acceso a Datos: “ControlarPersona_DAO_Imple”.

Clase de acceso a datos: “ControlAccesoVehiculo_DAO_Imple”.

Nombre: ControlAccesoVehiculo_DAO_Imple	
Tipo de clase: acceso a datos	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	EliminarAccesoVehiculo(aGuid : String, aFechaHora : Datetime) : boolean
Descripción:	Elimina el último acceso introducido en la BD local.
Nombre:	ObtenerAccesosVehiculos() : ArrayList
Descripción:	Devuelve un listado con todos los accesos de la BD local.
Nombre:	EliminarAccesosVehiculos() : void

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Descripción:	Elimina todos los accesos de la BD local.
Nombre:	RegistrarAccesosVehiculos(aListaAccesosVehiculos : ArrayList, Nombre_Cliente : String) : boolean
Descripción:	Registra todos los accesos de los vehículos desde la BD local hacia la remota.
Nombre:	RegistrarAccesoVehiculoLocal(aAccesoVehiculo : AccesoVehiculo) : boolean
Descripción:	Se registran los accesos de los vehículos en la BD local.

Tabla 53 Clase de acceso a datos: “ControlAccesoVehiculo_DAO_Imple”.

Clase de acceso a datos: “Control_Vehiculo_DAO_Imple”.

Nombre: Control_Vehiculo_DAO_Imple	
Tipo de clase: acceso a datos	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	BuscarVehiculo(aChapa : String) : Vehiculo
Descripción:	Devuelve el vehículo.
Nombre:	BuscarCredenciales(aTipoCredencial : String) : ArrayList
Descripción:	Busca todas las credenciales activas.
Nombre:	BuscarVehiculos() : ArrayList
Descripción:	Devuelve un listado con todos los vehículos que están en la remota.
Nombre:	EliminarVehiculo() : boolean
Descripción:	Elimina vehículos en la BD local.
Nombre:	RegistrarVehiculo(aListaVehiculo : ArrayList) : void
Descripción:	Registra en la BD local la lista de vehículos buscada en la BD remota.

Tabla 54 Clase de acceso a datos: “Control_Vehiculo_DAO_Imple”.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Clase de acceso a datos: “Controlar Visitante_DAO_Imple”.

Nombre: Controlar Visitante_DAO_Imple	
Tipo de clase: acceso a datos	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	BuscarVisitante_x_Ci(Ci : String) : Visitante
Descripción:	Devuelve el visitante.
Nombre:	EntrarVisitante(Nombre : String, NoCI : String, Organismo : String, LugarVisitar : String, DirigenteAut : String, Paquetes : boolean, FechaE : String, FechaS : String, NomChofer : String, CiChofer : String, LicConduccion : String, Circulacion : String, Marca : String, Chapa : String, TipoVehiculo : String) : void
Descripción:	Guarda en la BD local los datos del visitante.
Nombre:	EliminaVisitante() : boolean
Descripción:	Elimina visitantes en la BD local.
Nombre:	RegistrarVisitantes(aListaVisitantes : ArrayList) : void
Descripción:	Registra en la BD local la lista de visitantes buscada en la BD remota.
Nombre:	BuscarVisitantes() : ArrayList
Descripción:	Devuelve un listado con todos los visitantes que están en la remota.

Tabla 55 Clase de acceso a datos: “Controlar Visitante_DAO_Imple”.

Clase de acceso a datos: “AccesoVisitanteControl_DAO_Imple”.

Nombre: AccesoVisitanteControl_DAO_Imple	
Tipo de clase: acceso a datos	
Atributo	Tipo
Para cada responsabilidad:	

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Nombre:	RegistrarAccesoVIsitante(accesovisitante : AccesoVisitante) : boolean
Descripción:	Se registran los accesos de los visitantes en la BD local.
Nombre:	EliminarAccesoVisitanteLocal(CI : String, aFechaHora : Datetime) : boolean
Descripción:	Elimina el último acceso introducido en la BD local.
Nombre:	ObtenerAccesoVisitante() : ArrayList
Descripción:	Devuelve un listado con todos los accesos de la BD local.
Nombre:	EliminarAccesosVisitantes() : void
Descripción:	Elimina todos los accesos de la BD local.

Tabla 56 Clase de acceso a datos: “AccesoVisitanteControl_DAO_Imple”.

Clase de acceso a datos: “Control_Configuracion_DAO_Imple”.

Nombre: Control_Configuracion_DAO_Imple	
Tipo de clase: acceso a datos	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	CalcularProximaSincronizacion(Periodo : int, UltimaFecha : DateTime) : DateTime
Descripción:	Devuelve la próxima fecha en la que se va a sincronizar.
Nombre:	BuscarPuertasNoActivas() : ArrayList
Descripción:	Busca las puertas que no están en uso.
Nombre:	VerificarUsuario(aUsuario : String, aPassword : String) : boolean
Descripción:	Verifica que el usuario tiene permiso para acceder a la aplicación.
Nombre:	ObtenerCampoLocal(aNombreCampo : String) : String
Descripción:	Devuelve un campo de la BD.
Nombre:	ObtenerListadoDatos() : ArrayList

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Descripción:	Obtiene un listado con los campos en la BD.
Nombre:	Actualizar_Campos_Locales(P_Sinc_Persona : int, F_Sinc_Persona : Datetime, P_Sinc_Accesos_Persona : int, P_Sinc_Vehiculo : int, F_Sinc_Vehiculo : Datetime, P_Sinc_Acceso_Vehiculo : int, P_Sinc_Visitante : int, F_Sinc_Visitante : Datetime, P_Sinc_Acceso_Visitante : int) : void
Descripción:	Actualiza en la BD local los datos introducidos.
Nombre:	ActualizarCampoLocal(aCampo : String) : boolean
Descripción:	Actualiza un campo en la BD local.
Nombre:	ObtenerCausaAcceso() : ArrayList
Descripción:	Devuelve un listado con las causas de acceso.
Nombre:	RegistrarCausaAcceso(aCausaAcceso : String) : boolean
Descripción:	Guarda en la BD local causas de acceso.

Tabla 57 Clase de acceso a datos: “Control_Configuracion_DAO_Imple”.

Clase de acceso a datos: “CC_Control_Sincronizacion_DAO_Imple”.

Nombre: CC_Control_Sincronizacion_DAO_Imple	
Tipo de clase: acceso a datos	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	SincronizarAccesosPersonas(aNombreCliente : String) : void
Descripción:	Sube los accesos de personas de la BD local a la remota.
Nombre:	SincronizarAccesosVehiculos(aNombreCliente : String) : void
Descripción:	Sube los accesos de vehículos de la BD local a la remota.
Nombre:	SincronizarAccesosVisitantes(aNombreCliente : String) : void
Descripción:	Sube los accesos de visitantes de la BD local a la remota.
Nombre:	SincronizarPersonas() : void

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Descripción:	Baja listado de personas de la BD remota a la local.
Nombre:	SincronizarVehiculos() : void
Descripción:	Baja listado de vehículos de la BD remota a la local.
Nombre:	SincronizarVisitantes() : void
Descripción:	Baja listado de visitantes de la BD remota a la local.

Tabla 58 Clase de acceso a datos: “CC_Control_Sincronizacion_DAO_Imple”.

Clase de acceso a datos: “Passback_DAO_Imple”.

Nombre: PassBack_DAO_Imple	
Tipo de clase: acceso a datos	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	Verificar_PassBack(aGuid : String, Tipo : TipoAcceso) : boolean
Descripción:	Verifica que exista PassBack.
Nombre:	RegistrarPassback(passback :PassBack) : void
Descripción:	Guarda en la BD el motivo del PassBack.

Tabla 59 Clase de acceso a datos: “Passback_DAO_Imple”.

3.6 Conclusiones

En este capítulo se realizó una modelación del sistema en términos del análisis y del diseño de cada caso de uso, identificándose las clases interfaces, control, entidad y de acceso a datos, para dar una mejor explicación del diseño realizado.

Conclusiones

En el presente trabajo se demostró la necesidad de proponer el desarrollo de una aplicación de escritorio, para solucionar los problemas existentes referidos a la no existencia de un sistema, que controle de forma integrada los accesos del personal, vehículos y visitantes y que se ajuste a las políticas establecidas por la UCI. Con este fin se realizó el análisis y diseño de dicha aplicación para sentar las bases para el desarrollo de la misma dándole cumplimiento al objetivo propuesto, que incluirá mejoras como, eliminar el trabajo manual, el gasto de materiales y demoras en los accesos.

Para el desarrollo de este trabajo se siguieron los pasos que propone la metodología de desarrollo Rational Unified Process (RUP), se utilizó el Lenguaje Unificado de Modelado (UML) y Visual Paradigm para el modelado visual.

Se puede concluir que el análisis y diseño del “Sistema de Control de Acceso en la UCI” facilita su futura implementación, permitiendo agilizar el registro y control de los accesos, mediante una visión más ampliada y detallada de los requerimientos del sistema para el control de acceso.

Recomendaciones

Se recomienda:

- ❖ Continuar con la implementación de la aplicación para el sistema de control de acceso en próximas iteraciones.
- ❖ Adquirir más lectores de código de barras para agilizar el proceso de control de acceso.
- ❖ Evaluar la realización de una propuesta para un sistema de propósito general, que pueda ser utilizado en cualquier centro para controlar el acceso al mismo.

Bibliografía Referenciada

[1] Pérez Zurita, Paúl, Oña Cruz, Tiuska Lilia: SISTEMA DE CONTROL DE ACCESO A LA UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS. 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 2006.

[2] Tendencias actuales de los sistemas de control de acceso. [Disponible en: www.kimaldi.com]

[3] Prácticas de Ingeniería de Software, Una herramienta CASE para ADOO: Visual Paradigm [Disponible en: http://alarcos.inf-cr.uclm.es/per/fgarcia/isoftware/doc/LabTr1_VP.pdf]

Bibliografía Consultada

Jacobson, Ivar y Booch, Grady y Rumbaugh, James. El proceso unificado de software. Primera edición. Pearson Educación, S.A. 2000.

Pressman. R. Ingeniería de software. Un enfoque práctico. 2002. Capítulos 1, 2, 3, y 11. Páginas 3-48 y 181-195. Capítulos 5, 13 Páginas 55-58, 120-121, 157-162, 365.

JAMES RUMBAUGH, I. J., GRADY BOOCH *El Proceso Unificado de Desarrollo*. 1. Cuba, Félix Varela. La Habana, 29 abril de 2004. 136 p.

Booch, G., Rumbaugh, J., Jacobson, I. “El Lenguaje Unificado de Modelado”. Addison-Wesley. 1999.

Colaboradores de Wikipedia. Lenguaje Unificado de Modelado (2007). [Disponible en: <http://es.wikipedia.org/wiki/UML>]

Sistema de control de acceso a nivel internacional. [Disponible en: <http://www.portovilagarcia.com>]

BIBLIOGRAFÍA

María A. Mendoza Sánchez. Metodologías De Desarrollo De Software. (2004). [Disponible en:http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html]

Alarcos, G. Visual Paradigm. [En línea] enero de 2007. [Disponible en: <http://www.macupdate.com/info.php/id/16945/visual-paradigm-for-uml>].

VISUAL-PARADIGM. Herramienta CASE para el para el modelado UML, 2007. [Disponible en: <http://www.visual-paradigm.com/product/vpuml/>]

Wikipedia, Colaborades de . 2008. Wikipedia [Disponible en: http://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n_Java]

Conferencia_2. Fase de inicio. Modelo de negocio. [Disponible en: <http://teleformacion.uci.cu/mod/resource/view.php?id=6680>]

Conferencia_4.Flujo de trabajo Requerimientos. [Disponible en: <http://teleformacion.uci.cu/mod/resource/view.php?id=8865>]

Conferencia_5. Estimación y planificación de proyectos [Disponible en: <http://teleformacion.uci.cu/mod/resource/view.php?id=9007>]

Conferencia_6. Fase de elaboración. Flujo de trabajo de Análisis y diseño [Disponible en: <http://teleformacion.uci.cu/mod/resource/view.php?id=10349>]

LARMAN, C. *Intruducción al Analisis y diseño Orientado a Objeto*. 1. Cuba, Félix Varela 29 abril de 2004. 291 p.

REF Larman, Craig: UML y patrones, introducción al análisis y diseño orientado a objetos, Félix Varela, 2004.

GAMMA. E. HELM. R. JOHNSON. R. VLISSIDES.J. Patrones de diseño, 2000. [Disponible en: <http://www.vico.org/pages/PatronsDisseny.html>]

BIBLIOGRAFÍA

MBCESStore. [Disponible en: <http://www.mbcestore.com.mx/symbol/p460.htm>]

POSIFLEX. [Disponible en: <http://www.posiflex.es/productes/scanners.htm>]

Wikipedia, Colaborades de. 2008. Wikipedia. [Disponible en: <http://es.wikipedia.org/wiki/DB4O,DB4O>]

Java en Castellano. 2007 [Disponible en:

http://www.programacion.net/java/articulo/jap_persis_db4o/, Persistencia de Objetos Java utilizando db4o]

Glosario

Autenticar: Acción que ocurre cuando se requiere verificar si la persona que quiere acceder a algo es la permitida por el sistema o no, por tanto debe introducir su usuario y contraseña.

Biometría: Estudio estadístico y matemático de los fenómenos biológicos.

CI: Carné de identidad.

DCA: Diagrama de clases del análisis.

DCD: Diagrama de clases del diseño.

Despliegue: Ocurre cuando varios trabajos mas o menos independientes (flujos de control, procesos) se distribuyen entre diferentes dispositivos hardware.

Diagrama: Representación gráfica de una colección de elementos de modelado. Un diagrama muestra representaciones de elementos semánticos del modelo, pero su significado no se ve afectado por la forma en que son representados.

Display: Pantalla o indicador numérico utilizado para visualizar una determinada información de un aparato electrónico.

Ergonómico: De la ergonomía, relacionado con esta ciencia o que respeta sus principios: asiento ergonómico; posición ergonómica.

Interfonos: Red y aparato telefónico utilizado para las comunicaciones internas entre despachos de un mismo edificio.

Molinetes: Aparato provisto de un eje con aspas que permite el paso de las personas de una en una.

PassBack: Acción que ocurre cuando el acceso se registra 2 veces continuas hacia la misma dirección, es decir, una persona no puede registrar su entrada cuando está registrado que está adentro o viceversa una persona no puede registrar su salida cuando no ha registrado su anterior entrada.

Procesador: Circuito integrado de un ordenador capaz de efectuar el tratamiento completo de una serie de datos.

Radiofrecuencia: Cualquiera de las frecuencias de las ondas electromagnéticas empleadas en la radiocomunicación.

Relé: Dispositivo que, intercalado en un circuito, produce determinadas modificaciones en el mismo o en otro conectado con él.

Release del proyecto: Nueva versión de una aplicación informática.

Tornos: Armazón giratorio que sirve para pasar de una parte a otra.

UCI: Universidad de las Ciencias Informáticas.