

**Universidad de las Ciencias Informáticas
Facultad 1**



**SISTEMA DE GESTION DE CREDENCIALES
EN LA UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS**

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

Autor:

Guillermo Gómez Urquiza

Tutor:

Ing. Manuel Alejandro Gil Martín

Ciudad de La Habana, Cuba

Junio, 2007

DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor de este trabajo y autorizo a la Dirección de Informatización de la Universidad de las Ciencias Informáticas; así como a dicho centro para que hagan el uso que estimen pertinente con este trabajo.

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

Autor: Guillermo Gómez Urquiza

Tutor: Manuel Alejandro Gil Martín

OPINIÓN DEL TUTOR DEL TRABAJO DE DIPLOMA

Título: **Sistema de Gestión de Credenciales en la Universidad de las Ciencias Informáticas**

Autor: Guillermo Gómez Urquiza

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 __ puntos.

Tutor: Manuel Alejandro Gil Martín

Firma

Fecha

AGRADECIMIENTOS

Quisiera agradecer especialmente a mi familia, por todo lo que han podido aportar en mi formación.

Un profundo agradecimiento a Manuel Alejandro Gil, mi tutor, por sus consejos, ideas y guía.

A Danay, por su cariño y comprensión en todo momento.

A todos los que de una forma u otra pusieron su granito en este trabajo, especialmente a Rodolfo, Jeandy, Adisley, Imirys, Damián.

A Michel Pérez Vargas, por inspirarme a ser mejor cada día.

A todos mis compañeros con los que he compartido estos cinco años.

A los profesores, a la Universidad, a la Revolución y en especial al Comandante en Jefe Fidel Castro Ruz, por haber hecho posible formarme como profesional para el desarrollo de la sociedad.

DEDICATORIA

A mis padres...

RESUMEN

La identificación de personas en el marco de la Universidad de las Ciencias Informáticas toma especial importancia a partir de la gran cantidad de personas que de alguna forma reside o tiene acceso a ella. Hoy en día existe un sistema automatizado en la universidad, pero que no se ajusta a las nuevas medidas de la Dirección de Seguridad y Protección, pues está muy atado al modelo de datos que existía cuando se creó así como a las reglas de negocios existentes en aquel momento. Debido a cambios en otros sistemas de la UCI que almacenan información de las personas, había que hacer soluciones “temporales” que luego permanecían permanentemente en el sistema, comprometiendo el buen funcionamiento y la seguridad del mismo.

De esta manera surge la necesidad de crear un nuevo Sistema de Gestión e Impresión de Credenciales que tiene como objetivo concreto, rediseñar e implementar todo el proceso de la acreditación de personas, haciéndolo lo suficientemente flexible como para poder utilizarlo en otros entornos de negocios. Este sistema vinculado a otros que funcionan en la Universidad como son el de Control de Acceso, Control de Acceso a los Comedores, al Sistema de Reservación de Pases puede convertirse en una gran herramienta de control para la Dirección de la UCI. En este se añaden nuevas funcionalidades como la organización de credenciales en grupos definidos por el usuario de la aplicación, vistas previas a la impresión y se mantiene un control del historial de las credenciales, pudiéndose conocer cómo fue una credencial en un momento dado.

Con el presente trabajo se muestra un estudio realizado en la UCI para la construcción del sistema. Se explican además conceptos relacionados con él y se hace una propuesta.

ÍNDICE

INTRODUCCIÓN	1
CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA	5
INTRODUCCIÓN	5
TECNOLOGÍAS Y TENDENCIAS ACTUALES	5
Sistemas automatizados en el mundo relacionados con la identificación	5
Tecnologías de hardware utilizadas para la identificación.....	6
Tecnologías de software a utilizar.....	9
FUNDAMENTACIÓN DE LA METODOLOGÍA	14
HERRAMIENTAS UTILIZADAS	15
CONCLUSIONES	18
CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA	19
INTRODUCCIÓN	19
OBJETO DE ESTUDIO	19
PROBLEMA Y SITUACIÓN PROBLÉMICA	19
OBJETO DE AUTOMATIZACIÓN	20
INFORMACIÓN QUE SE MANEJA	21
PROPUESTA DE SISTEMA	21
MODELO DE NEGOCIO	22
Descripción de actores y trabajadores.....	23
Representación gráfica del diagrama de casos de uso del negocio.....	23
ESPECIFICACIÓN DE LOS REQUISITOS DE SOFTWARE	29
Requerimientos Funcionales.....	29
Requerimientos no Funcionales.....	29
MODELO DE CASOS DE USO DEL SISTEMA	30
Definición de los actores del sistema a automatizar	31
Listado de casos de uso	31
Descripción de los Diagramas de Caso de Uso del Sistema.....	33
CONCLUSIONES	41
CAPÍTULO 3. DISEÑO DEL SISTEMA	42
INTRODUCCIÓN	42
DIAGRAMA DE CLASES DEL DISEÑO	43
DESCRIPCIÓN DE LAS CLASES DE DISEÑO	49

Clases de la Interfaz	49
Clases del Negocio	51
Clases de Datos	53
Clases de Comunicación	54
Clases de Impresión	55
Clases Comunes	57
Clases del ProveedorUCI	58
DISEÑO DE LA BASE DE DATOS.....	59
Descripción de las tablas	60
TRATAMIENTO DE ERRORES	61
SEGURIDAD	61
INTERFAZ	62
CONCEPCIÓN DE LA AYUDA.....	62
CONCLUSIONES.....	62
CAPÍTULO 4. IMPLEMENTACIÓN.....	63
INTRODUCCIÓN.....	63
DIAGRAMA DE DESPLIEGUE.....	63
DIAGRAMA DE COMPONENTES.....	64
CONCLUSIONES.....	65
CAPÍTULO 5. ESTUDIO DE FACTIBILIDAD.....	67
INTRODUCCIÓN.....	67
PLANIFICACIÓN BASADA EN CASOS DE USO.....	67
COSTO DEL PROYECTO.....	73
BENEFICIOS TANGIBLES E INTANGIBLES	73
ANÁLISIS DE COSTOS Y BENEFICIOS	74
CONCLUSIONES.....	74
REFERENCIAS BIBLIOGRÁFICAS	77
BIBLIOGRAFÍA.....	78
GLOSARIO DE TÉRMINOS.....	80
ANEXOS	81

ÍNDICE DE TABLAS

Tabla 1 Descripción de actores del negocio	23
Tabla 2 Descripción de trabajadores del negocio	23
Tabla 3 Descripción del caso de uso del negocio "Devolver Credencial"	24
Tabla 4 Descripción del caso de uso del negocio "Solicitar Credencial"	26
Tabla 5 Descripción CU Pagar Credencial	28
Tabla 6 Definición de actores del sistema	31
Tabla 7 Caso de Uso Modificar "Nombre Grupo"	31
Tabla 8 Caso de Uso "Crear Grupos"	31
Tabla 9 Caso de Uso "Visualizar Credencial"	32
Tabla 10 Caso de Uso "Eliminar Grupos"	32
Tabla 11 Caso de uso "Imprimir Credencial"	32
Tabla 12 Caso de Uso "Actualizar Credencial"	32
Tabla 13 Caso de Uso "Filtrar Credenciales"	33
Tabla 14 Caso de Uso "Mostrar Listado de Credenciales"	33
Tabla 15 Caso de Uso "Crear Nueva Credencial"	33
Tabla 16 Caso de Uso "Busqueda Origen Datos"	33
Tabla 17 Descripción extendida CU "Modificar Nombre Grupos"	34
Tabla 18 Descripción extendida CU "Crear Grupos"	35
Tabla 19 Descripción extendida CU Visualizar Credencial.....	36
Tabla 20 Descripción extendida CU Eliminar Grupos.....	36
Tabla 21 Descripción extendida CU "Imprimir Credencial"	37
Tabla 22 Descripción extendida CU "Actualizar credencial"	37
Tabla 23 Descripción extendida CU "Filtrar Credenciales"	38
Tabla 24 Descripción extendida la salida de los pases.	38
Tabla 25 Descripción extendida CU "Crear Nueva Credencial"	39
Tabla 26 Descripción extendida CU "Busqueda Origen Datos"	39
Tabla 27 Descripción de la clase "Gestor"	50
Tabla 28 Descripción de la clase "CrearGrupo"	50
Tabla 29 Descripción de la clase "Modificar Grupo"	50
Tabla 30 Descripción de la Clases "Vista previa"	51
Tabla 31 Descripción de la clase "Impresión"	51
Tabla 32 Descripción de la clase "Gestor Credenciales"	51
Tabla 33 Descripción de la clase ""	52
Tabla 34 Descripción de la clase "Gestore Proveedor"	52
Tabla 35 Descripción de la clase "FabricaProveedor"	52
Tabla 36 Descripción de la clase "GestorPlantillas"	52
Tabla 37 Descripción de la clase "Comunicador"	53
Tabla 38 Descripción de la clase "CtrlCredenciales"	53
Tabla 39 Descripción de la clase "CtrlGrupos"	54
Tabla 40 Descripción de la clase "IComunicador"	54
Tabla 41 Descripción de la clase "IProveedor"	55
Tabla 42 Descripción de la clase "GestorImpresion"	55
Tabla 43 Descripción de la clase "Hoja"	55
Tabla 44 Descripción de la clase "Tarjeta"	56

Tabla 45 Descripción de la clase "Lector Xml"	56
Tabla 46 Descripción de la clase "Configuracion"	57
Tabla 47 Descripción de la clase "Grupo"	57
Tabla 48 Descripción de la clase "Credencial"	57
Tabla 49 Descripción de la clase "Campo"	57
Tabla 50 Descripción de la clase "Constantes"	58
Tabla 51 Descripción de la clase "ProveedorUCI"	59
Tabla 52 Descripción de la tabla grupo	60
Tabla 53 Descripción de la tabla credencial	61
Tabla 54 Descripción de la tabla campo	61
Tabla 55 Factor de peso de los actores sin ajustar	68
Tabla 56 Factor de peso de los casos de uso sin ajustar	68
Tabla 57 Factor de complejidad técnica	70
Tabla 58 Factor de ambiente	71
Tabla 59 Esfuerzo del proyecto	72

ÍNDICE DE FIGURAS

Figura 1. 1 Código 39	9
Figura 2. 1 Diagrama de casos de uso del negocio.....	24
Figura 2. 2 Diagrama de actividades CU del Negocio "Devolver Credencial"	25
Figura 2. 3 Diagrama de Actividad CU "Solicitar Credencial"	27
Figura 2. 4 Modelo de objetos	28
Figura 2. 5 Diagrama Casos de Uso del Sistema	40
Figura 3. 1 Diagrama de Clases "Interfaz"	43
Figura 3. 2 Diagrama de Clases "Negocio"	44
Figura 3. 3 Diagrama de Clases "Datos"	45
Figura 3. 4 Diagrama de Clases "Comunicación"	46
Figura 3. 5 Diagrama de Clases "Comunes"	47
Figura 3. 6 Diagrama de Clases "Impresion"	48
Figura 3. 7 Diagrama De Clases "ProveedorUCI"	49
Figura 3. 8 Diagrama Clases Persistentes	59
Figura 3. 9 Diagrama Entidad Relación	60
Figura 4. 1 Diagrama de despliegue.....	64
Figura 4. 2 Diagrama de componentes.....	65

INTRODUCCIÓN

La Universidad de las Ciencias Informáticas (UCI) constituye hoy en día una ciudad. A ella accede diariamente un gran flujo de personas, así como las que residen, esto hace que sea necesario encontrar una forma eficiente para identificarlas. Actualmente es considerado un problema el hecho de que el número de personas que conviven y/o trabajan en la Universidad se haya incrementado con el tiempo, debido a los nuevos ingresos de estudiantes y a las contrataciones de servicios de trabajo, entre otros, lo cual ha provocado que el trámite de acreditar a las personas tenga que ser más dinámico, lo cual conlleva al **problema**: El sistema de credenciales actual en la Universidad de las Ciencias Informáticas no permite una adecuada gestión para la variedad de situaciones en el presente.

En la Universidad existe un Sistema desarrollado desde hace algunos años, que se encarga de la impresión y gestión de las credenciales de las personas. Hasta el momento, este sistema sigue cumpliendo esas funciones; pero el mismo ya no resulta factible, producto a las nuevas necesidades surgidas con vistas a mejorar la seguridad del personal y las instalaciones universitarias, definiéndose la **situación problemática**: El sistema que existe en la actualidad no puede ajustarse a las nuevas regulaciones de la Dirección de Seguridad y Protección, puesto que está muy ligado al modelo de datos de la universidad de aquel entonces y a las reglas de negocios sobre las que se desarrolló el mismo. Este sistema estaba estrechamente vinculado a otros del área de la Dirección de Informatización y al ir evolucionando estos últimos, fue preciso encontrar soluciones temporales para que el sistema siguiera prestando servicios, soluciones que pasaron a ser permanentes en la aplicación, comprometiendo el buen funcionamiento de la misma.

Por todo ello, se hace necesario diseñar un nuevo sistema de impresión y gestión de credenciales capaz de ajustarse a las necesidades actuales, garantizando no tener que hacer grandes cambios futuros y que sea posible su reutilización en otros entornos.

Dicho sistema deberá poseer una interfaz amigable, sencilla, con una amplia capacidad de configuración, reduciendo al mínimo la posibilidad de cambios; utilizará un nuevo tipo de soporte para la impresión de credenciales que permitirá una mayor duración de las credenciales y deberá ser capaz de ajustarse a cualquier entidad para acreditar personas u objetos según sea necesario. Este sistema deberá realizar filtrados según los campos que conformen las credenciales, además mostrará una vista previa de la

credencial a imprimir, así como las vistas de las credenciales anteriores que ha tenido esa persona. También agrupará a las personas en grupos de forma intuitiva, con el objetivo de lograr que el usuario pueda organizar las credenciales según estime conveniente, lo que hace que el sistema pueda utilizarse en otras instituciones.

Con el uso de esta aplicación se mejorará el proceso de gestión e impresión de credenciales ya que la búsqueda y organización de las mismas se hará de forma sencilla, permitiendo que su impresión se realice de forma rápida, además se podrá conocer el historial de las credenciales dado un grupo, así como su estado cuando se encontraban activas.

La acreditación en la UCI consiste en entregarle a cada persona que tiene acceso al centro una credencial que la identifique como autorizada a permanecer a tiempo completo o no en la institución, además dicha credencial será portadora de información acerca de qué tipo de persona es, dígame Profesor, Trabajador, Estudiante, Dirigente, Tercerizados, o de cualquier otro tipo que la Dirección de Seguridad y Protección decida crear a fin de tener un mayor control en el centro.

Por lo tanto el **objeto de estudio** es la Seguridad y Protección en la UCI, específicamente lo referido a la acreditación de personas.

De donde se deriva que nuestro **campo de estudio** queda enmarcado específicamente en la carnetización en la Universidad.

Como **hipótesis** se tiene que el desarrollo de una nueva aplicación permitirá manejar adecuadamente la gestión e impresión de credenciales en la universidad.

De ahí que el **objetivo general** sea desarrollar una aplicación que permita gestionar e imprimir credenciales en la UCI o en cualquier otro entorno, mejorando así este proceso.

Por lo que como **objetivos específicos** tenemos:

Confección de un sistema de impresión y gestión de credenciales que permita:

- Crear credenciales a personas y entidades en una institución.
- Ser independiente del origen de datos.
- Imprimir y mostrar vista previas de las credenciales.
- Brindar facilidades de búsqueda dado ciertos criterios.
- Organizar de manera intuitiva las credenciales.

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

- Realizar un estudio del entorno de trabajo.
- Identificar las necesidades del cliente.
- Declarar los requisitos que debe cumplir el sistema.
- Modelar conceptualmente las clases que están implicadas en el sistema.
- Desarrollar los diagramas de actividad.
- Desarrollar los diagramas que describen el diseño del sistema.
- Describir las clases del diseño.
- Diseñar la Base de Datos.
- Diseñar la interfaz.
- Implementar la aplicación.

Este trabajo ha sido organizado de la siguiente manera:

Capítulo 1: Incluye un estado del arte del tema tratado, a nivel internacional, nacional y de la Universidad, además se mencionan las tendencias, técnicas, tecnologías, metodologías y software usados en la actualidad o en las que se apoya para la solución del problema que se enfrenta, los cuales resultan la base fundamental para el desarrollo e implementación de la aplicación.

Capítulo 2: Describe cómo se realiza el proceso de acreditación en la Universidad actualmente. En él se mencionan los principales problemas que generaron la necesidad del cambio; se obtienen los procesos que son objetos de automatización, la propuesta del sistema y los requisitos funcionales y no funcionales del mismo.

También describe el negocio a través de un Modelo de Negocio, así como las reglas a considerar, los actores y trabajadores que intervienen, los casos de uso del negocio así como el diagrama de ellos y el modelo de objetos; además los actores y casos de uso del sistema son descritos y se representa gráficamente el diagrama de casos de uso del sistema.

Capítulo 3: Trata el diseño del sistema a desarrollar utilizando herramientas de modelación para mostrar detalladamente, los principales procesos que ocurren. Se representan el diagrama de clases del diseño, el diagrama de interacción, el diagrama de clases y el diagrama entidad relación con sus descripciones respectivamente.

Capítulo 4: Aborda la implementación, contiene los diagramas de despliegue y de componente, así como una breve descripción de estos conceptos.

Capítulo 5: Muestra un estudio de factibilidad para la planificación y construcción del sistema, se hace un análisis de costo y se mencionan los beneficios tangibles e intangibles del desarrollo de la propuesta.

CAPÍTULO 1. Fundamentación teórica

Introducción

El presente capítulo refiere las tecnologías actuales para la identificación de personas, así como las herramientas y software para el desarrollo de la solución. Se ejemplifican algunos sistemas automatizados relacionados con la identificación que se utilizan en el ámbito internacional.

En Cuba no existen en la actualidad sistemas de gestión e impresión de credenciales; pero, en la Universidad de las Ciencias Informáticas, se puede mencionar un anterior Sistema de Gestión de Credenciales, que se encuentra en funcionamiento desde hace tres años. Con este sistema se han impreso todas las credenciales del personal vinculado a la UCI hasta el momento.

Tecnologías y tendencias actuales

En el mundo existe una gran variedad de tecnologías con el fin de facilitar la identificación de personas y medios. Esto se debe a que en cualquier negocio o institución se hace necesario tener en mayor o menor medida cierto control sobre las personas que acceden al mismo. Teniendo en cuenta las exigencias propias de cada entidad, se debe escoger un sistema apropiado y el factible económicamente; lo ideal, por tanto, es hallar la solución que lo garantice.

Sistemas automatizados en el mundo relacionados con la identificación

En el mundo existe un gran número de sistemas que de una forma u otra, se relacionan con la identificación. Algunos de los más conocidos son:

Number Five CardFive

Es un software para el diseño de credenciales con diseño personalizado y soporte de las mayores impresoras de tarjetas PVC (un material plástico). El mismo ofrece flexibilidad para la creación y diseño de las propias credenciales PVC. El diseño a utilizar es avanzado y amigable. En sus distintas versiones

permite utilizar los datos automáticamente mediante el acceso a una base de datos existente, donde se encuentra la información necesaria para la impresión en la tarjeta plástica.

Evolis eMedia Software

Es un programa que permite al usuario el diseño personalizado de credenciales para su posterior salida en las impresoras de tarjetas de PVC. En sus distintas versiones permite utilizar los datos automáticamente accediendo a una base de datos existente, la cual posee la información necesaria para la impresión en la tarjeta plástica. Posee un alto nivel de configuración y trabajo con imágenes. Este software es líder en impresoras de PVC de alta calidad a nivel mundial. Tiene más de 10 años de experiencia y ha sido usado en más de 70 países.

DataCard Id Work Id Cards Software

Es un diseño de aplicación personalizable para el programa de tarjetas de identificación. El software de identificación de ID Works mejora cada aspecto del diseño de tarjeta y reportes, así como la producción. Es extremadamente fácil de usar, pues tiene un diseño flexible y modular, permite seleccionar sólo los componentes específicos que se necesitan. DataCard es una compañía con más de 30 años de experiencia, con una marcada connotación internacional en lo que a soluciones informáticas respecta.

Tecnologías de hardware utilizadas para la identificación

Claves por Teclado

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

Tarjetas de Banda Magnética

Es la tecnología más conocida y difundida, ya que se utiliza en todos los sistemas de tarjetas de crédito y compra (de hecho, se pueden utilizar esas mismas tarjetas en muchos sistemas de control de acceso). Su ventaja es su difusión, popularidad y bajo costo, pero en sí, es, de todos los medios de identificación, el

más vulnerable. La banda magnética de la tarjeta, debe ser tratada con cierto cuidado para evitar que se raye o sea expuesta a campos magnéticos que puedan borrarla, por tal motivo, no son recomendables para usar en ambientes industriales. Sólo se recomiendan en oficinas o establecimiento administrativos.

Tarjetas de Código de Barras

El código de barras es una tecnología de identificación automática. Permite recolectar datos con precisión y rapidez. Un código de barras consiste en una serie de barras adyacentes paralelas y espacios. Los diseños predeterminados de anchura se utilizan para codificar datos en el código. Para leer información en un símbolo de código de barras, un dispositivo de lectura, tal como un lápiz óptico, se desliza a través del símbolo de un lado al otro, la anchura de barras y los espacios son analizados por el decodificador del lector, y los datos originales se recuperan. La aplicación más visible en esta tecnología es en la industria de supermercado, donde ha estado en el uso desde 1970.

Touch Memories

Se trata de una pastilla electrónica, encapsulada en acero inoxidable de unos 16 mm., tan práctica que puede ser colgado en un llavero.

Se les denomina comúnmente llave electrónica y brindan un alto nivel de seguridad, ya que son altamente resistentes al desgaste, siendo ideales para ambientes industriales, no así para ambientes con alto grado de generación de corriente estática (Ej.: oficinas con mucha alfombra y ambientes muy secos). Su tecnología de avanzada evita la posibilidad de duplicarlas. Por su costo es de destacar que son unos de los medios más caros, sin embargo, nunca se desgastan y son muy confiables.

Tarjetas de Proximidad o Radio Frecuencia (RF)

Estas tarjetas poseen un diseño tecnológico que las hace prácticamente imposibles de duplicar, lo cual las convierte en una de las tecnologías más modernas y efectivas. Son muy prácticas y de bajo costo de mantenimiento, idóneas en situaciones de máxima seguridad y alta tecnología, incluso pueden ser leídas dentro de una billetera, una cartera, etc.

Estas tarjetas son llamadas también de Radio Frecuencia (RF) y estos sistemas constan de dos partes; la unidad lectora y el tag, transponder o tarjeta. Se denomina proximidad activa cuando las tarjetas tienen alimentación interna, y proximidad pasiva cuando no tienen esa batería interna, esta última es la más

usada, el lector emite una onda electromagnética que genera una corriente interna en la tarjeta, proporcionando su alimentación, dentro de ésta existe un microchip que tiene una memoria que guarda el código y una antena que es la encargada de recibir y transmitir, al inducirse esa corriente en la tarjeta, extrae de la memoria el dato codificado y lo devuelve, enviándolo al lector, el mismo lo recibe, lo decodifica, lo filtra, lo amplía y lo envía a la unidad de control de accesos a la que esté conectada.

Sistemas Biométricos

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

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

Tecnología utilizada en la UCI: El código de barras

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

Código 39

Código 39 ASCII Total

Codabar

Intercalado 2 de 5

Código 128

UPC (Código Universal de Producto)

EAN (European Article Numbering o Sistema de Numeración Europeo)

En la UCI se utiliza el **Código 39**, que 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 cantidad de dígitos. Este formato ha pasado a ser la norma para el gobierno, la industria, la educación y las aplicaciones de negocios.



Figura 1. 1 Código 39

Beneficios del código de barras

Los beneficios de utilizar códigos de barras como sistema de recolección de datos automática son muy sencillos: la velocidad y exactitud. Una y otra vez se ha probado que capturar datos con códigos de barras es por lo menos 100 veces más rápido y más exacto que la captura normal por teclado, que se traduce en un aumento dramático en la eficiencia y productividad para cualquier operación. (Pérez Zurita y Oña Cruz, 2006)

Tecnologías de software a utilizar

Tecnología .Net

La versión 2.0 de Microsoft .NET Framework extiende la versión 1.1 con nuevas características, mejoras en las características existentes y ampliaciones en la documentación. Esta sección proporciona información sobre algunas de las principales adiciones y modificaciones.

- **ADO.NET**

Las nuevas características de ADO.NET incluyen la compatibilidad para tipos definidos por el usuario (UDT), operaciones asíncronas de bases de datos, tipos de datos XML, tipos de valor grandes,

aislamiento de capturas y nuevos atributos que permiten a las aplicaciones admitir varios conjuntos de resultados activos (MARS) con SQL Server 2005.

- **API de protección de datos**

La nueva API de protección de datos (DPAPI) incluye cuatro métodos que permiten a las aplicaciones cifrar contraseñas, claves, cadenas de conexiones, etc. sin llamar a la invocación de la plataforma. También pueden cifrarse bloques de memoria en equipos con Windows Server 2003 o sistemas operativos posteriores.

- **Detección de cambios en la conectividad de red**

La clase NetworkChange permite a las aplicaciones recibir una notificación cada vez que cambia la dirección de Protocolo de Internet (IP) de una interfaz de red (también conocidas como tarjetas o adaptadores de red). Una dirección de interfaz puede cambiar por varios motivos, como, por ejemplo, la desconexión de un cable de red, el desplazamiento fuera del alcance de una red de área local inalámbrica o un error de hardware. La clase **NetworkChange** ofrece esta función de notificación provocando eventos cada vez que se detecta un cambio.

- **Ping**

La clase Ping permite a las aplicaciones determinar si un equipo remoto es accesible desde la red. Esta clase, compatible con las llamadas sincrónicas y asincrónicas, proporciona una funcionalidad parecida a la de la herramienta de línea de comandos Ping.exe.

- **Características relacionadas con formularios Windows Forms**

- ✓ **Implementación de ClickOnce**

La implementación de ClickOnce permite implementar aplicaciones de actualización automática para Windows que pueden instalarse y ejecutarse tan fácilmente como las aplicaciones Web. También pueden implementarse aplicaciones de línea de comandos y de cliente de Windows. Hay nuevos comandos **Publish Project** en los menús **Build** y **Project** en Visual Studio.

- ✓ **Configuración de la aplicación**

La configuración de aplicaciones para formularios Windows Forms facilita la creación, el almacenamiento y el mantenimiento de preferencias de usuarios y aplicaciones personalizadas en el cliente. Con la configuración de formularios Windows Forms, no sólo es posible almacenar datos de las aplicaciones (por ejemplo, cadenas de conexión a bases de datos),

sino también datos específicos de los usuarios (por ejemplo, posiciones de la barra de herramientas y listas de archivos usados recientemente).

✓ **Nuevos controles de formularios Windows Forms**

El control DataGridView proporciona una forma eficaz y flexible de mostrar datos en formato de tabla. Puede utilizarse para mostrar vistas de sólo lectura de una cantidad pequeña de datos o puede ajustar su tamaño para mostrar vistas modificables de conjuntos muy grandes de datos. Los datos pueden residir en un origen de datos externo o bien agregarse directamente al control.

Los controles ToolStrip son barras de herramientas capaces de alojar menús, controles y controles de usuario en aplicaciones de Windows Forms. La clase ToolStrip y sus clases asociadas permiten crear barras de herramientas y otros elementos de la interfaz de usuario en un estilo coherente con Microsoft Windows XP, Microsoft Office y Microsoft Internet Explorer. También puede utilizar estas clases para personalizar barras de herramientas y elementos de la interfaz de usuario.

El control SplitContainer de formularios Windows Forms se puede considerar como un elemento compuesto: se trata de dos paneles separados por una barra móvil. Cuando se pasa el puntero por la barra, el cursor cambia para mostrar que ésta es móvil. Este control puede utilizarse para crear interfaces de usuario complejas y de tamaño variable, muy eficaces para mostrar y explorar datos.

El control ListView ya admite tres funciones de Windows XP y la familia de Windows Server 2003: la vista en mosaico, la agrupación y la recolocación de elementos mediante arrastrar y colocar.

Los controles ListView, TreeView y ToolTip ya admiten la funcionalidad de dibujo de propietarios parecida a la compatibilidad de los controles ComboBox, ListBox, MenuItem y TabControl de la versión anterior.

El control WebBrowser permite alojar páginas Web en las aplicaciones de Windows Forms. WebBrowser puede utilizarse para proporcionar a una aplicación funciones de ayuda integrada basada en HTML o de exploración Web en Internet. Además, WebBrowser permite convertir aplicaciones Web ya existentes en aplicaciones clientes capaces de integrar a la perfección

código HTML dinámico (DHTML) con las funciones de la interfaz de usuario de los formularios Windows Forms.

El control FlowLayoutPanel organiza su contenido en una dirección de flujo horizontal o vertical. El TableLayoutPanel, por su parte, lo organiza en una cuadrícula. Como este diseño se realiza tanto en tiempo de diseño como de ejecución, puede cambiar dinámicamente, con los cambios en el entorno de la aplicación.

✓ **Funciones relacionadas con XML**

.NET Framework 2.0 ofrece muchas mejoras, entre las que se incluye un nuevo procesador de Transformación XSL (XSLT); la compatibilidad de tipos en las clases XmlReader; XmlWriter y XPathNavigator; y nuevas capacidades de edición en la clase XPathNavigator. Además, hay un nuevo modelo para crear objetos XmlReader y XmlWriter, así como muchas mejoras de rendimiento. (Microsoft, 2007b)

Lenguaje C#

C# es un lenguaje orientado a objetos simple, elegante y con seguridad en el tratamiento de tipos, que permite a los programadores de aplicaciones empresariales crear una gran variedad de aplicaciones.

C# también proporciona la capacidad de generar componentes de sistema duraderos en virtud de las siguientes características:

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

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

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

SQL

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

Es de eso de lo que trata el SQL (Structured Query Language), que no es más que un lenguaje estándar de comunicación con bases de datos, normalizado, que permite trabajar con cualquier tipo de lenguaje en combinación con cualquier tipo de base de datos (MS Access, SQL Server, MySQL, PostgreSQL, SAP Database, Firebird, Oracle *etc*).

El hecho de que sea estándar no quiere decir que sea idéntico para cada base de datos. En efecto, determinadas bases de datos implementan funciones específicas. (Pérez Zurita y Oña Cruz, 2006)

El hecho que SQL sea un estándar no quiere decir que en todas las bases de datos se haga de forma idéntica, sino que cada gestor implementa funciones específicas, de esa forma tenemos PL/SQL que es una extensión de SQL para el trabajo con Oracle, PgSQL es otra extensión pero específicamente para interactuar con bases de datos en PostgreSQL.

Fundamentación del Gestor de Bases de Datos utilizado

Los principales objetivos de un Gestor de Base de Datos (SGBD) son evitar la redundancia, eliminando así la inconsistencia, y mejorar los mecanismos de seguridad de los datos y la privacidad.

Podemos distinguir cuatro tipos de contextos para usar mecanismos de seguridad: contra accesos indebidos a los datos, contra accesos no autorizados a la BD, contra destrucción causada por el entorno (fuego, inundación, robo), y contra fallos del propio sistema (fallos del hardware, del software, etc.). (Pérez Zurita y Oña Cruz, 2006)

Debido a la gran cantidad de información que se almacenará en el sistema propuesto, se ha decidido utilizar PostgreSQL como gestor de base de datos, por ser una aplicación poderosa, robusta, que garantiza gran seguridad, así como una gran escalabilidad y confiabilidad, fundamentales para el éxito de bases de datos de gran tamaño. Este gestor fue diseñado para trabajar con altos volúmenes de información, escala muy bien sobre distribuciones de Linux. Además realiza comprobaciones de integridad referencial, permite procedimientos en el servidor y posee ciertas características orientada a objetos.

Fundamentación de la metodología

Racional Unified Process (RUP)

Se necesita un método común, un proceso que:

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

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

El Proceso Unificado utiliza el *Lenguaje Unificado de Modelado* (Unified Modeling Language, UML) para preparar todos los esquemas de un sistema software , De hecho, UML, es una parte esencial del Proceso Unificado – sus desarrollos fueron paralelos.(Pérez Zurita, Oña Cruz, 2006)

En síntesis RUP queda definido por sus tres características principales: dirigido por casos de uso, centrado en la arquitectura, iterativo e incremental.

UML (Unified Model Language)

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

Divide cada proyecto en un número de diagramas que representan las diferentes vistas del proyecto. Estos diagramas juntos son los que representa la arquitectura del proyecto. Permite modelar, construir y documentar los elementos que forman un sistema software orientado a objetos. Se ha convertido en el estándar de la industria, debido a que ha sido impulsado por los autores de los tres métodos más usados de orientación a objetos: Grady Booch, Ivar Jacobson y Jim Rumbaugh. Tiene como objetivo brindar un material de apoyo que le permita al lector poder definir diagramas propios como también entender diagramas ya existentes.

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

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

Herramientas utilizadas

Para la confección de esta aplicación se utilizó Microsoft Visual Studio 2005, por ser un entorno de desarrollo integrado muy poderoso en el trabajo con el lenguaje C Sharp. Para el modelado de la aplicación se utilizó Visual Paradigm que se integra a varios IDE (Entornos de Desarrollos Integrados) de Java y a Visual Studio, por lo que siguiendo las políticas de la Dirección de Informatización de utilizar herramientas libres decidió utilizar Visual Paradigm para su migración a Java.

Visual Paradigm UML

Se mencionará a continuación varias características que posee esta aplicación:

- Soporte para UML completo, permite realizar disímiles diagramas para modelar un software.
- Administración de requerimientos.
- Modelado del proceso de negocio.
- Modelado de la base de datos.

- Mapeo Objeto – Relacional.
- Amplia posibilidad de variación de estilos y formatos.
- Genera documentación en variados formatos y reportes HTML.
- Facilidades de impresión.
- Integra a varios IDE.
- Ingeniería inversa a C++, C#, VB.NET, PHP, Java, XML, Corba IDL, Ada9x.
- Generación e código para Java, C++, Delphi, Perl, XSD, C#, VB.NET, Python, Ada, IDL, Flash Action Script, PHP 5.0.
- Generación de código para máquinas de estado en Java, C#, VB.NET.
- Editor de formas.
- Posibilidad de presentación automática.
- Interoperabilidad.
- Interfaz de usuario intuitiva.
- Actualización automática.

Visual Studio 2005

Lenguaje y compilador

El lenguaje de C# admite ahora tipos genéricos, iteradores y tipos parciales. La versión última del compilador de C# también incluye nuevas características y opciones.

Editor de código

El editor de código contiene las siguientes nuevas características para Visual C# 2005.

Fragmentos de código

Los fragmentos de código aceleran la entrada de construcciones de código común mediante una plantilla que se puede cumplimentar. Los fragmentos de código se almacenan como archivos XML que se modifican y personalizan con facilidad.

- Fragmentos de código (C#)
- Utilizar fragmentos de código (C#)

- Utilizar fragmentos de código envolventes

Refactorización

Las herramientas de refactorización reestructuran automáticamente el código fuente, por ejemplo, promocionando variables locales a parámetros o convirtiendo un bloque de código en un método.

- Promocionar una variable local a parámetro
- Extraer método
- Encapsular campo
- Extraer interfaz
- Cambiar nombre
- Quitar parámetros
- Reordenar parámetros

Entorno de desarrollo

El entorno de desarrollo incluye las siguientes mejoras para Visual C# 2005.

IntelliSense

IntelliSense se ha optimizado con estas nuevas características:

- La lista de finalización de la Lista de miembros aparece automáticamente cuando el cursor retrocede a un operador de ámbito que precede a un objeto o cuando deshace la acción de finalización.
- Cuando se escribe código de control de errores, la Lista de miembros ayuda a descubrir la excepción que se debe detectar filtrando los miembros irrelevantes de la lista de finalización en una cláusula catch.
- Cuando sea preciso insertar código normalizado, Generación automática de código permite pedir a IntelliSense que inserte el código.
- IntelliSense está disponible cuando se crean aplicaciones Web.

Diseñador de clases

El Diseñador de clases es un nuevo editor que muestra clases y tipos gráficamente, y permite agregar o modificar métodos. También se pueden utilizar las herramientas de refactorización de la ventana

Diseñador de clases.

Herramienta de prueba de objetos

La Herramienta de prueba de objetos está diseñada para llevar a cabo pruebas sencillas en el nivel de objetos. Permite crear una instancia de un objeto y llamar a sus métodos..

Implementación ClickOnce

La implementación de ClickOnce permite publicar aplicaciones para Windows en un servidor Web o recurso compartido de red con el fin de simplificar la instalación.

Soporte de herramientas para ensamblados con nombre seguro

Se ha rediseñado el cuadro de diálogo **Propiedades del proyecto**, que ahora incluye compatibilidad para firmar ensamblados.

Documentación y especificaciones del lenguaje

La documentación de C# para dar una información completa sobre cuestiones de uso simples y avanzadas que los desarrolladores se planteen durante la creación de aplicaciones en C#. (Microsoft, 2007a)

Conclusiones

En este capítulo se dio a conocer el estado del arte del tema tratado, además se mencionaron algunas de las tendencias, técnicas, tecnologías, metodologías y software usados en la actualidad, quedando expuestas las utilizadas en el diseño e implementación de la aplicación.

CAPÍTULO 2. Características del sistema

Introducción

En el presente capítulo se describen las características del sistema, o sea, el objeto de estudio, la propuesta del sistema, el modelo del negocio con la descripción general de los procesos de negocio propuestos, los actores y trabajadores que intervienen y los casos de uso del negocio con sus respectivos diagramas y el modelo de objetos. Además se especifican los requisitos principales del software mediante los requerimientos funcionales y no funcionales.

Objeto de estudio

El objeto de estudio es la Seguridad y Protección en la UCI, específicamente lo referido a la acreditación de personas.

Problema y situación problemática

Objetivo estratégico de la organización

La Universidad de las Ciencias Informáticas, fue creada en la antigua base de radio escucha Lourdes en el año 2002 siendo la primera universidad nacida al calor de la Batalla de Ideas. Su principal desafío es el de formar miles de jóvenes en la rama de la informática. En este centro se garantiza una elevada preparación docente, además de recibir preparación desde el punto de vista profesional al estar vinculados directamente a proyectos productivos. Los planes de estudios tienen un gran nivel de flexibilidad, incluso dentro de un mismo año académico, teniendo en cuenta las necesidades existentes en dicho momento o las que tenga un determinado proyecto productivo. Todo esto permitirá en gran medida desarrollar la industria de software nacional, así como llevar a cabo la informatización de la sociedad cubana.

Flujo actual de los procesos involucrados en el campo de acción

La acreditación de las personas en la UCI ocurre de la siguiente manera: A las personas que han sido aprobadas, se les introducen sus datos en las bases de datos de la UCI, y posteriormente estas deben

dirigirse a la Oficina de Acreditación de la Dirección de Seguridad y Protección, donde solicitan su credencial, una vez impresa la credencial, se le entrega a la persona la cual podrá usarla a partir de ese momento para acceder a los diferentes servicios del centro para los cuales se requiere dicha identificación. Si existe algún problema con la credencial, las personas nuevamente se dirigen a dicha oficina y solicitan verificar el estado de su credencial, donde se les informa qué tipo de problemas tiene la misma. Cuando se trata de pérdida de credencial, es preciso pagarla y entonces es reportada como perdida y es creada una nueva. Para el caso de las bajas, por cualquier motivo, la credencial es recogida en la misma oficina y se reporta como devuelta.

Análisis crítico de cómo se ejecutan actualmente esos procesos, las causas que originan la situación problemática y las consecuencias

En la actualidad este proceso está demasiado ligado al entorno UCI, lo cual implica que al más mínimo cambio haya que hacer arreglos no previstos en el código, y debido a la urgencia con la que hay que realizarlos, muchas veces los mismos no favorecen el buen rendimiento del sistema. A esto se suma que cuando se desea crear credenciales que no se concibieron desde un principio no es posible hacerlo utilizando el sistema que ha funcionado hasta la actualidad. El proceso de acreditación de personas es muy común en muchísimas entidades y de la forma que está funcionando hoy en la UCI no es capaz de reutilizarse para otras empresas.

Objeto de automatización

Serán objeto de automatización los procesos que intervienen en la acreditación de la persona, lo cual incluye la impresión de la credencial, la creación de nuevas credenciales y almacenar el historial de credenciales de una persona dada.

En nuestra universidad existen varios sistemas que se relacionan con el campo de acción, como son: el Sistema de Control de Acceso, el cual controla las entradas y salidas a la UCI y el Sistema de Control de Acceso a los Comedores, que controla el acceso diario al comedor en cada una de las sesiones,

desayuno, almuerzo y comida. Vale señalar que estos sistemas pueden tener este control debido al código de barra que posee cada credencial.

Información que se maneja

La información que se maneja es la que aparece en las credenciales, en las cuales se muestra, en dependencia, datos tales como, el nombre de la persona, foto, nombre del área a la que pertenece, cargo, número de solapín y el código de barras.

Propuesta de Sistema

La propuesta de solución de este trabajo es realizar un sistema que sea capaz de gestionar grupos de credenciales, los cuales van a estar asociados a un proveedor de datos, el que permitirá adicionar credenciales a dicho grupo, con esto se logra que el usuario del sistema pueda seleccionar un grupo y que se muestren las credenciales que anteriormente adicionó al mismo pudiendo filtrar por cada uno de los campos que componen la credencial, es válido aclarar que el criterio para adicionar una credencial a un grupo se deja a consideración de la persona encargada de trabajar con la aplicación, sólo teniendo en cuenta que todas las credenciales que pertenecen a un grupo deben tener los mismos campos de información. También se podrá seleccionar una credencial mostrándose una vista previa de cómo debe quedar esta, para imprimirla. Esta solución tiene como una de sus principales ventajas que puede reutilizarse el sistema en otros entornos, teniendo en cuenta que sólo se tendría que desarrollar el proveedor que es el que conoce el modelo de datos de esa organización y además las reglas del negocio del mismo. Luego este proveedor al asignársele a un grupo determinado se cargaría dinámicamente y podría ser utilizado.

Las otras soluciones que se han encontrado, con gran prestigio a nivel internacional, no permiten manejar el proceso de la impresión de credenciales en entornos complejos, como es el de la UCI, además de ser muy caros, estos sistemas sólo funcionan obteniendo la información de una base de datos que se le indique, pero cuando la información se encuentre en diferentes bases de datos o se necesite utilizar servicios Web y actualizar algún estado, estos sistemas no son capaces de cumplir esta tarea, ellos se

centran más en el diseño de las credenciales, o sea, dónde poner la información, determinada imagen o un texto. Por eso es que se decidió seguir adelante con la propuesta de solución.

Modelo de negocio

Para satisfacer las necesidades del cliente y lograr que la aplicación responda a los requerimientos, es preciso conocer y comprender el negocio actual del entorno de la Universidad.

Conceptos asociados al modelo del negocio:

Persona: Todo ciudadano perteneciente vinculado a la UCI: estudiantes, profesores, tercerizados, trabajadores de la UCI y otros que prestan servicios en ella.

Credencial: Documento para identificar a las personas pertenecientes a la UCI, el mismo posee varios datos como son: nombre completo, número de solapín, área en que trabaja, código de barra que lo identifica, cargo en caso que sea dirigente y si la persona vive o no en el centro.

Reglas del negocio que se deben tener en cuenta para modelar o automatizar todo el proceso de desarrollo:

- ✓ Las personas no deben tener más de una credencial que las identifique que sea válida.
- ✓ En caso de que se pierda la credencial es necesario pagar la misma.
- ✓ Para poder tener una credencial las personas tienen que estar registradas como que pertenecen a la UCI.
- ✓ Las credenciales tienen que tener estado Pendiente cuando se crea, luego pasa a estado Impresa y de ahí puede pasar a los estados Devuelta si se entrega la credencial en la oficina de Seguridad y Protección o Perdida en caso de que se extravíe la misma.

Descripción de actores y trabajadores

Actores del negocio	Justificación
Persona	Persona que trabaja o estudia en la universidad, tiene que tener un solapín con datos que los identifiquen, pueden vivir dentro o fuera de la UCI y necesita autenticarse en el comedor.

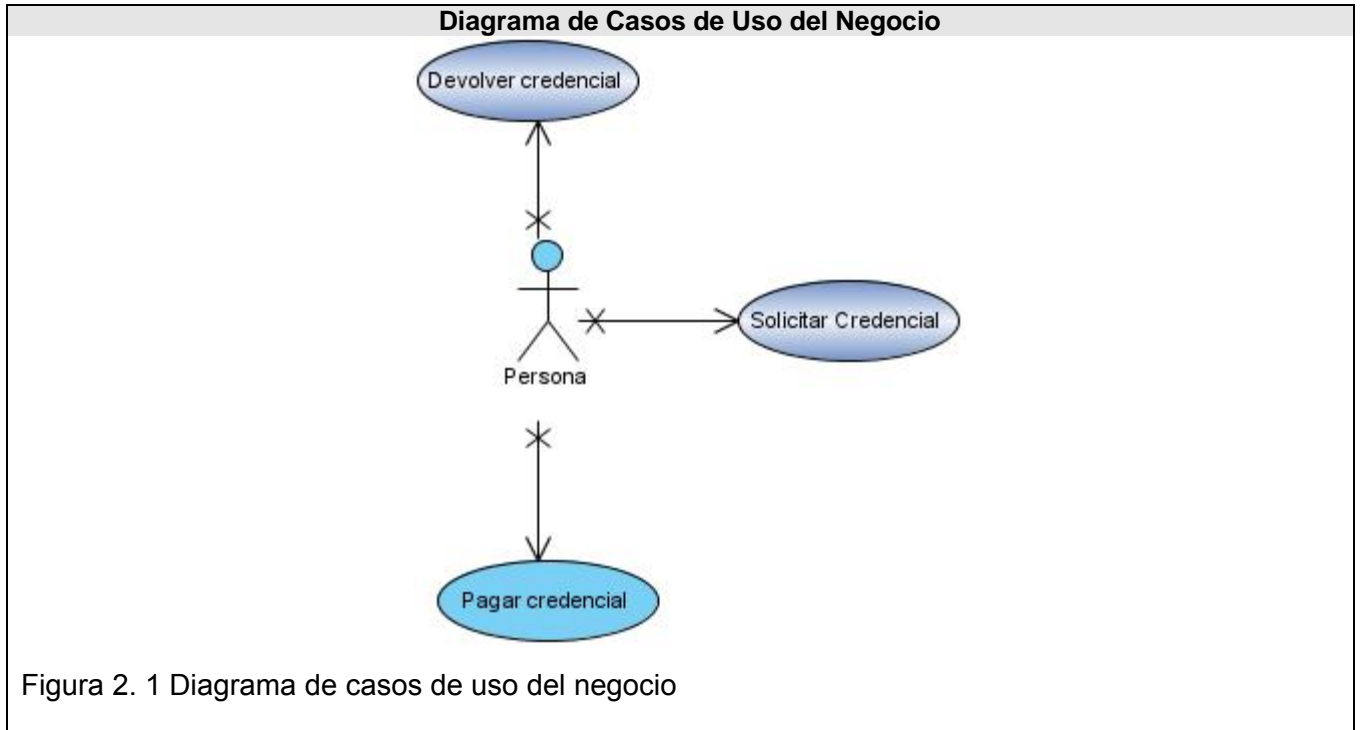
Tabla 1 Descripción de actores del negocio

Trabajadores del negocio	Justificación
Especialista de Seguridad y Protección	Especialista de la Dirección de Seguridad y Protección encargado de controlar todo lo relacionado con las credenciales. Esta es la persona que las crea, imprime, recoge, reporta si está perdida.

Tabla 2 Descripción de trabajadores del negocio

Representación gráfica del diagrama de casos de uso del negocio

El diagrama de casos de uso (CU) del negocio posee gran importancia pues a través de él se observa la interacción entre los actores y los casos de uso del negocio. Este modelo permite tener una perspectiva de cómo funciona el sistema desde el punto de vista de su uso y describe como cada caso de uso es llevado a cabo por los trabajadores que utilizan entidades.



Caso de uso del negocio: Devolver credencial	
Actores del negocio: Persona	
Propósito: Registrar que la persona devolvió la credencial, en caso de poseer una activa.	
Resumen: El actor solicita devolver una credencial, por lo que se busca si tiene alguna credencial activa, la devuelve y se registra que está devuelta.	
Curso normal de los eventos	
Acción del actor	Respuesta del negocio
1- El actor solicita devolver la credencial	1.1- Busca a ver si la persona posee una credencial activa. 1.2- Si la posee le solicita la credencial
2- El actor hace acto de entrega	2.1- Se cambia el estado de la credencial a devuelta.
Flujo Alternativo(1.1)	
	1.1- Si no posee una credencial activa se le notifica que no es válida la devolución.
Prioridad: Crítico	
Mejoras:	

Tabla 3 Descripción del caso de uso del negocio “Devolver Credencial”

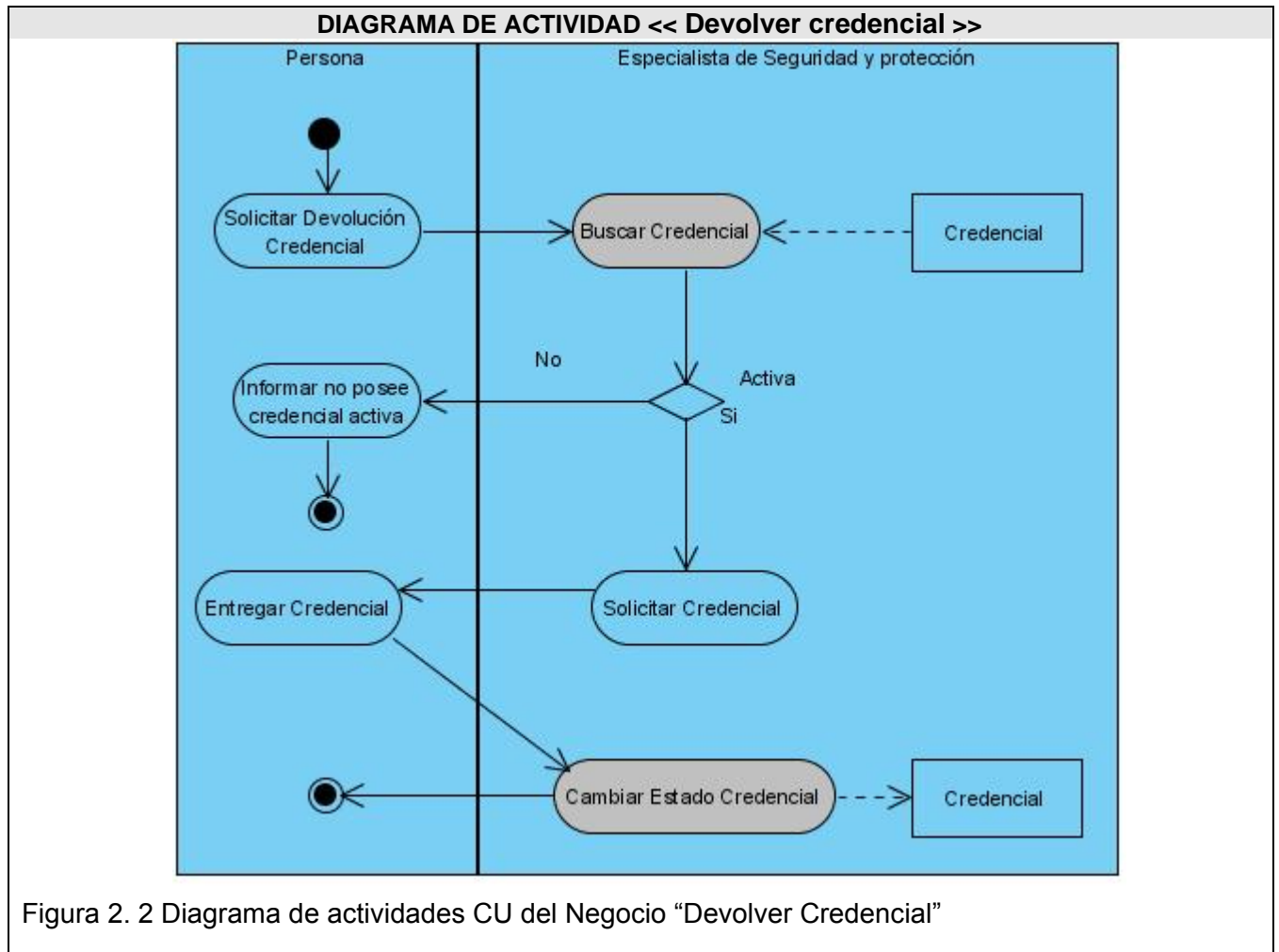


Figura 2. 2 Diagrama de actividades CU del Negocio “Devolver Credencial”

Caso de uso del negocio: Solicitar credencial	
Actores del negocio: Persona	
Propósito: Crearle una credencial a una persona y entregársela para su uso.	
Resumen: El caso de uso se inicia cuando el actor solicita una nueva credencial, se verifica que no tenga otra credencial activa y se procede a crearle la credencial, imprimirla y entregársela.	
Curso normal de los eventos	
Acción del actor	Respuesta del negocio
1- El actor solicita una nueva credencial.	1.1- Se le piden sus datos
2- Informa datos que lo identifiquen	2.1- Se verifica si la persona tiene alguna credencial activa.

	2.2- Si no posee alguna credencial activa se le crea una nueva credencial. 2.3- Se imprime la credencial. 2.4- Solicita la firma de entrega.
3- Firma la constancia de que recibió la credencial.	3.1- Se le entrega la credencial.
Flujo Alternativo(2.1)	
	2.1- Si posee una credencial activa se le notifica que no puede tener dos credenciales activas a la vez.
Prioridad: Crítico	
Mejoras:	
Otras secciones:	

Tabla 4 Descripción del caso de uso del negocio “Solicitar Credencial”

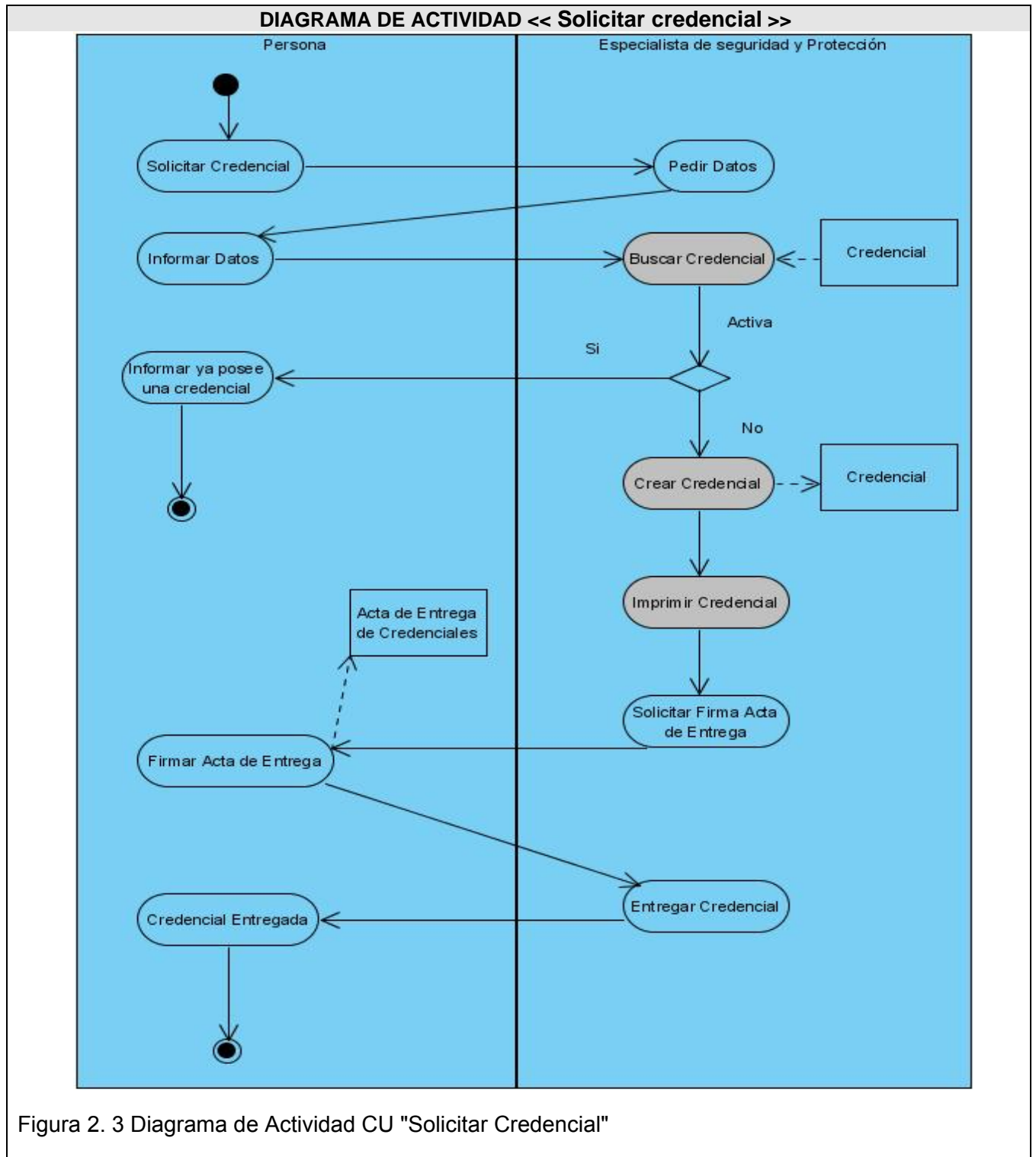


Figura 2. 3 Diagrama de Actividad CU "Solicitar Credencial"

Caso de uso del negocio: Pagar credencial	
Actores del negocio: Persona	
Propósito: Pagar la credencial en caso de pérdida.	
Resumen: El caso de uso lo inicia la persona cuando solicita pagar una credencial, se busca si tiene alguna credencial activa y se procede a cobrar la credencial.	
Curso normal de los eventos	
Acción del actor	Respuesta del negocio
1- El actor solicita pagar la credencial	1.1- Se busca si la persona posee una credencial activa. 1.2- Si posee alguna credencial activa se procede a pedirle el importe de esa credencial.
2- El actor abona la cantidad estipulada.	2.1- Se llena el modelo de pago 2.2- Se solicita la firma del actor.
3- El actor procede a firmar el modelo.	3.1 Se actualiza el estado de la credencial.
Flujo Alternativo(1.2)	
	1.2- Sino posee ninguna credencial activa se informa que no tiene credencial que pagar.
Prioridad: Opcional	
Mejoras:	
Otras secciones:	

Tabla 5 Descripción CU Pagar Credencial

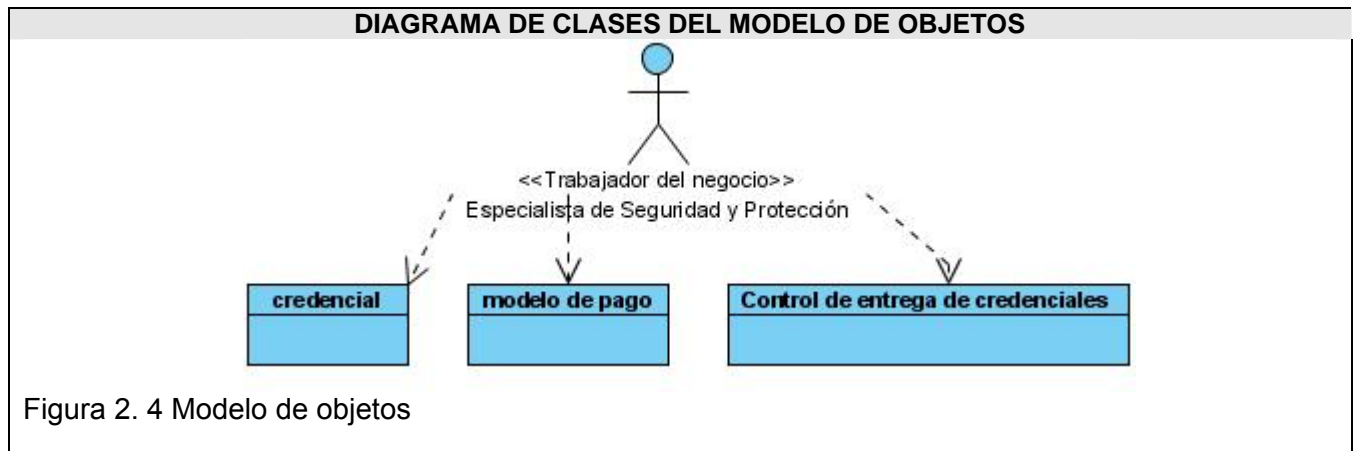


Figura 2. 4 Modelo de objetos

Especificación de los requisitos de software

Este sistema depende de otros tales como Akademos (para los estudiantes) y el Sistema de Recursos Humanos (para los trabajadores); y de este depende a su vez, el Sistema de Control de Acceso y el Sistema de Control de Acceso a Comedores.

Requerimientos Funcionales

1. Crear Grupos.
 - 1.1. Asignar un nombre al grupo.
 - 1.2. Escoger un tipo de plantilla
 - 1.3. Seleccionar un proveedor de datos.
 - 1.4. Seleccionar de qué grupo es hijo.
2. Adicionar credenciales a un grupo
 - 2.1. Buscar personas teniendo en cuenta las posibilidades que brinde el proveedor de datos.
 - 2.2. Seleccionar las que deseamos añadir al grupo.
3. Mostrar listado de credenciales de un grupo
4. Imprimir credencial.
5. Mostrar vista previa de la credencial.
6. Actualizar el estado de la credencial.
 - 6.1. Reportar como perdida.
 - 6.2. Reportar devuelta.
7. Eliminar grupo.
8. Modificar nombre de grupo.
9. Filtrar credenciales.

Requerimientos no Funcionales

- Apariencia de o interfaz externa: este será un sistema muy sencillo y básico de usar, permitiendo que el tiempo de aprendizaje con el sistema sea pequeño.

- Usabilidad: El sistema podrá ser utilizado por personas que posean conocimientos básicos de computación.
- Rendimiento: El sistema operará con grande volúmenes de información, debido a que él almacena todos los datos de cada una de las credenciales a imprimir, por lo que se necesita tener un gran rendimiento, con vistas a que no existan grandes demoras para imprimir.
- Soporte: Se requiere grande volúmenes de datos y alta capacidad de procesamiento. Tiempo de respuesta pequeño. Plataforma Framework 2.0.
- Portabilidad: Este sistema funcionará en cualquier equipo con sistema operativo Microsoft Windows 98 o posterior y posteriormente se migrará a Java, logrando que sea completamente multiplataforma.
- Seguridad: Verificación ante acciones irreversibles (eliminar).
- Legales: Debe cumplir con el plan de seguridad y protección del centro.
- Ayuda y documentación: Se impartirán pequeños cursos con el fin de enseñar a usar la aplicación, además se está trabajando en el desarrollo de una ayuda para la utilización del sistema.
- Software: Se requiere:
 - Plataforma Net Framework 2.0 para el funcionamiento del sistema.
 - PostgreSQL 8.01.
- Hardware:
 - Para la plataforma NET Framework 2.0:
 - 280Mb de espacio en el disco duro
 - Procesador 133 MHz o superior.
 - 128 MB de memoria RAM.

Modelo de Casos de Uso del Sistema

Los casos de uso del sistema serán en definitiva los encargados de guiar todo el proceso de desarrollo del sistema a implementar. Es importante determinar en esta etapa los actores del sistema así como los casos de uso de los cuales se benefician.

Definición de los actores del sistema a automatizar

Actores	Justificación
Especialista de Seguridad y Protección	Especialista de la Dirección de Seguridad y Protección encargado de controlar todo lo relacionado con las credenciales. Esta es la persona que las crea, imprime, recoge, reporta si está perdida.

Tabla 6 Definición de actores del sistema

Listado de casos de uso

Se presenta un listado de los casos de uso del sistema con una breve información de quien es el actor que lo inicializa, el nombre del caso de uso y una breve descripción del mismo.

CU-1	Modificar Nombre Grupos
Actor	Especialista de Seguridad y Protección
Descripción	Este caso de uso ocurre cuando el actor selecciona un grupo y le cambia el nombre.
Referencia	Requerimiento funcional 8.

Tabla 7 Caso de Uso Modificar “Nombre Grupo”

CU-2	Crear Grupos
Actor	Especialista de Seguridad y Protección
Descripción	El actor del sistema crea un grupo, para lo cual debe seleccionar el proveedor de datos de ese grupo, la plantilla a utilizar y además del nombre, si es hijo de algún otro grupo.
Referencia	Requerimiento funcional 1

Tabla 8 Caso de Uso “Crear Grupos”

CU-3	Visualizar Credencial
Actor	Especialista de Seguridad y Protección

Descripción	Ocurre cuando el actor selecciona una credencial para ver cómo quedará una vez impresa.
Referencia	Requerimiento funcional 5

Tabla 9 Caso de Uso "Visualizar Credencial"

CU-4	Eliminar Grupos
Actor	Especialista de Seguridad y Protección
Descripción	El actor selecciona un grupo y lo borra eliminando consigo todas las credenciales que contenía el mismo.
Referencia	Requerimiento funcional 7

Tabla 10 Caso de Uso "Eliminar Grupos"

CU-5	Imprimir Credencial
Actor	Especialista de Seguridad y Protección
Descripción	El actor selecciona un conjunto de credenciales de un grupo determinado y las imprime.
Referencia	Requerimiento funcional 4

Tabla 11 Caso de uso "Imprimir Credencial"

CU-6	Actualizar Credencial
Actor	Especialista de Seguridad y Protección
Descripción	El actor selecciona un campo de una credencial y si este es se puede actualizar puede cambiar dicho valor.
Referencia	Requerimiento funcional 6

Tabla 12 Caso de Uso "Actualizar Credencial"

CU-7	Filtrar Credenciales
Actor	Especialista de Seguridad y Protección
Descripción	Este caso de uso ocurre cuando el actor escribe datos en los filtros que hay sobre cada uno de los campos que tienen las credenciales de un grupo, permitiendo obtener sólo las credenciales que cumplan con esos

	critérios.
Referencia	Requerimiento funcional 9

Tabla 13 Caso de Uso "Filtrar Credenciales"

CU-8	Mostrar Listado de Credenciales
Actor	Especialista de Seguridad y Protección
Descripción	Cuando el actor selecciona un grupo se muestran todas las credenciales que existen en ese grupo, en caso de poseer alguna.
Referencia	Requerimiento funcional 3

Tabla 14 Caso de Uso "Mostrar Listado de Credenciales"

CU-9	Crear Nueva Credencial
Actor	Especialista de Seguridad y Protección
Descripción	El actor accede a crear nuevas credenciales a un grupo previamente seleccionado.
Referencia	Requerimiento funcional 2

Tabla 15 Caso de Uso "Crear Nueva Credencial"

CU-10	Búsqueda Origen Datos
Actor	Especialista de Seguridad y Protección
Descripción	El actor busca las personas utilizando los medios que le brinda el buscador del proveedor seleccionando finalmente a las personas que desea.
Referencia	Requisitos funcionales 2.1, 2.2

Tabla 16 Caso de Uso "Busqueda Origen Datos"

Descripción de los Diagramas de Caso de Uso del Sistema

Nombre del caso de uso:	Modificar Nombre Grupos
Actores:	Especialista de Seguridad y Protección "(inicia)"
Propósito:	Cambiar nombre a un grupo.

Resumen: El caso de uso se inicia cuando el especialista de seguridad y protección oprime el botón para modificar el nombre de un grupo. Luego se abre una ventana para introducir el nuevo nombre, si este no se encuentra entre los que ya han sido creados y no contiene caracteres no alfanuméricos entonces se modifica el nombre del grupo, sino se eliminan estos caracteres. Si el nombre del grupo ya existe, este no se modifica.	
Referencias:	RF-8
Precondiciones:	Se haya seleccionado grupo.
Poscondiciones:	Nombre de grupo cambiado.
Curso Normal de Eventos	
Acciones del Actor	Respuesta del Sistema
1- El actor selecciona el botón modificar grupo.	1.1- Se abre un nuevo formulario.
2- El actor escribe el nuevo nombre.	2.1- Valida que el nombre no exista. 2.2- Valida que no contenga caracteres no alfanuméricos. 2.3- Modifica el nombre del grupo. 2.4- Se le envía una notificación al actor.
Curso Alterno(2.1)	
	2.1- Si ya existe el nombre introducido se le notifica al actor.
3- Puede reintentar modificar nombre.	
Curso Alterno(2.2)	
	2.2- Si contiene caracteres no alfanuméricos, estos son eliminados. 2.2.1- En caso de ser todos los caracteres no alfanuméricos, se borrarán todos y se le notificará al actor.
3- Puede reintentar modificar nombre.	
Prioridad	Media

Tabla 17 Descripción extendida CU “Modificar Nombre Grupos”

Nombre del caso de uso:	Crear Grupos
Actores:	Especialista de Seguridad y Protección “(inicia)”
Propósito:	Crear un grupo para agrupar credenciales.
Resumen: El caso de uso se inicia cuando el especialista de seguridad y protección selecciona la opción nuevo grupo, luego es posible seleccionar el proveedor de dicho grupo, la plantilla, el grupo padre, y el nombre del nuevo grupo.	
Referencias:	RF-1, 1.1, 1.2, 1.3, 1.4
Poscondiciones:	Se crea un grupo
Curso Normal de Eventos	
Acciones del Actor	Respuesta del Sistema
1- El actor selecciona el botón crear nuevo grupo.	1.1- Se abre un nuevo formulario. 1.2- Se cargan dinámicamente los valores de los

	menús desplegables de los proveedores, las plantillas y los grupos padres.
2- El actor introduce los datos correspondientes.	2.1- Valida que el nombre no exista. 2.2- Valida que no contenga caracteres no alfanuméricos. 2.3- Crea el grupo. 2.4- Se le envía una notificación al actor.
Curso Alterno (2.1)	
	2.1- Si ya existe el nombre introducido se le notifica al actor.
3- Puede reintentar introducir los datos.	
Curso Alterno (2.2)	
	2.2- Si contiene caracteres no alfanuméricos, estos son eliminados. 2.2.1- En caso de ser todos los caracteres no alfanuméricos, se borrarán todos y se le notificará al actor.
3- Puede reintentar introducir los datos.	
Prioridad	Alta

Tabla 18 Descripción extendida CU "Crear Grupos"

Nombre del caso de uso:	Visualizar Credencial
Actores:	Especialista de Seguridad y Protección "(inicia)"
Propósito:	Ver cómo quedará la credencial antes de proceder a su impresión.
Resumen:	El caso de uso se inicia cuando el especialista de seguridad y protección presiona el botón ver vista previa y se muestra una ventana con la imagen de cómo quedará la credencial.
Referencias:	RF - 5
Precondiciones:	Se tiene que haber seleccionado una credencial.
Poscondiciones:	Se muestra la vista previa.
Curso Normal de Eventos	
Acciones del Actor	Respuesta del Sistema
1- El actor presiona el botón con la opción vista previa.	1.1- Se abre un nuevo formulario. 1.2- Busca en el fichero de la plantilla correspondiente las ubicaciones de cada elemento en la plantilla. 1.3- Dibuja la credencial. 1.4- Se muestra la vista previa en la ventana.
Curso Alterno(1.2)	
	1.2- Sino se encuentra el fichero con la información de la plantilla, se muestra un mensaje de error, advirtiendo al actor.
3- El actor busca que se encuentre la plantilla adecuada en el directorio de las mismas.	

Prioridad	Media
------------------	-------

Tabla 19 Descripción extendida CU Visualizar Credencial

Nombre del caso de uso:	Eliminar Grupos
Actores:	Especialista de Seguridad y Protección "(inicia)"
Propósito:	Eliminar un grupo.
Resumen: El caso de uso se inicia cuando el especialista de seguridad y protección presiona el botón eliminar grupo, donde se mostrará una advertencia de si realmente quiere eliminarlo y finalmente se elimina.	
Referencias:	RF - 7
Precondiciones:	Exista un grupo seleccionado.
Poscondiciones:	El grupo es eliminado.
Curso Normal de Eventos	
Acciones del Actor	Respuesta del Sistema
1- El actor oprime el botón eliminar grupo.	1.1- Se muestra un mensaje de advertencia informándole al usuario que de borrar el grupo se eliminarán a su vez todas las credenciales que pertenecen a dicho grupo. 1.2- Se elimina el grupo. 1.3- Se muestra una notificación de que el grupo se eliminó correctamente.
Curso Alterno(1.1)	
	1.1- El usuario selecciona el botón cancelar.
Prioridad	Media

Tabla 20 Descripción extendida CU Eliminar Grupos

Nombre del caso de uso:	Imprimir Credencial
Actores:	Especialista de Seguridad y Protección "(inicia)"
Propósito:	Imprime la credencial.
Resumen: El caso de uso se inicia cuando el especialista de seguridad y protección presiona el botón imprimir credencial.	
Referencias:	RF - 4
Precondiciones:	Se hayan seleccionado credenciales del listado.
Poscondiciones:	Las credenciales quedan impresas.
Curso Normal de Eventos	
Acciones del Actor	Respuesta del Sistema
1- El actor oprime el botón imprimir credenciales	1.1- Se abre un nuevo formulario. 1.2- Se muestran datos de cada una de las credenciales que se imprimirán.
2- El actor oprime el botón imprimir.	2.1- Se conforman las hojas con sus respectivas credenciales en ellas. 2.2- Se imprimen las hojas de credenciales.
Curso Alterno(2)	

2- El actor oprime el botón Cancelar.	2.1- Se cierra la ventana de impresión.
Prioridad	Alta.

Tabla 21 Descripción extendida CU "Imprimir Credencial"

Nombre del caso de uso:	Actualizar Credencial
Actores:	Especialista de Seguridad y Protección "(inicia)"
Propósito:	Actualizar campos de una credencial.
Resumen:	El caso de uso se inicia cuando el especialista de seguridad y protección modifica los valores de determinado campo de una credencial.
Referencias:	RF 6, 6.1, 6.2
Precondiciones:	Se haya seleccionado una credencial del listado.
Poscondiciones:	Se actualiza el campo.
Curso Normal de Eventos	
Acciones del Actor	Respuesta del Sistema
1- El actor hace doble clic en el campo de la credencial que desea editar	1.1- Si el campo es editable se permite la escritura en el mismo.
2- El actor entra el nuevo valor	2.1- Se almacena el nuevo valor de dicho campo.
Curso Alterno(1.1)	
	1.1- Si el campo no es editable no se permite modificar su valor.
Prioridad	Media

Tabla 22 Descripción extendida CU "Actualizar credencial"

Nombre del caso de uso:	Filtrar Credenciales
Actores:	Especialista de Seguridad y Protección "(inicia)"
Propósito:	Filtrar las credenciales de un grupo
Resumen:	El caso de uso se inicia cuando el especialista de seguridad y protección oprime el botón de filtrar las credenciales, mostrándose sólo las que cumplan con la condición que se les pasa.
Referencias:	RF- 9
Precondiciones:	Que se hayan mostrado las credenciales
Poscondiciones:	Se muestran sólo las credenciales que cumplan todas las condiciones.
Curso Normal de Eventos	
Acciones del Actor	Respuesta del Sistema
1- El actor oprime el botón filtrar.	1.1- Construye una consulta con cada uno de los datos pasados. 1.2- Le pasa la consulta armada a la colección de credenciales, mostrándose sólo las que cumplan todas las condiciones.
Curso Alterno(1.1)	
	1.1- La consulta se queda vacía. 1.2- Se muestran todas las credenciales de ese grupo.

Prioridad:	Baja
-------------------	------

Tabla 23 Descripción extendida CU "Filtrar Credenciales"

Nombre del caso de uso:	Mostrar Listado de Credenciales	
Actores:	Especialista de Seguridad y Protección "(inicia)"	
Propósito:	Mostrar las credenciales de un grupo.	
Resumen:	El caso de uso se inicia cuando el especialista de seguridad y protección selecciona un grupo y se muestran todas las credenciales de ese grupo.	
Referencias:	RF- 3	
Precondiciones:	Se haya seleccionado un grupo.	
Poscondiciones:	Se muestran las credenciales.	
Curso Normal de Eventos		
Acciones del Actor	Respuesta del Sistema	
1- El actor selecciona un grupo.	1.1- Muestra todas las credenciales del grupo seleccionado. 1.2- se construyen campos de texto por cada uno de los campos visibles de las credenciales.	
2- Si el actor lo desea, filtra las credenciales usando los campos mostrados.	Punto de extensión 1 "Filtrar credenciales"	
Puntos de Extensión		
Punto de extensión 1: CU extendido "Filtrar credenciales"		
Prioridad	Alta	

Tabla 24 Descripción extendida la salida de los pases.

Nombre del caso de uso:	Crear Nueva Credencial	
Actores:	Especialista de Seguridad y Protección "(inicia)"	
Propósito:	Adicionar una nueva credencial a un grupo	
Resumen:	El caso de uso se inicia cuando el especialista de seguridad y protección oprime el botón crear nueva credencial, selecciona las personas a las que se le va a crear la credencial y se crean las credenciales y son adicionadas al grupo seleccionado.	
Referencias:	RF - 2	
Precondiciones:	Exista un grupo seleccionado.	
Poscondiciones:	Se crean las credenciales en el grupo dado.	
Curso Normal de Eventos		
Acciones del Actor	Respuesta del Sistema	
1- El actor presiona el botón adicionar credenciales.	1.1- Muestra el proveedor de datos. Ver Punto de extensión - 1 "Busqueda Origen Datos" 1.2- Se adicionan las credenciales a dicho grupo.	
Curso Alterno(1.2)		
	1.2- Los campos de las credenciales seleccionadas no coinciden con los campos de las credenciales de dicho grupo, por lo que se le	

	notifica al actor el error.
Puntos de Extensión	
Punto de extensión 1:CU incluido "Busqueda Origen Datos"	
Prioridad	Alta

Tabla 25 Descripción extendida CU "Crear Nueva Credencial"

Nombre del caso de uso:	Busqueda Origen Datos
Actores:	Especialista de Seguridad y Protección "(inicia)"
Propósito:	Buscar personas para crearle credenciales.
Resumen:	El caso de uso se inicia cuando el especialista de seguridad y protección busca a las personas para crearles credenciales.
Referencias:	RF 2.1, 2.2
Precondiciones:	Exista el proveedor del grupo seleccionado.
Poscondiciones:	Se crean las credenciales.
Curso Normal de Eventos	
Acciones del Actor	Respuesta del Sistema
1- El actor introduce el texto de la búsqueda y oprime el botón buscar.	1.1- El sistema busca las personas que cumplen las condiciones dadas. 1.2- Se muestra el resultado de la búsqueda.
2- El actor selecciona el resultado las personas que realmente el quiere crearle la credencial y hace clic en el botón OK.	2.1- Se crean las credenciales a las personas seleccionadas.
Curso Alterno(1)	
1- El actor presiona el botón búsqueda avanzada.	1.1- Se muestra un nuevo formulario con opciones de búsqueda avanzada.
2- El actor llena los campos y presiona el botón OK.	2.1- Se genera un patrón de búsqueda y se devuelve a la forma principal el cual se muestra en el campo de texto de la búsqueda.
3- Remitirse a la acción de oprimir el botón buscar del flujo normal de los eventos.(1)	
Prioridad	Alta

Tabla 26 Descripción extendida CU "Busqueda Origen Datos"

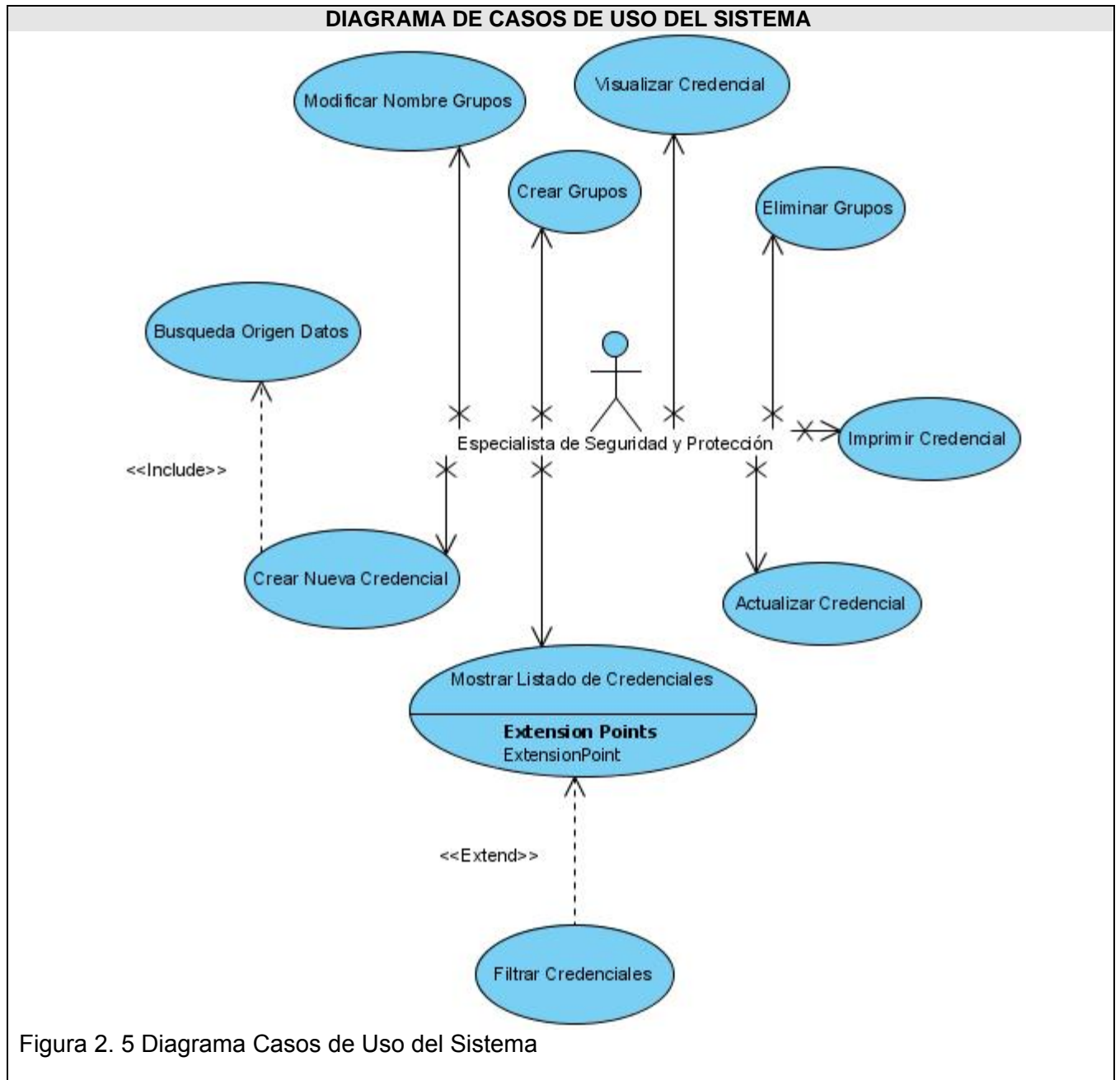


Figura 2. 5 Diagrama Casos de Uso del Sistema

Conclusiones

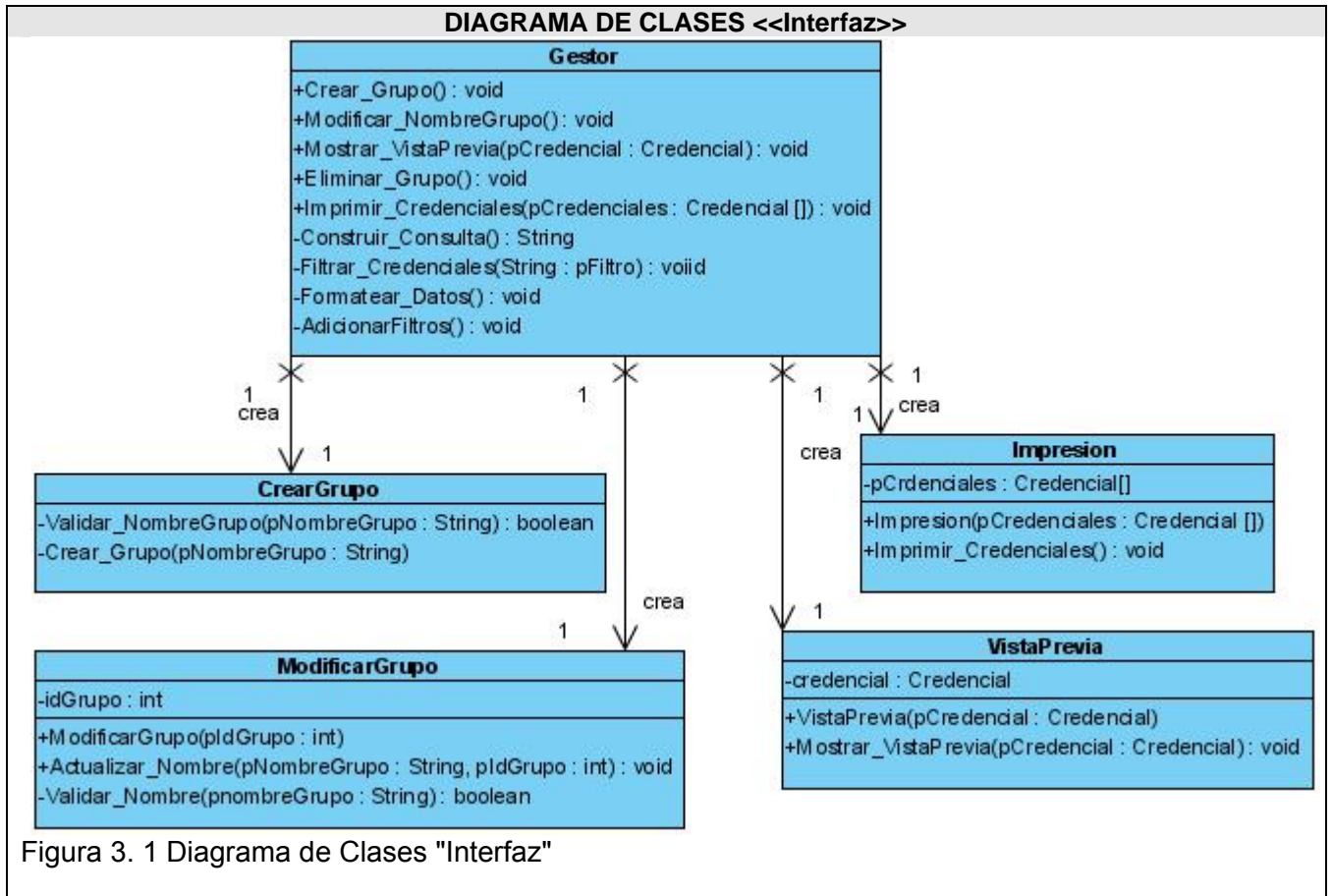
En este capítulo se reanalizaron los procesos del negocio, se identificaron los requerimientos funcionales y no funcionales del sistema, y se identificaron además los casos de uso del sistema necesarios para satisfacer los requerimientos de la propuesta de solución que se presenta. Con todo ello se logra una mayor claridad en cuanto cómo analizar e implementar dicho sistema.

CAPÍTULO 3. Diseño del sistema

Introducción

Este capítulo presenta diseño del sistema propuesto, lo cual es fundamental para el desarrollo de la aplicación. Aquí se podrá ver el diagrama de clases del diseño que posee cada componente, así como la descripción de cada una de ellas para una mejor comprensión. En aras de apreciar la interacción entre las clases se mostrarán los diagramas de secuencias en el anexo 1. Para una mayor visión de la propuesta se mostrarán también diagrama de clases persistentes y el modelo de datos.

Diagrama de Clases del Diseño



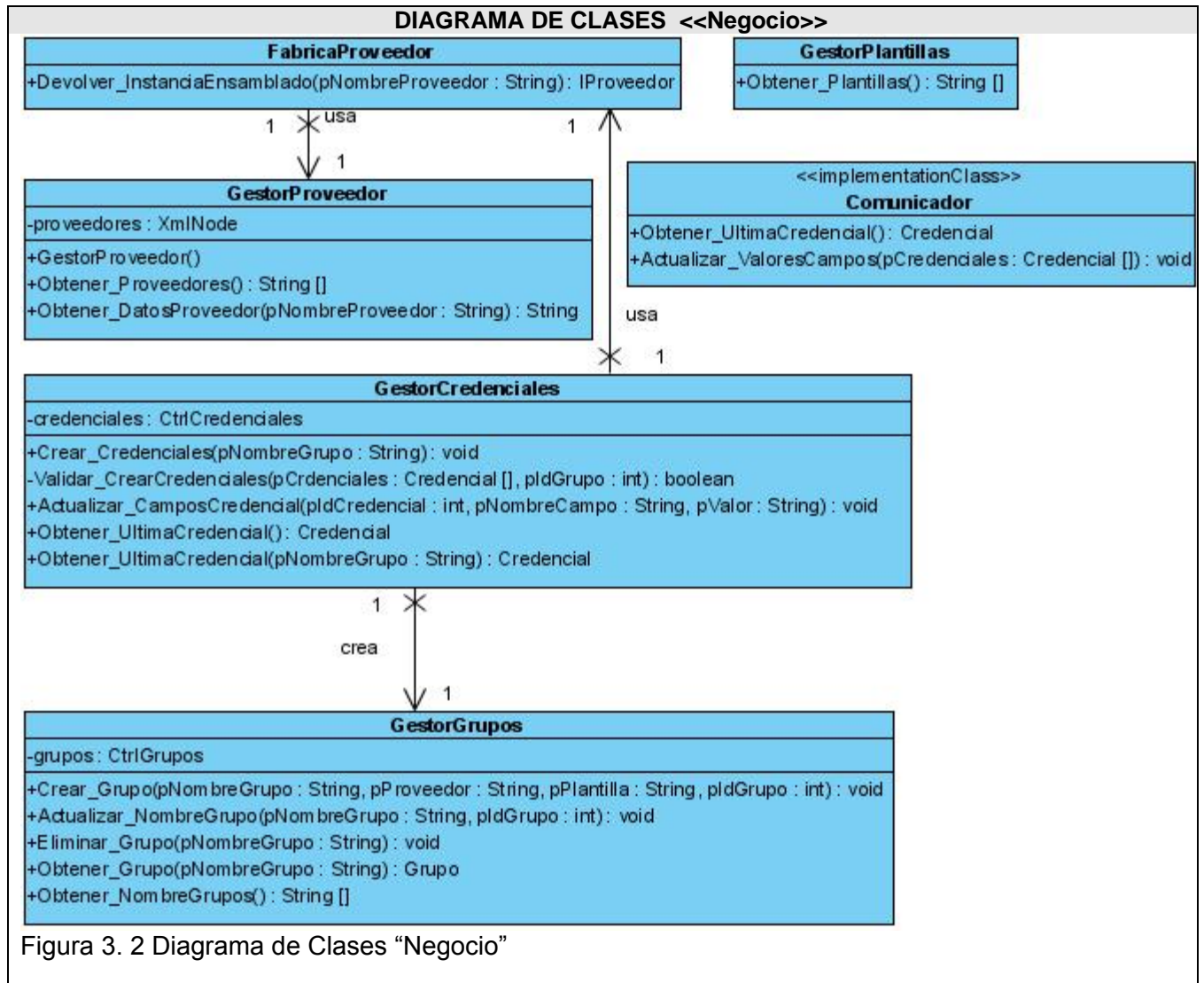
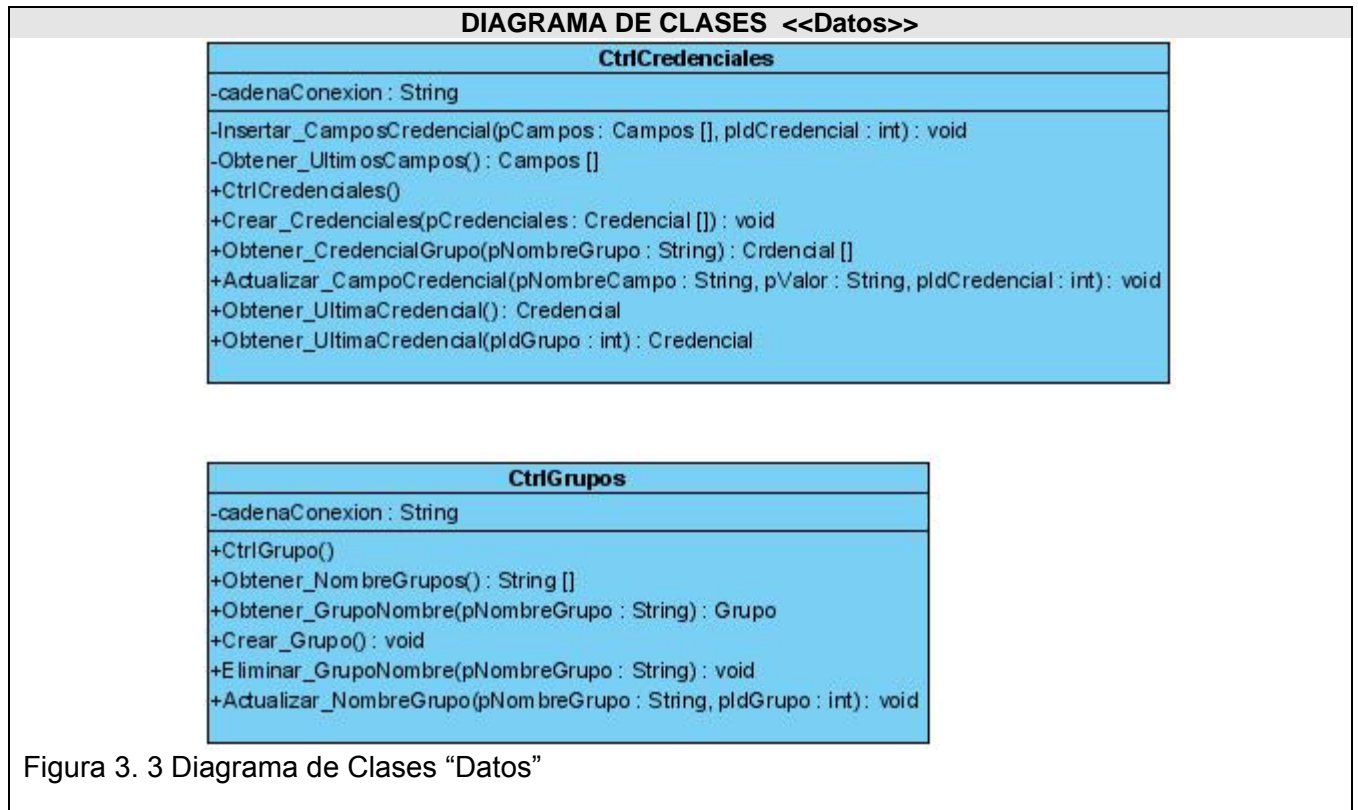


Figura 3. 2 Diagrama de Clases “Negocio”



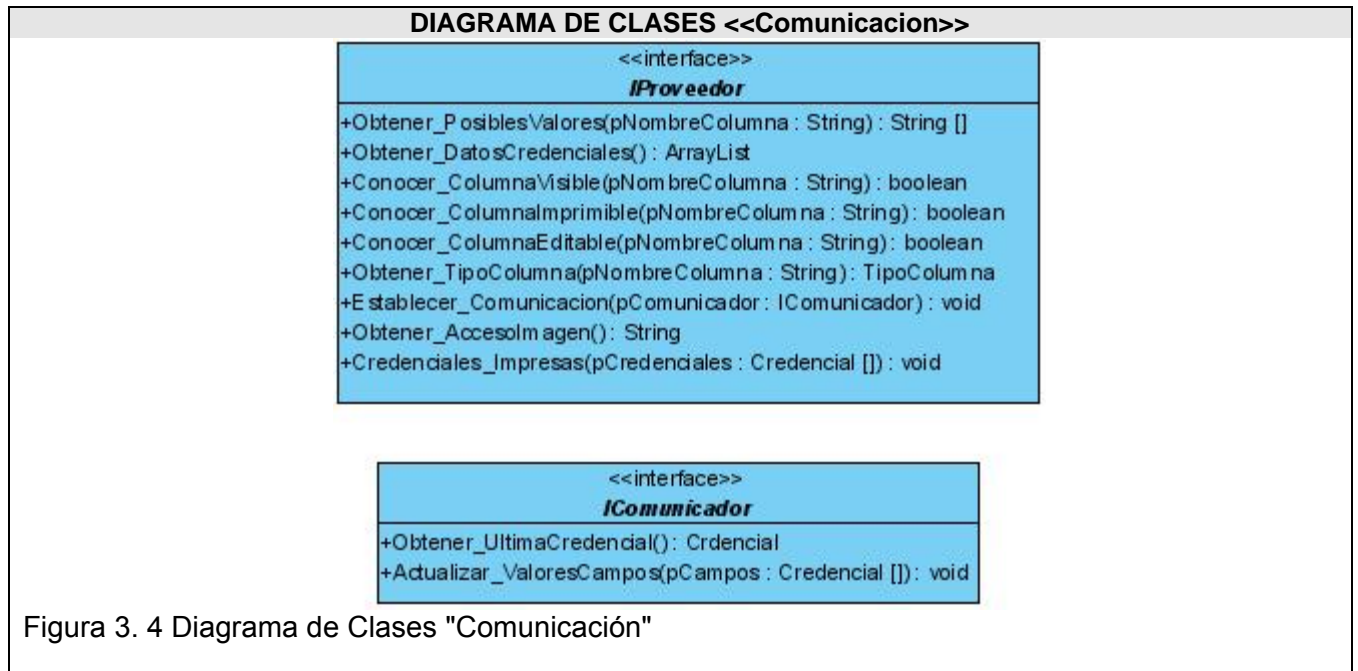


Figura 3. 4 Diagrama de Clases "Comunicación"

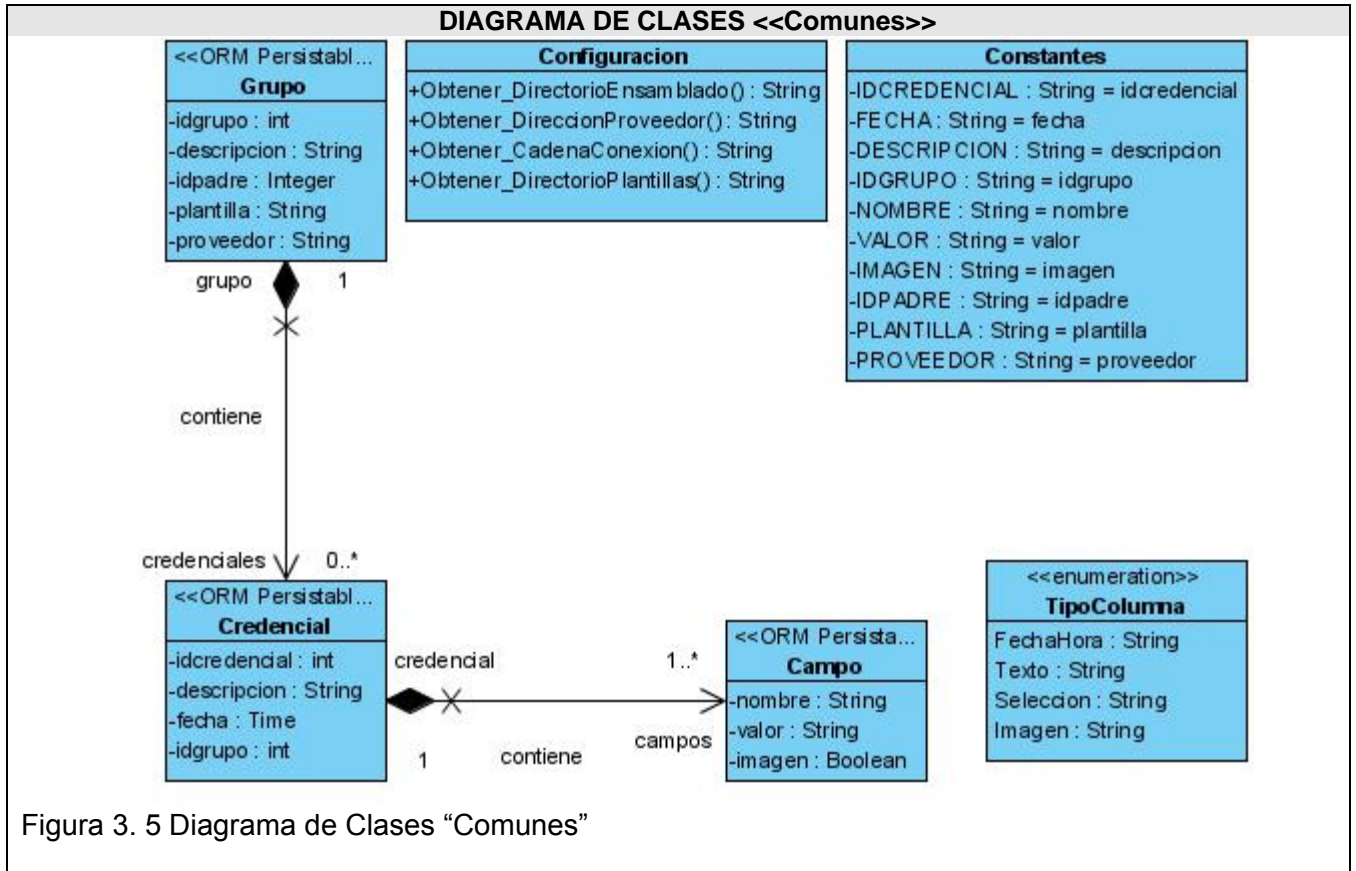


Figura 3. 5 Diagrama de Clases “Comunes”

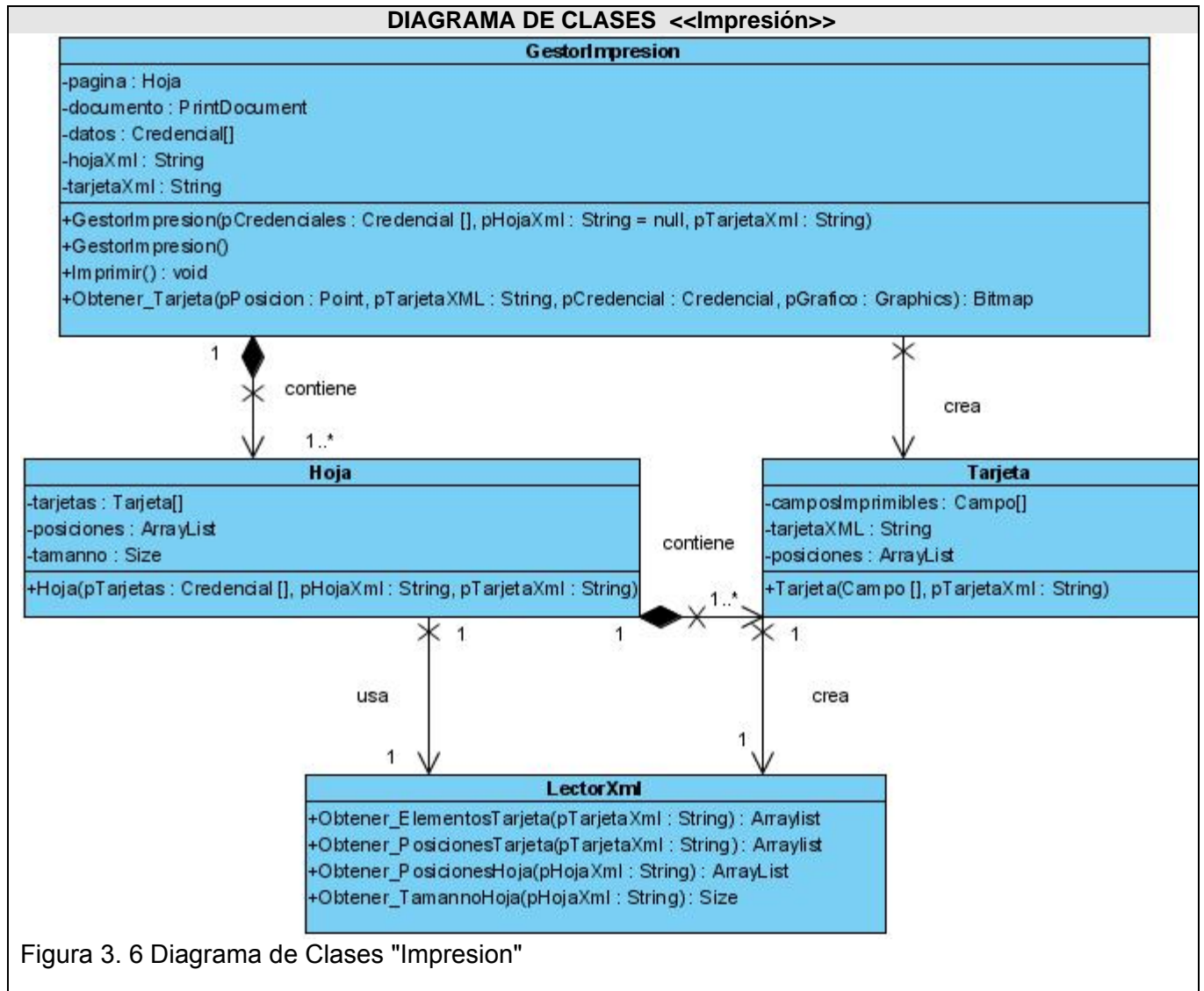
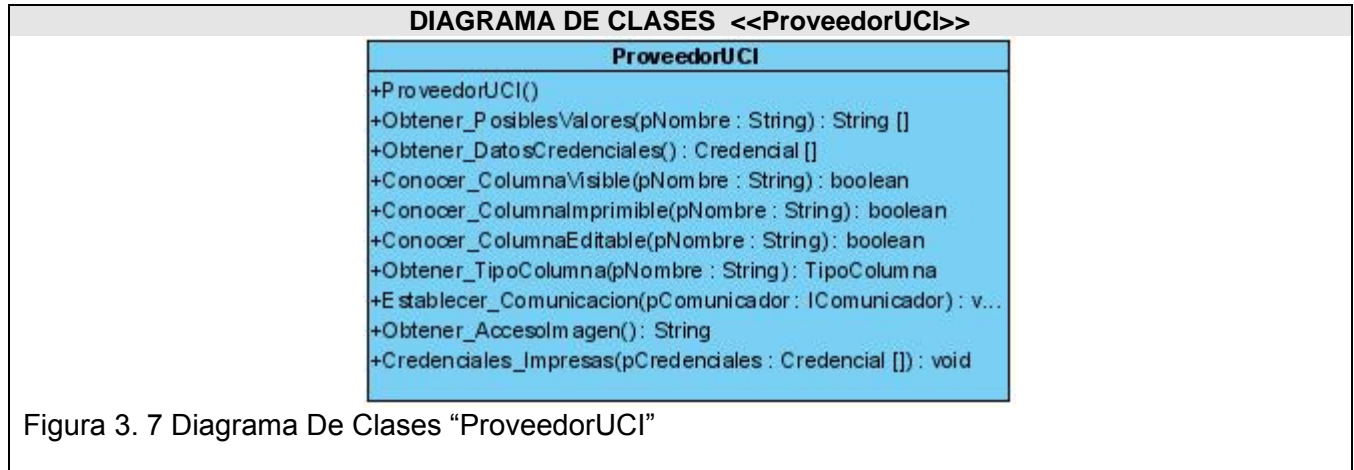


Figura 3. 6 Diagrama de Clases "Impresion"



Descripción de las Clases de Diseño

A continuación se hace una descripción de las clases del diseño, lo cual consideramos fundamental en el desarrollo del trabajo ya que de dice a los desarrolladores que responsabilidades va poseer cada clase, así como sus miembros, ya sean atributos o métodos.

Clases de la Interfaz

Nombre: Gestor	
Tipo de clase: Interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	Crear_Grupo()
Descripción:	Tiene la responsabilidad de llamar a la interfaz correspondiente para crear un nuevo grupo.
Nombre:	Modificar_NombreGrupo()
Descripción:	Tiene la responsabilidad de invocar la interfaz correspondiente para modificarle el nombre a un grupo.
Nombre:	Mostrar_VistaPrevia(pCredencial)
Descripción:	Tiene la responsabilidad de invocar la interfaz para mostrar una vista previa a la impresión de cómo quedará la credencial seleccionada.
Nombre:	Eliminar_Grupo()
Descripción:	Comienza el proceso de eliminar un grupo.
Nombre:	Imprimir_Credenciales(pCredenciales)
Descripción:	Invoca la interfaz correspondiente para la impresión de las credenciales.
Nombre:	Construir_Consulta()
Descripción:	En dependencia de los parámetros entrados construye una cadena con la consulta para

	filtrar las credenciales.
Nombre:	Filtrar_Credenciales(pFiltro)
Descripción:	Lleva a cabo el proceso de filtrado para que sólo se muestren las credenciales que cumplan las condiciones de la consulta dada.
Nombre:	Formatear_datos()
Descripción:	Este oculta las columnas que no deben ser visibles, así como asigna un color diferente a las columnas con datos imprimibles y define cuales van a ser modificables.
Nombre:	AdicionarFiltros()
Descripción:	Construye los campos de textos necesarios para poder filtrar las credenciales por cada una de las columnas de las credenciales mostradas.

Tabla 27 Descripción de la clase "Gestor"

Nombre: CrearGrupo	
Tipo de clase Interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	CrearGrupo(pNombreGrupo)
Descripción:	Inicia el proceso de crear un grupo
Nombre:	Validar_Nombre(pNombreGrupo)
Descripción:	Valida que el nombre del nuevo grupo no exista entre los que ya han sido creados, así como que no posea caracteres que no sean alfanuméricos.

Tabla 28 Descripción de la clase "CrearGrupo"

Nombre: ModificarGrupo	
Tipo de clase Interfaz	
Atributo	Tipo
idGrupo	Int
Para cada responsabilidad:	
Nombre:	ModificarGrupo(pldGrupo)
Descripción:	Crea la interfaz almacenando el identificador del grupo al que se le va a modificar el nombre
Nombre:	Validar_Nombre(pNombreGrupo)
Descripción:	Valida que el nombre del nuevo grupo no exista entre los que ya han sido creados, así como que no posea caracteres que no sean alfanuméricos.
Nombre:	ActualizarNombreGrupo(pNombreGrupo, pldGrupo)
Descripción:	Inicia el proceso de modificarle el nombre a un grupo.

Tabla 29 Descripción de la clase "Modificar Grupo"

Nombre: VistaPrevia	
Tipo de clase Interfaz	
Atributo	Tipo
Credencial	Credencial
Para cada responsabilidad:	
Nombre:	VistaPrevia(pCredencial)
Descripción:	Construye una instancia de esta clase y almacena la credencial a mostrar
Nombre:	Mostrar_VistaPrevia(pCredencial)
Descripción:	Inicia el proceso para mostrar la vista previa a la impresión y finalmente la muestra

	visualmente.
--	--------------

Tabla 30 Descripción de la Clases "Vista previa"

Nombre: Impresion	
Tipo de clase Interfaz	
Atributo	Tipo
pCredenciales	Credencial[]
Para cada responsabilidad:	
Nombre:	Impresión(pCredenciales)
Descripción:	Tiene la responsabilidad de crear la interfaz y almacenar el listado de credenciales a imprimir.
Nombre:	Imprimir_Credenciales()
Descripción:	Este método es el que inicia el proceso de impresión de credenciales.

Tabla 31 Descripción de la clase "Impresión"

Clases del Negocio

Nombre: Gestor_Credenciales	
Tipo de clase Controladora	
Atributo	Tipo
credenciales	CtrlCredenciales
Para cada responsabilidad:	
Nombre:	Crear_Credenciales(pNombreGrupo)
Descripción:	Responsable de obtener todas las credenciales que se desean adicionar a la base de datos y enviarlas si tienen los mismos campos que el grupo en el que las están insertando.
Nombre:	Validar_CrearCredenciales(pCredenciales, pldGrupo)
Descripción:	Este método compara la cantidad de campos de una credencial con la del grupo en el que la quieren adicionar, si tienen los mismo campos devuelve verdadero, sino falso.
Nombre:	Actualizar_CamposCredencial(pldCredencial, pNombreCampo, pValor)
Descripción:	Este es el responsable de enviar un nuevo valor de un campo en una credencial para su posterior actualización en la base de datos.
Nombre:	Obtener_UltimaCredencial()
Descripción:	Este método devuelve la última credencial que fue añadida en la base de datos.
Nombre:	Obtener_UltimaCredencial(pNombreGrupo)
Descripción:	Este método devuelve la última credencial adicionada en un grupo dado.

Tabla 32 Descripción de la clase "Gestor Credenciales"

Nombre: GestorGrupos	
Tipo de clase Controladora	
Atributo	Tipo
Grupos	CtrlGrupos
Para cada responsabilidad:	
Nombre:	Crear_Grupo(pNombreGrupo, pProveedor, pPlantilla, pldPadre)
Descripción:	Este método recibe todos los datos necesarios para crear un grupo y los envía para posteriormente adicionarlos en la base de datos.
Nombre:	Actualizar_NombreGrupo(pNombreGrupo, pldGrupo)

Descripción:	Este método recibe el identificador de un grupo y un nuevo nombre para ese grupo, por lo que lo envía para su actualización.
Nombre:	Eliminar_Grupo(pNombreGrupo)
Descripción:	Recibe el nombre de un grupo el cual va a enviar para su posterior eliminación en la base de datos.
Nombre:	Obtener_Grupo(pNombreGrupo)
Descripción:	Este método recibe el nombre de un grupo y obtiene una instancia de la clase Grupo con todos sus datos.
Nombre:	Obtener_NombreGrupos()
Descripción:	Tiene la responsabilidad de obtener todos los nombres de los grupos que hay en la base de datos.

Tabla 33 Descripción de la clase ""

Nombre: GestorProveedor	
Tipo de clase Controladora	
Atributo	Tipo
proveedores	XmlNode
Para cada responsabilidad:	
Nombre:	GestorProveedor()
Descripción:	Carga el fichero Xml donde se encuentra la información de los proveedores y la almacena en proveedores.
Nombre:	Obtener_Proveedores()
Descripción:	Este método devuelve los nombres de todos los proveedores.
Nombre:	Obtener_DatosProveedor(pNombreProveedor)
Descripción:	Este método devuelve el nombre del fichero con el ensamblado de un determinado proveedor.

Tabla 34 Descripción de la clase "Gestore Proveedor"

Nombre: FabricaProveedor	
Tipo de clase Controladora	
Atributo	Tipo
proveedores	XmlNode
Para cada responsabilidad:	
Nombre:	Devolver_InstanciaEnsamblado(pNombreProveedor)
Descripción:	Tiene la responsabilidad de devolver una instancia de un proveedor dado.

Tabla 35 Descripción de la clase "FabricaProveedor"

Nombre: GestorPlantillas	
Tipo de clase Controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	Obtener_Plantillas()
Descripción:	Devuelve todos los nombres de todas las plantillas que se encuentran en el directorio de plantillas.

Tabla 36 Descripción de la clase "GestorPlantillas"

Nombre: Comunicador	
Tipo de clase Controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	Obtener_UltimaCredencial()
Descripción:	Este método devuelve la última credencial que fue adicionada a la base de datos.
Nombre:	Actualizar_ValoresCampo(pCredenciales)
Descripción:	Este método envía un grupo de credenciales con campos que necesitan ser actualizados en la base de datos.

Tabla 37 Descripción de la clase "Comunicador"

Clases de Datos

Nombre: CtrlCredenciales	
Tipo de clase Controladora	
Atributo	Tipo
cadenaConexion	String
Para cada responsabilidad:	
Nombre:	Crear_Credenciales(pCredenciales)
Descripción:	Es el responsable de adicionar una colección de credenciales a la base de datos.
Nombre:	Insertar_CamposCredencial(pCampos, pldCredencial)
Descripción:	Este método recibe una colección de campos de una credencial y los adiciona en la base de datos.
Nombre:	Actualizar_CampoCredencial(pNombreCampo, pValor, pldCredencial)
Descripción:	Este método actualiza el valor de un campo de una credencial en la base de datos.
Nombre:	Obtener_UltimosCampos(pldCredencial)
Descripción:	Obtiene los campos de la credencial que cuyo identificador recibe como parámetro.
Nombre:	Obtener_CredencialGrupo(pNombreGrupo)
Descripción:	Este método tiene la responsabilidad de obtener todas las credenciales de un grupo cuyo nombre recibe como parámetro.
Nombre:	Obtener_UltimaCredencial()
Descripción:	Este método obtiene la última credencial que fue añadida en la base de datos.
Nombre:	Obtener_UltimaCredencial(pNombreGrupo)
Descripción:	Este método devuelve la última credencial adicionada en un grupo dado.

Tabla 38 Descripción de la clase "CtrlCredenciales"

Nombre: CtrlGrupos	
Tipo de clase Controladora	
Atributo	Tipo
Cadenaconexion	String
Para cada responsabilidad:	
Nombre:	Crear_Grupo(pGrupo)
Descripción:	Este método adiciona un grupo que recibe como parámetro en la base de datos.
Nombre:	Actualizar_NombreGrupo(pNombreGrupo, pldGrupo)
Descripción:	Este método actualiza el nombre de un grupo dado con uno que recibe como parámetro en la base de datos.
Nombre:	Eliminar_Grupo(pNombreGrupo)

Descripción:	Este método elimina un grupo cuyo nombre recibe como parámetros, así como todas sus credenciales y campos de ellas.
Nombre:	Obtener_Grupo(pNombreGrupo)
Descripción:	Este método recibe el nombre de un grupo y devuelve una instancia de la clase Grupo con todos los datos que posee en la base de datos.
Nombre:	Obtener_NombreGrupos()
Descripción:	Tiene la responsabilidad de obtener todos los nombres de los grupos que hay en la base de datos.

Tabla 39 Descripción de la clase "CtrlGrupos"

Clases de Comunicación

Nombre: IComunicador	
Tipo de clase Interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	Obtener_UltimaCredencial()
Descripción:	Este método devuelve la última credencial que fue adicionada a la base de datos.
Nombre:	Actualizar_ValoresCampo(pCredenciales)
Descripción:	Este método envía un grupo de credenciales con campos que necesitan ser actualizados en la base de datos.

Tabla 40 Descripción de la clase "IComunicador"

Nombre: IProveedor	
Tipo de clase Interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	Obtener_PosiblesValores(pNombreColumna)
Descripción:	Este método obtiene los posibles valores que puede tomar un campo dado.
Nombre:	Obtener_DatosCredenciales()
Descripción:	Este método devuelve una colección de credenciales que serán adicionadas en la base de datos.
Nombre:	Conocer_ColumnaVisible(pNombreColumna)
Descripción:	Este método devuelve si los campo de una columna dada deben mostrarse cuando se muestren las credenciales de un grupo
Nombre:	Conocer_ColumnaImprimible(pNombreColumna)
Descripción:	Este método devuelve si los campo de una columna son imprimibles o no.
Nombre:	Conocer_ColumnaEditable(pNombreColumna)
Descripción:	Este método tiene la responsabilidad de devolver si se puede actualizar un campo de una columna dada.
Nombre:	Obtener_TipoColumna(pNombreColumna)
Descripción:	Este método dado el nombre de una columna devuelve de qué tipo es.
Nombre:	Obtener_AccesoImagen()
Descripción:	Este método devuelve la dirección mediante la cual se puede acceder a una imagen que necesite la credencial.
Nombre:	Creencias_Impresas(pCredenciales)
Descripción:	Este método envía todas las credenciales que se han impreso, por si se necesita actualizar algún campo.

Nombre:	Establecer_Comunicacion(pComunicador)
Descripción:	Este método es el responsable de que pueda interactuar el proveedor y la aplicación principal dada la implementación de una interfaz que se envía como parámetro.

Tabla 41 Descripción de la clase "IProveedor"

Clases de Impresión

Nombre: GestorImpresion	
Tipo de clase Controladora	
Atributo	Tipo
pagina	Hoja
documento	PrintDocument
datos	Credencial[]
hojaXml	String
tarjetaXml	String
Para cada responsabilidad:	
Nombre:	GestorImpresion(pCredenciales, pHojaXml, pTarjetaXml)
Descripción:	Se encarga de almacenar las credenciales y de construir las hojas con sus respectivas credenciales dentro.
Nombre:	GestorImpresion()
Descripción:	Este método simplemente crea una instancia de la clase.
Nombre:	Imprimir()
Descripción:	Este método tiene la responsabilidad de imprimir las hojas de credenciales ya armadas.
Nombre:	Obtener_Tarjeta(pPosicion, pTarjetaXml, pCredencial, pGrafico)
Descripción:	Este método devuelve un objeto Bitmap dado el Xml, su posición y el objeto Graphics sobre el cual va a dibujar la credencial.

Tabla 42 Descripción de la clase "GestorImpresion"

Nombre: Hoja	
Tipo de clase Entidad	
Atributo	Tipo
tamanno	Size
posiciones	ArrayList
tarjetas	Tarjeta[]
Para cada responsabilidad:	
Nombre:	Hoja(pCredenciales, pHojaXml, pTarjetaXml)
Descripción:	Tiene la responsabilidad de construir luna hoja posicionando cada credencial a imprimir en el lugar que le corresponde.

Tabla 43 Descripción de la clase "Hoja"

Nombre: Tarjeta	
Tipo de clase Entidad	
Atributo	Tipo
campos	Campo[]
tarjetaXml	String
posiciones	ArrayList
Para cada responsabilidad:	

Nombre:	Tarjeta(pCampos, pTarjetaXml)
Descripción:	Tiene la responsabilidad de construir la credencial y posicionar todos los campos dentro de ella.

Tabla 44 Descripción de la clase "Tarjeta"

Nombre: LectorXml	
Tipo de clase Controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	Obtener_ElementosTarjetas(pTarjetaXml)
Descripción:	Se encarga de obtener todos los elementos que hay que imprimir en la credencial.
Nombre:	Obtener_PosicionesTarjeta(pTarjetaXml)
Descripción:	Se encarga de obtener todas las posiciones de los elementos que componen la tarjeta.
Nombre:	Obtener_PosicionesHoja(pHojaXml)
Descripción:	Devuelve todas las posiciones de las credenciales en una hoja.
Nombre:	Obtener_TamannoHoja(pHojaXml)
Descripción:	Este método devuelve el tamaño del material donde se van a imprimir las credenciales.

Tabla 45 Descripción de la clase "Lector Xml"

Clases Comunes

Nombre: Configuracion	
Tipo de clase Controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	Obtener_DirectorioEnsamblado()
Descripción:	Obtiene el camino donde se encuentran los ensamblados de los proveedores.
Nombre:	Obtener_DireccionProveedor()
Descripción:	Se encarga de obtener la dirección donde se encuentra ubicado el Xml con los datos de los proveedores.
Nombre:	Obtener_CadenaConexion()
Descripción:	Este método devuelve la cadena de conexión a la base de datos que utiliza la aplicación.
Nombre:	Obtener_DirectorioPlantillas()
Descripción:	Devuelve la dirección donde se encuentran ubicada las plantillas a usar por cada grupo.

Tabla 46 Descripción de la clase "Configuracion"

Nombre: Grupo	
Tipo de clase Entidad	
Atributo	Tipo
idGrupo	int
descripcion	String
idPadre	int
plantilla	String
proveedor	String
Para cada responsabilidad:	

Tabla 47 Descripción de la clase "Grupo"

Nombre: Credencial	
Tipo de clase Entidad	
Atributo	Tipo
idGrupo	int
descripcion	String
Fecha	DateTime
proveedor	String
Para cada responsabilidad:	

Tabla 48 Descripción de la clase "Credencial"

Nombre: Campo	
Tipo de clase Entidad	
Atributo	Tipo
nombre	String
valor	String
imagen	boolean
Para cada responsabilidad:	

Tabla 49 Descripción de la clase "Campo"

Nombre: Constantes	
Tipo de clase Entidad	
Atributo	Tipo
IDCREDENCIAL	String
FECHA	String
DESCRIPCION	String
IDGRUPO	String
NOMBRE	String
VALOR	String
IMAGEN	String
IDPADRE	String
PLANTILLA	String
PROVEEDOR	String
Para cada responsabilidad:	

Tabla 50 Descripción de la clase "Constantes"

Clases del ProveedorUCI

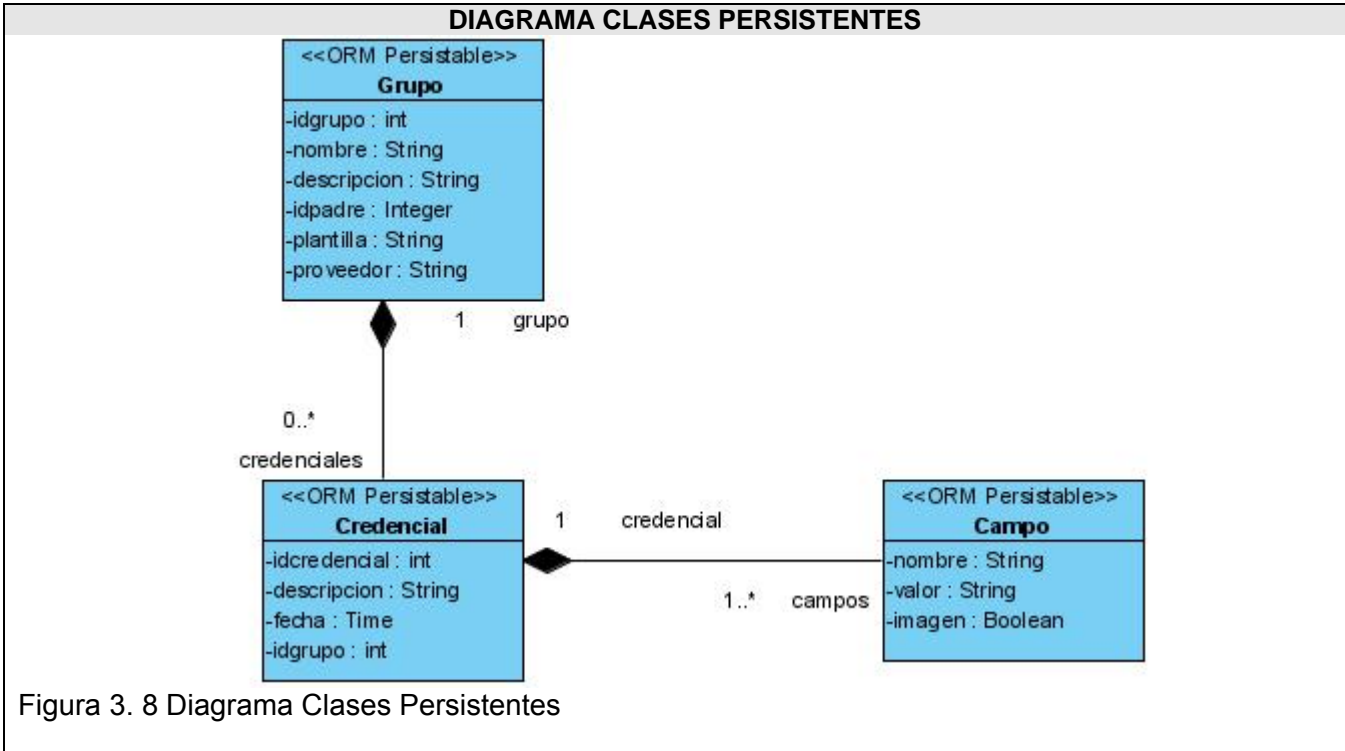
Nombre: ProveedorUCI	
Tipo de clase Controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	Obtener_PosiblesValores(pNombreColumna)
Descripción:	Este método obtiene los posibles valores que puede tomar un campo dado.
Nombre:	Obtener_DatosCredenciales()
Descripción:	Este método devuelve una colección de credenciales que serán adicionadas en la base de datos para lo cual abre una interfaz visual donde se buscarán a las personas a quienes se le quieren crear una credencial.
Nombre:	Conocer_ColumnaVisible(pNombreColumna)
Descripción:	Este método devuelve si los campo de una columna dada deben mostrarse cuando se muestren las credenciales de un grupo
Nombre:	Conocer_ColumnaImprimible(pNombreColumna)
Descripción:	Este método devuelve si los campo de una columna son imprimibles o no.
Nombre:	Conocer_ColumnaEditable(pNombreColumna)
Descripción:	Este método tiene la responsabilidad de devolver si se puede actualizar un campo de una columna dada.
Nombre:	Obtener_TipoColumna(pNombreColumna)
Descripción:	Este método dado el nombre de una columna devuelve de qué tipo es.
Nombre:	Obtener_AccesoImagen()
Descripción:	Este método devuelve la dirección mediante la cual se puede acceder a una imagen que necesite la credencial.
Nombre:	Creenciales_Impresas(pCredenciales)
Descripción:	Este método recibe las credenciales que ya se han impreso para actualizarle el estado en que se encuentran.
Nombre:	Establecer_Comunicacion(pComunicador)
Descripción:	Este método recibe una instancia de la clase IComunicacion para poder interactuar con la

aplicación principal.

Tabla 51 Descripción de la clase "ProveedorUCI"

Diseño de la base de datos

Uno de los aspectos más importantes para cualquier aplicación es el almacenamiento de la información.



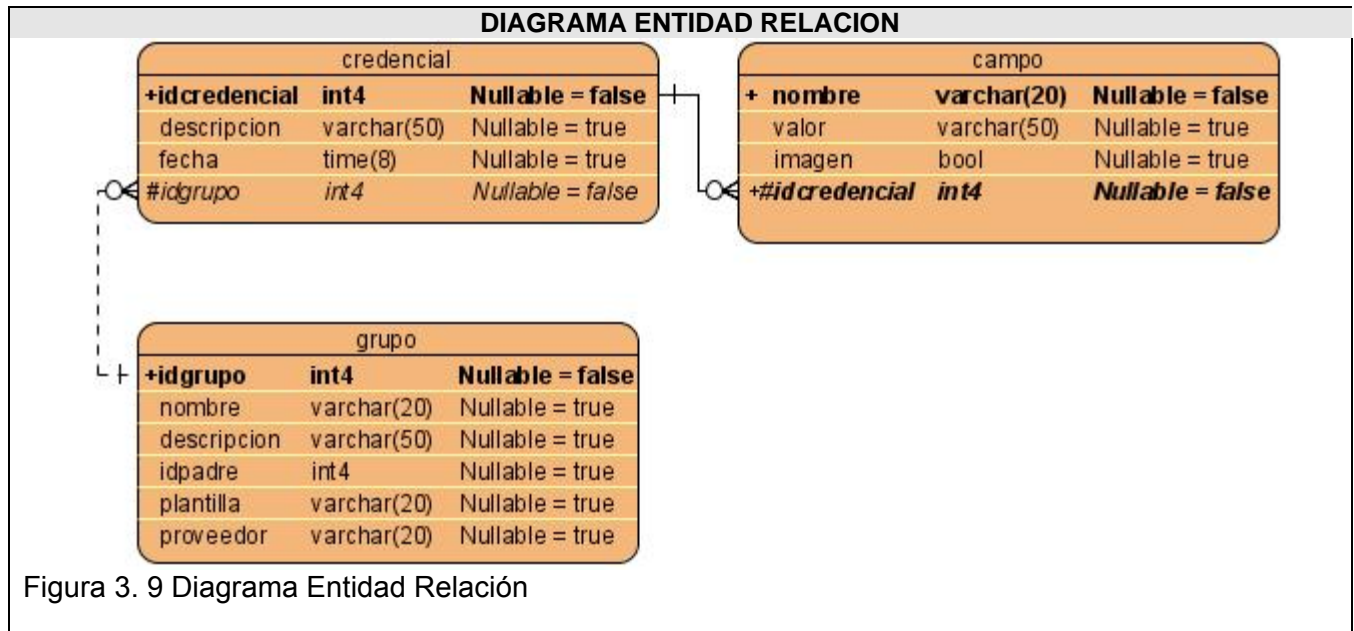


Figura 3. 9 Diagrama Entidad Relación

Descripción de las tablas

A continuación se mostrará una descripción de cada una de las tablas del diagrama entidad relación de la base de datos de este sistema.

Nombre: grupo		
Descripción: Agrupa conjuntos de credenciales y define la plantilla y proveedor de datos a utilizar.		
Atributo	Tipo	Descripción
idgrupo	int	Llave de cada grupo.
nombre	varchar	Nombre del grupo.
descripcion	varchar	Breve descripción del grupo.
idpadre	int	Identificador del grupo padre.
plantilla	varchar	Nombre de la plantilla que se usará a la hora de imprimir las credenciales de dicho grupo.
proveedor	varchar	Nombre del proveedor que define las reglas de negocio de las credenciales de este grupo.

Tabla 52 Descripción de la tabla grupo

Nombre: credencial		
Descripción: Contiene las credenciales, así como la fecha de su creación.		
Atributo	Tipo	Descripción
idcredencial	int	Llave que identifica cada credencial.
descripcion	varchar	Breve descripción de la credencial creada.

fecha	Time	Fecha y hora en que se creó la credencial.
idgrupo	int	Llave foránea que contiene el identificador del grupo a que pertenece la credencial.

Tabla 53 Descripción de la tabla credencial

Nombre: campo		
Descripción: Almacena los valores que contiene cada credencial		
Atributo	Tipo	Descripción
nombre	varchar	Llave que identifica esta tabla y que contiene el nombre del campo de la credencial.
valor	varchar	Contiene el valor del campo de la credencial.
imagen	boolean	almacena si el campo es imagen o no.
idcredencial	int	Llave foránea que identifica la credencial a la que pertenece el campo.

Tabla 54 Descripción de la tabla campo

Tratamiento de errores

En el desarrollo de la aplicación un factor muy importante a tener en cuenta es el tratamiento de los errores. El sistema que se propone es capaz de prevenir errores por parte del usuario, esto se logra minimizando las áreas donde el usuario tiene que entrar datos de forma manual, como es el caso de a la hora de crear o modificar un grupo en que se valida que los nombres no existan ya en la base de datos, así como que no contengan caracteres no alfanuméricos. También se valida a la hora de hacer búsquedas en el proveedor de datos, validándose la cadena entrada. El resto de las acciones que realiza este sistema lo hace obteniéndolos de bases de datos y WebServices. Cada vez que ocurre una acción incorrecta el usuario es notificado, así como alertado cuando va a eliminar grupos de credenciales.

Seguridad

Este sistema no posee una gran seguridad debido a que sólo hay una persona que deberá usar el sistema, es decir, no existirá roles, ni varios usuarios, sólo una persona se encargará de trabajar con el sistema. La aplicación que se propone es de escritorio, o sea no acceden personas a ella por la red.

Interfaz

Teniendo en cuenta que el sistema será utilizado por personas que no deberán tener grandes conocimientos de computación, se ha elaborado una interfaz lo más sencilla posible, de forma tal que quien use la aplicación sepa intuitivamente qué tiene que hacer para llevar a cabo una acción determinada. Además se muestran pequeños mensajes explicativos sobre cada uno de los elementos que componen los formularios utilizados en esta propuesta. Existe un diseño único que se ha aplicado a todos los formularios para lograr que entre todos los elementos haya uniformidad, aunque hayan sido creados de forma dinámica como ocurre con la mayoría en este sistema.

Concepción de la ayuda

El sistema contará con un manual de usuario que podrá ser utilizado por la persona que operará el sistema, pues aunque la aplicación se ha hecho de forma tal que sea intuitiva, este podrá servir de apoyo ante cualquier duda del usuario. Además para garantizar que quienes vayan a operar el sistema comprendan completamente su funcionamiento cómo agregar proveedores y plantillas, se impartirán cursos de entrenamientos donde se el operador podrá ver cómo se trabaja con el mismo.

Conclusiones

En este capítulo se describieron las clases y el resto de los elementos necesarios para la implementación del sistema propuesto. Además dichas clases fueron organizadas en paquetes de acuerdo a su papel en la aplicación. Con el diagrama de clases del diseño se logró determinar las clases que serán implementadas, así como sus atributos y responsabilidades. También se definió es el diseño de la interfaz de la propuesta, que hará que el usuario final de la aplicación disfrute de una interfaz amigable, sencilla, intuitiva.

CAPÍTULO 4. Implementación

Introducción

Se expondrán en este capítulo el diagrama de despliegue y el de implementación. Ellos son sumamente importantes dentro del flujo de trabajo de implementación, pues ambos diagramas conforman lo que se conoce como modelo de implementación. Además se describen y explican los conceptos relacionados con dichos diagramas.

Arquitectura

Para la definición de la arquitectura de esta propuesta se tuvo en cuenta el modelo n capas el cual utiliza como uno de sus principios el desarrollo de aplicaciones basadas en componentes. Este tipo de arquitectura presenta varios beneficios:

- Aplicaciones más robustas debido al encapsulamiento.
- Mantenimiento y soporte más sencillo (es más fácil cambiar un componente que no una aplicación monolítica)
- Mayor flexibilidad (se le pueden añadir más componentes que aumenten las funcionalidades del sistema)
- Soluciones altamente flexibles y reutilizables.
- Alta escalabilidad.

Diagrama de despliegue

El diagrama de despliegue representa cómo va a estar distribuido el sistema de forma física. Se utiliza un servidor con la Base de Datos de la aplicación, en el computador cliente se encontrará el sistema, con sus proveedores y plantillas y se utiliza una impresora para llevar a cabo la impresión de las credenciales. Los nodos entre los diagramas se conectan con asociaciones de comunicación entre las que pueden estar: enlaces de red, conexiones TCP/IP, microondas. En esta propuesta no se puso un protocolo de impresión en específico, ya que puede ser utilizado cualquiera de ellos, ya sea USB, conexiones por puerto serie o

paralelo. La vista representa la distribución de las instancias de componentes en ejecución en instancias de nodos que se encuentran conectados por enlaces de comunicación. Un nodo es un recurso en ejecución como puede ser un computador, un dispositivo, un procesador o memoria. Esto se identifica utilizando estereotipos.

Existe un servidor PostgreSQL que contiene la base de datos que utiliza el sistema para almacenar la información que el mismo maneja.

Diagrama de Despliegue

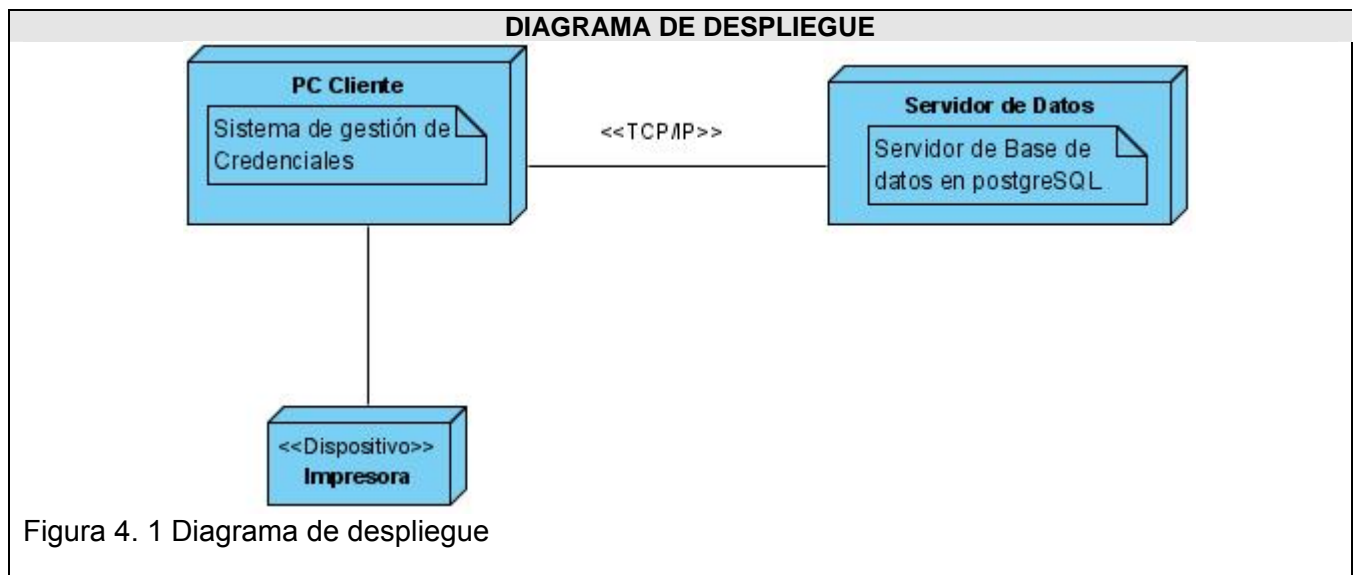


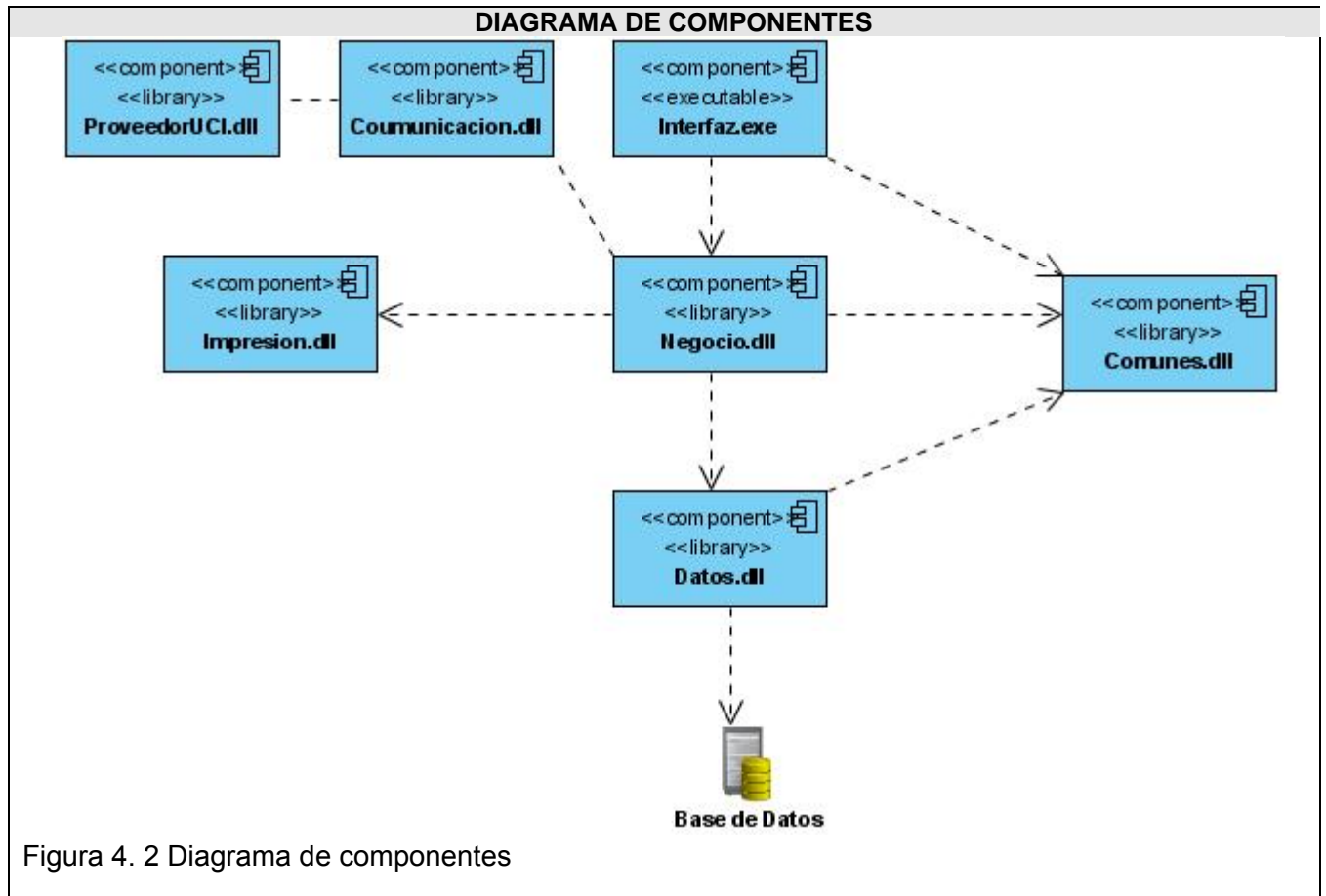
Figura 4. 1 Diagrama de despliegue

Diagrama de componentes

Los componentes representan todos los tipos de elementos software que entran en la fabricación de aplicaciones informáticas. Las relaciones de dependencia se utilizan en los diagramas de componentes para indicar que un componente utiliza los servicios ofrecidos por otro componente. Un diagrama de componentes representa las dependencias entre componentes software, incluyendo componentes de código fuente, componentes del código binario, y componentes ejecutables. Un módulo de software se

puede representar como componente. Algunos componentes existen en tiempo de compilación, algunos en tiempo de enlace y algunos en tiempo de ejecución, otros en varias de éstas. Los componentes tienen dos características: Empaquetan el código que implementa la funcionalidad de un sistema, y algunas de sus propias instancias de objetos que constituyen el estado del sistema.

Diagrama de Componentes



Conclusiones

En este capítulo se representó en un diagrama los componentes a ser implementados, además del diagrama de despliegue que brinda información acerca de cómo estará físicamente distribuido el sistema.

Se hizo además una descripción de la arquitectura utilizada en el desarrollo de la aplicación y de cuáles son las ventajas de hicieron que se seleccionara para en esta propuesta.

CAPÍTULO 5. Estudio de Factibilidad

Introducción

Para desarrollar un proyecto de software es imprescindible realizar el análisis del costo y los beneficios que éste reportará, considerando las ventajas de minimizar los gastos de esfuerzos, de fondos y de tiempo de planificación, lo cual permitirá evaluar si es conveniente llevarlo a cabo.

En este capítulo se realizará un estudio del costo y los beneficios del sistema propuesto y a partir de éste se obtendrán el tiempo de desarrollo del proyecto y el costo del mismo.

Planificación basada en casos de uso

1er. Paso: Cálculo de los Puntos de casos de uso Desajustados

El cálculo de los Puntos de Casos de Uso sin ajustar es el primer paso para la estimación mediante el análisis de Puntos de Casos de Uso. Este valor 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

Haciendo un análisis de la cantidad de Actores presentes en el sistema y la complejidad de cada uno de ellos se calcula el Factor de Peso de los Actores sin ajustar (UAW).

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		

	(API, Application Programming Interface).			
Medio	Otro sistema que interactúa con el sistema a desarrollar mediante un protocolo o una interfaz basada en texto.	2		
Complejo	Una persona que interactúa con el sistema mediante una interfaz gráfica.	3	1	3
Total			1	3

Tabla 55 Factor de peso de los actores sin ajustar

$$UAW = \sum cant \ actores * peso$$

UAW=3*1=3 factor de peso de los actores sin ajustar

Haciendo un análisis de la cantidad de la cantidad de Casos de Uso presentes en el sistema y la complejidad de cada uno de ellos se calcula el Factor de Peso de los Casos de Uso sin ajustar (UUCW). Dicha complejidad se establece teniendo en cuenta la cantidad de transacciones efectuadas en el mismo.

Tipo de CU	Descripción	Factor de peso	Cantidad de CU	Total
Simple	El caso de uso tiene de 1 a 3 transacciones.	5	10	50
Medio	El caso de uso tiene de 4 a 7 transacciones.	10		
Complejo	El caso de uso tiene más de 8 transacciones.	15		
Total			10	50

Tabla 56 Factor de peso de los casos de uso sin ajustar

$$UUCW = \sum cant\ CU * Peso$$

UUCW = 10 * 5 = 50 factor de peso de los CU sin ajustar

$$UUCP = UAW + UUCW$$

UUCP = 3 + 50 = 53 puntos de casos de uso sin ajustar

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

$$UCP = UUCP * TCF * EF$$

Donde:

UCP: puntos de casos de uso ajustados.

UUCP: puntos de casos de uso sin ajustar.

TCF: factor de complejidad técnica.

EF: factor de ambiente.

El factor de complejidad técnica (TCF) se determina mediante la cuantificación de un conjunto de factores que determinan la complejidad técnica del sistema.

Cada uno de los factores es cuantificado con un valor de 0 a 5, donde 0 significa un aporte irrelevante y 5 un aporte relevante.

Factor	Descripción	Peso	Comentarios	Valor Asignado	Total
T1	Sistema distribuido	2	El sistema es centralizado.	0	0
T2	Tiempo de respuesta	1	La velocidad es limitada por la conectividad.	3	3
T3	Eficiencia del usuario final	1	Requiere de conocimiento para tratar con el sistema.	3	3
T4	Funcionamiento Interno complejo	1	No existen cálculos muy complejos.	1	1
T5	El código debe ser	1	El código será reutilizable.	5	5

	reutilizable				
T6	Facilidad de instalación	0.5	No implica una instalación compleja.	3	1.5
T7	Facilidad de uso	0.5	Fácil de usar.	5	2.5
T8	Portabilidad	2	No se requiere que el sistema sea portable.	0	0
T9	Facilidad de cambio	1	Costo moderado de mantenimiento.	4	4
T10	Concurrencia	1	No existe concurrencia.	0	0
T11	Incluye objetivos especiales de seguridad	1	No incluye objetivos especiales de seguridad	4	4
T12	Provee acceso directo a terceras partes	1	Si los tiene.	4	4
T13	Se requieren facilidades especiales de entrenamiento de usuarios	1	Pocos usuarios, fácil de usar.	2	2
Total				34	30

Tabla 57 Factor de complejidad técnica

$$TCF = 0.6 + 0.01 * \sum (peso * valor asignado)$$

$$TCF = 0.6 + 0.01 * 30 = 0.9 \text{ factor de complejidad técnica}$$

El factor de ambiente (EF) está relacionado con el entrenamiento y las habilidades del grupo protagonista en el desarrollo del sistema. El cálculo del mismo se determina mediante la cuantificación de un conjunto de factores con un valor de 0 a 5, donde 0 significa un aporte irrelevante y 5 un aporte relevante.

Factor	Descripción	Peso	Comentarios	Valor Asignado	Total
E1	Familiaridad con el modelo	1.5	El grupo está	3	4.5

	de proyecto utilizado		medianamente familiarizado con el modelo.		
E2	Experiencia en la aplicación	0.5	La mayoría del grupo ha trabajado durante algún tiempo en ésta aplicación.	2	1
E3	Experiencia en la orientación a objetivos.	1	La mayoría del grupo programa en objetos.	4	4
E4	Capacidad del analista líder.	0.5	El líder es un Ing. Informático.	5	2.5
E5	Motivación.	1	Existe motivación por parte del grupo.	3	3
E6	Estabilidad de requerimientos	2	Existe probabilidad de cambios.	4	8
E7	Personal Part–Time	-1	No todo el grupo trabaja a tiempo completo.	2	-2
E8	Dificultad del lenguaje de programación	-1	Se usará lenguaje C#.	4	-4
Total				28	17.5

Tabla 58 Factor de ambiente

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

$$EF = 1.4 - 0.03 * 17 = 0.89 \text{ factor de ambiente}$$

$$UCP = UUCP * TCF * EF$$

$$UCP = 53 * 0.9 * 0.89 = 42.453 \text{ puntos de casos de uso ajustados}$$

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

A partir de la siguiente ecuación matemática se calcula el esfuerzo en horas hombres:

$$E = UCP * CF$$

Donde:

E: esfuerzo estimado en horas hombres

UCP: punto de casos de usos ajustados

CF: factor de conversión

Se utiliza el factor de conversión 20 Horas-Hombre / Punto de Casos de uso.

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

E = 42.453 * 20 = 849.06 Horas-Hombre

4to. Paso: Calcular esfuerzo de todo el proyecto

Para una estimación por completo de la duración total del proyecto, hay que agregar las estimaciones de esfuerzo de las demás actividades relacionadas con el desarrollo de software a la estimación del esfuerzo obtenida por los Puntos de Casos de Uso. Para ello se puede tener en cuenta el siguiente criterio, que estadísticamente se considera aceptable. El criterio plantea la distribución del esfuerzo entre las diferentes actividades de un proyecto, según la siguiente aproximación [7]:

Actividad	% esfuerzo	Valor esfuerzo (horas-hombre)
Análisis	10%	212.265
Diseño	20%	424.53
Implementación	40%	849.06
Prueba	15%	318.3975
Sobrecarga	15%	318.3975
Total	100%	2122.65

Tabla 59 Esfuerzo del proyecto

Si el esfuerzo total (E_T) es de 2122.65 **horas-hombre** y por cada 240 horas laborables se tiene 1 mes, eso daría un esfuerzo total de 8.844375 **mes-hombre**.

Esto quiere decir que 1 persona puede realizar el sistema analizado en aproximadamente 9 meses.

Costo del Proyecto

Se asume como salario promedio mensual \$100.00

Cantidad de hombres: $CH = 1$

Costo Hombre-Mes: $CHM = 1 * \text{Salario Promedio}$

$CHM = 100.00 \text{ \$/mes}$

$\text{Costo} = CHM * E_T / CH$

$\text{Costo} = 100.00 * 8.844375 / 1 = 884.4375$

$\text{Costo} = \$884.4375 \approx \884

Tiempo ≈ 9 meses

Beneficios tangibles e intangibles

El Sistema de Gestión de Credenciales no es un software con fines comerciales, aunque puede fácilmente reutilizarse en otros entornos ya que el mismo constituye una solución general.

Su objetivo principal es mejorar la gestión e impresión de credenciales en la Universidad de las Ciencias Informáticas, para aumentar la seguridad interna, razón por la cual los beneficios inmediatos son mayormente intangibles:

- Mayor organización y control de las credenciales de las personas.
- Aumenta la flexibilidad para incorporar nuevos tipos de credenciales.
- Permite mantener una imagen de cómo son y han sido cada una de las credenciales generadas por el sistema.
- Vista previa de cómo quedará la credencial, lo que permite corregir errores antes de imprimir.
- Fácil manejo, instalación y configuración.

Análisis de costos y beneficios

El desarrollo de la propuesta no supone grandes gastos de tiempo, ni de recursos. La base de datos puede ser alojada en el servidor de PostgreSQL que está montando la Dirección de Informatización, por lo que no tendrá problemas de prestaciones. El sistema utiliza el Framework 2.0 de Microsoft el cual es gratis.

En el transcurso del trabajo se investigaron otras soluciones que existen en el mundo respecto a la gestión de credenciales y se llegó a la conclusión de que no se ajustan a las características del entorno de negocio de la UCI, siendo además soluciones muy costosas.

El sistema propuesto dado su gran flexibilidad puede convertirse sin prácticamente cambios en un producto comercializable, convirtiéndose en una fuente de ingresos.

Conclusiones

En este capítulo se describió el estudio de factibilidad realizado correspondiente al sistema propuesto, teniendo en cuenta el costo estimado y los beneficios que reportará al ser puesto en marcha. Teniendo esto en cuenta se decidió que era factible llevar a cabo la implementación de la propuesta, ya que los beneficios que reporta a la universidad son considerables.

CONCLUSIONES

Con el desarrollo de este trabajo se hace una propuesta de solución al problema de la gestión e impresión de credenciales en la Universidad de las Ciencias Informáticas, para lo cual primero se hizo un estudio de los diferentes sistemas que existen a nivel mundial en lo que se refiere a la gestión de credenciales. Se presenta una solución informática capaz de llevar a cabo la creación de credenciales, su impresión, de agrupar las credenciales de forma intuitiva, de poder actualizar los campos no imprimibles de la credencial. Esta aplicación puede llegar a convertirse en una importante herramienta para tener el control de las credenciales que existen en nuestro centro y poder gestionar este proceso con eficacia y rapidez como se hace necesario en la actualidad, con el gran cúmulo de personas que necesitan una. Además el hecho de que posea un historial de cómo fue exactamente una credencial en un momento dado, permite conocer cómo fue la credencial de una persona o si esa persona alguna vez poseyó una credencial aumentando por consiguiente la seguridad interna del centro.

Con el presente trabajo queda una documentación sobre el proceso de gestión de credenciales, el cual no existía hasta el momento en la universidad, lo cual permitirá su estudio para futuras modificaciones cuando sea necesario darle soporte.

El sistema construido cumplió con todos los objetivos propuestos, lográndose una solución integral en lo que respecta a la gestión de credenciales. El sistema resultante está provisto de una interfaz amigable, sencilla e intuitiva, aplicando un diseño uniforme y utilizando modernas técnicas de programación orientada a objetos.

RECOMENDACIONES

Los objetivos que se trazó el trabajo han sido logrados, pero han surgido nuevas ideas que pueden servir para mejorar el mismo.

1. Migrar la aplicación a Java, para logra que la aplicación sea hecha completamente utilizando herramientas libres, logrando además que el sistema sea multiplataforma y funcione en todos los sistemas operativos en los que se puede instalar la máquina virtual de Java.
2. Utilizar el sistema en otros centros o instituciones convirtiéndose así en una fuente de ingresos.
3. Continuar desarrollando el sistema adecuándolo a nuevas necesidades.

REFERENCIAS BIBLIOGRÁFICAS

1. PÉREZ ZURITA, P. and T. L. OÑA CRUZ. *Sistema de Control de Accesos a la Universidad de las Ciencias Informáticas*, Universidad de las Ciencias Informáticas, 2006. 96. p.
2. Visual Paradigm. *Features*, 2007. [Disponible en:
<http://www.visual-paradigm.com/product/vpuml/features/>]
3. Microsoft Corporation. Lo nuevo en Visual C# 2005, 2007a. [Disponible en:
[http://msdn2.microsoft.com/es-es/library/t27ahy37\(vs.80\).aspx](http://msdn2.microsoft.com/es-es/library/t27ahy37(vs.80).aspx)]
4. Microsoft Corporation. Lo nuevo de Net Framework 2.0, 2007b. [Disponible en:
[http://msdn2.microsoft.com/es-es/library/t357fb32\(vs.80\).aspx](http://msdn2.microsoft.com/es-es/library/t357fb32(vs.80).aspx)]

BIBLIOGRAFÍA

Sun Microsystems. *UML Modeling: Creating Sequence Diagrams*, 2007. [Disponible en: http://developers.sun.com/jenterprise/learning/tutorials/jse8/uml_sequence_diagram.html]

Bell, Donald. *UML's Sequence Diagrams*, 2004. [Disponible en: <http://www.ibm.com/developerworks/rational/library/3101.html>]

W. Ambler, Scott. *UML 2 Sequence Diagrams*, 2006. [Disponible en: <http://www.agilemodeling.com/artifacts/sequenceDiagram.htm>]

Gil Martín, M.A.: *Sistema de Control de Accesos*. Trabajo de Diploma para optar por el título de Ingeniero Informático, Instituto Superior Politécnico “José Antonio Echeverría”, Universidad de las Ciencias Informáticas 2005, 106 p.

MBCESStore Mexico. *Evolis eMedia Software*, 2006. [Disponible en: <http://www.mbcestore.com.mx/evolis/emedial.htm>]

MBCESStore Mexico. *Datacard ID Works ID Card Software*, 2006. [Disponible en: <http://www.mbcestore.com.mx/datacard/id-works-id-card.htm>]

MBCESStore Mexico. *Number Five Card Five Software*, 2006. [Disponible en: <http://www.mbcestore.com.mx/number-five/cardfive.htm>]

Daniel Díaz, Moisés. *Cómo desarrollar una arquitectura de software: lenguaje de patrones*, 2007. [Disponible en: <http://www.webtaller.com/maletin/articulos/como-desarrollar-una-arquitectura-software-lenguajes-patrones.php>]

Wikipedia. *Arquitectura de software*, 2007. [Disponible en: http://es.wikipedia.org/wiki/Arquitectura_software]

Canchala, Armando. *Fundamentos de la POO*, 2007. [Disponible en:
http://www.microsoft.com/spanish/msdn/comunidad/mtj.net/voices/MTJ_2578.asp]

Microsoft Corporation. *Estilos y Patrones en la Estrategia de la Arquitectura de Microsoft*, 2007.
[Disponible en: http://www.microsoft.com/spanish/msdn/arquitectura/roadmap_arg/style.asp]

Dirección de Informatización. *Arquitectura para los Sistemas que Conforman la Intranet Universitaria*, 2007
[Disponible en: <http://uddi/arquitectura.2007.5.9.pdf>]

Francia H, Joel. *Reflexion y sus aplicaciones*, 2007. [Disponible en:
<http://www.microsoft.com/spanish/msdn/comunidad/mtj.net/voices/art121.asp>]

GLOSARIO DE TÉRMINOS

USB: (Universal Serial Bus, Puerto Serie Universal) Es un puerto de conexión del computador, que permite conectar un dispositivo y utilizarlo sin tener que reiniciar la máquina.

PVC: (PolyVinyl Chloride, Cloruro de polivinilo): material plástico altamente flexible.

TCP/IP: (Transfer Control Protocol/ Internet Protocol, Protocolo de Control de Transmisión/ Protocolo de Internet) Protocolo de transmisión de datos en Internet, ampliamente utilizado, permite conectar computadoras con diferentes sistemas operativos.

Credencial: Documento que sirve para que a un empleado se le reconozca como perteneciente a su centro de trabajo y se reconozca su plaza.

Entidad: En este caso es todo lo que describe existencia, animada o material.

Acreditación: Proceso de creación de credenciales a una entidad dada.

HTML: (Hyper Text Markup Language, Lenguaje de Marcas de Hipertextuales) lenguaje de marcación diseñado para estructurar textos y presentarlos en forma de hipertexto, que es el formato estándar de las páginas Web.

XML: (eXtensible Markup Language, Lenguaje de Marcado Extensible) desarrollado por el World Wide Web Consortium (W3C).

DIAGRAMA DE INTERACCION <<Actualizar Credencial>>

sd Actualizar Credencial

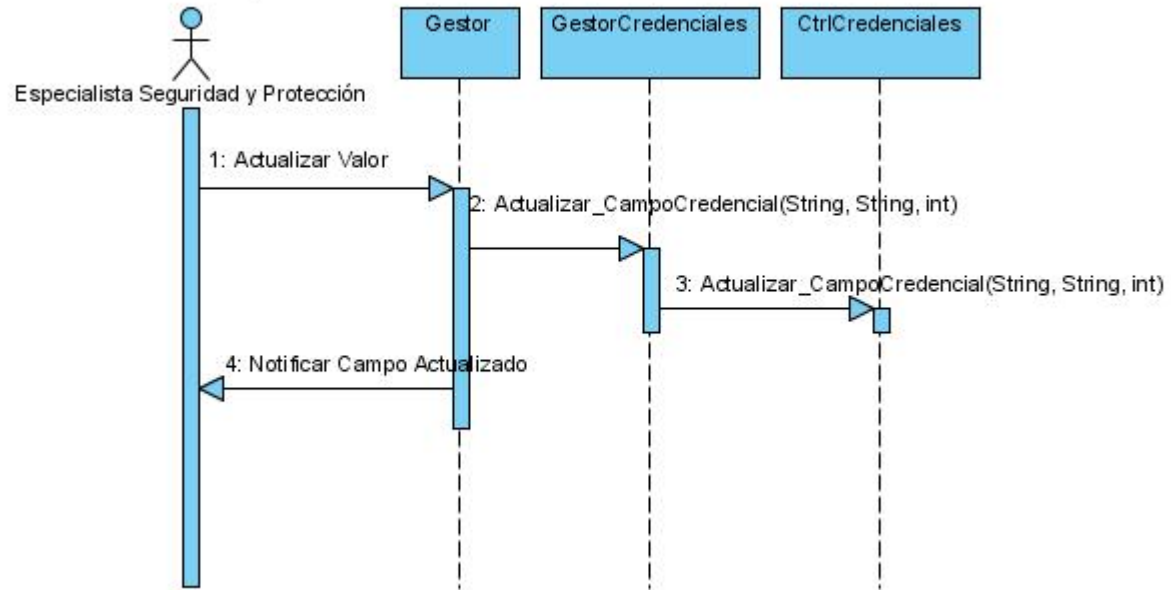


DIAGRAMA DE INTERACCION <<Eliminar Grupo>>

sd Eliminar Grupo

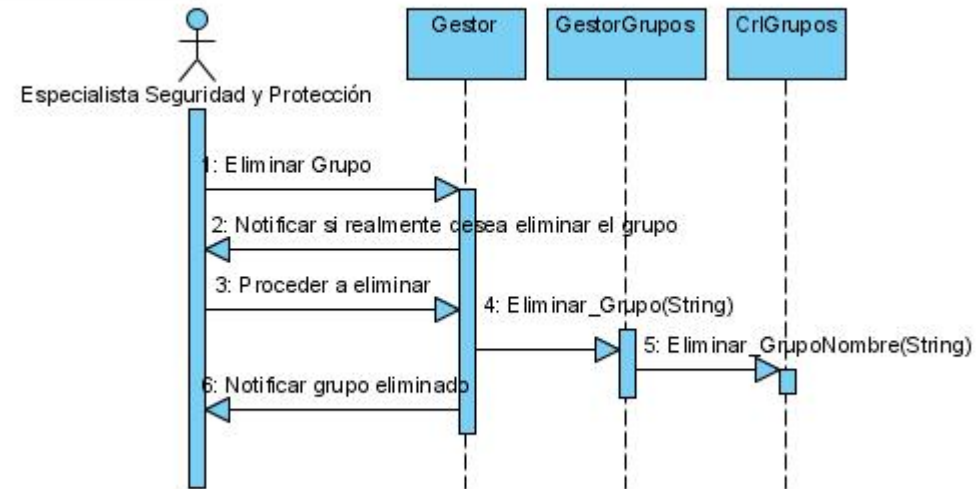


DIAGRAMA DE INTERACCION <<Filtrar Credenciales>>

sd Filtrar Credenciales

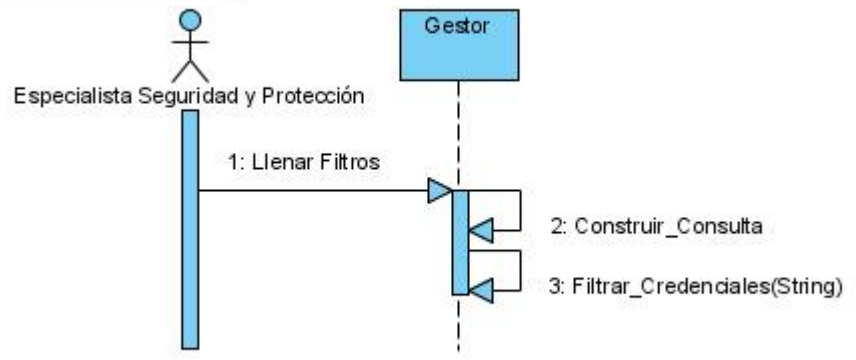


DIAGRAMA DE INTERACCIÓN <<Mostrar Listado de Credenciales>>

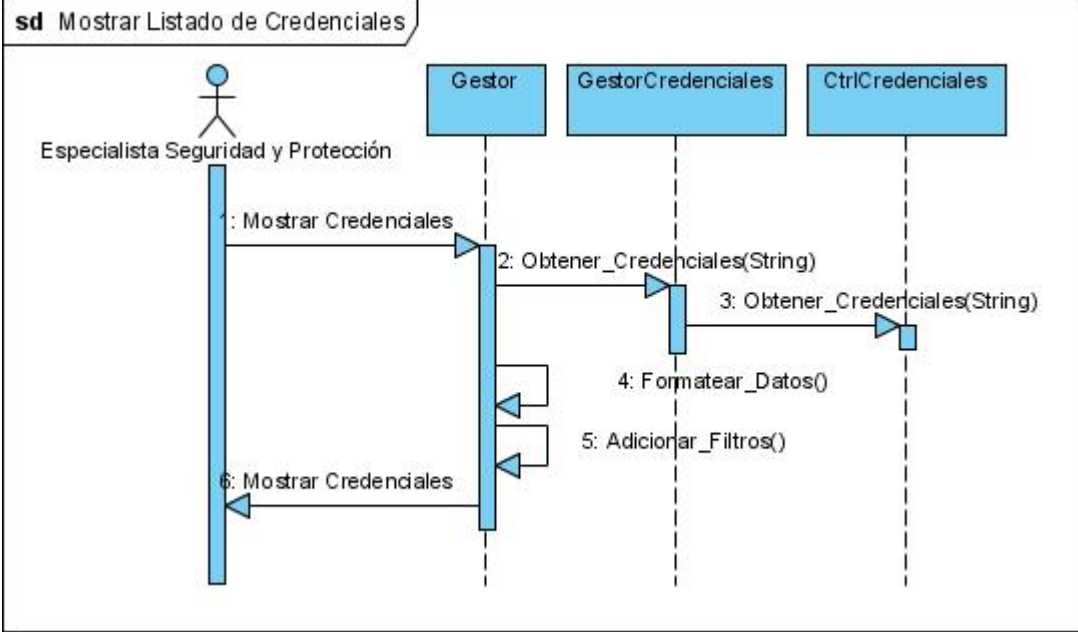


DIAGRAMA DE INTERACCION <<Modificar Nombre Grupo>>

sd Modificar Nombre Grupo

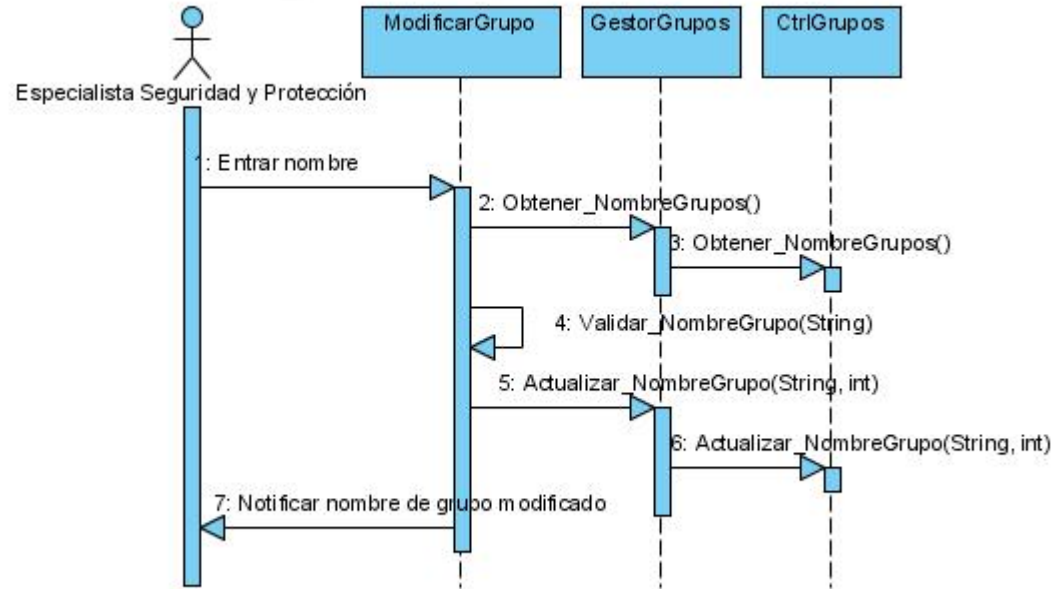


DIAGRAMA DE INTERACCION <<Visualizar Credencial>>

sd Visualizar Credencial

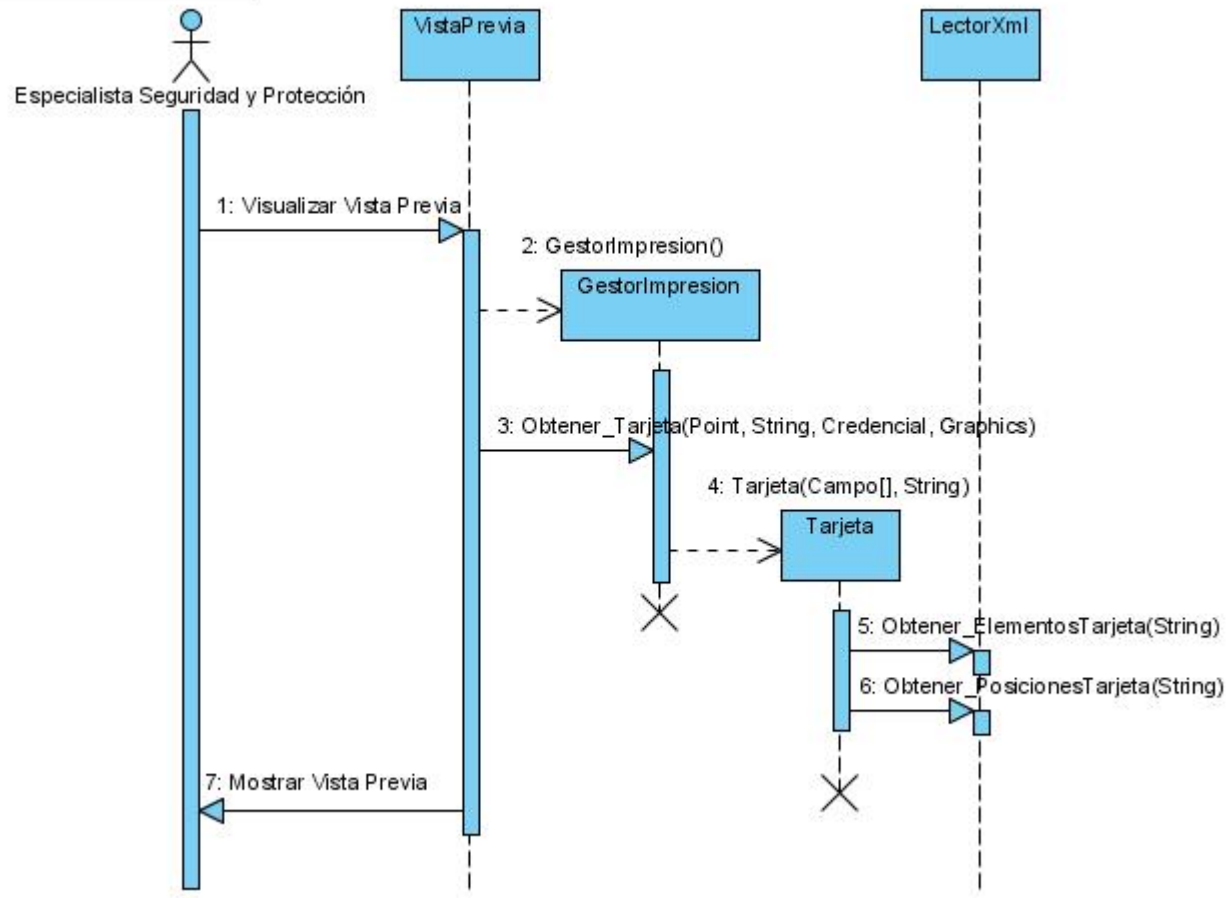


DIAGRAMA DE INTERACCION <<Crear Nueva Credencial>>

sd Crear Nueva Credencial

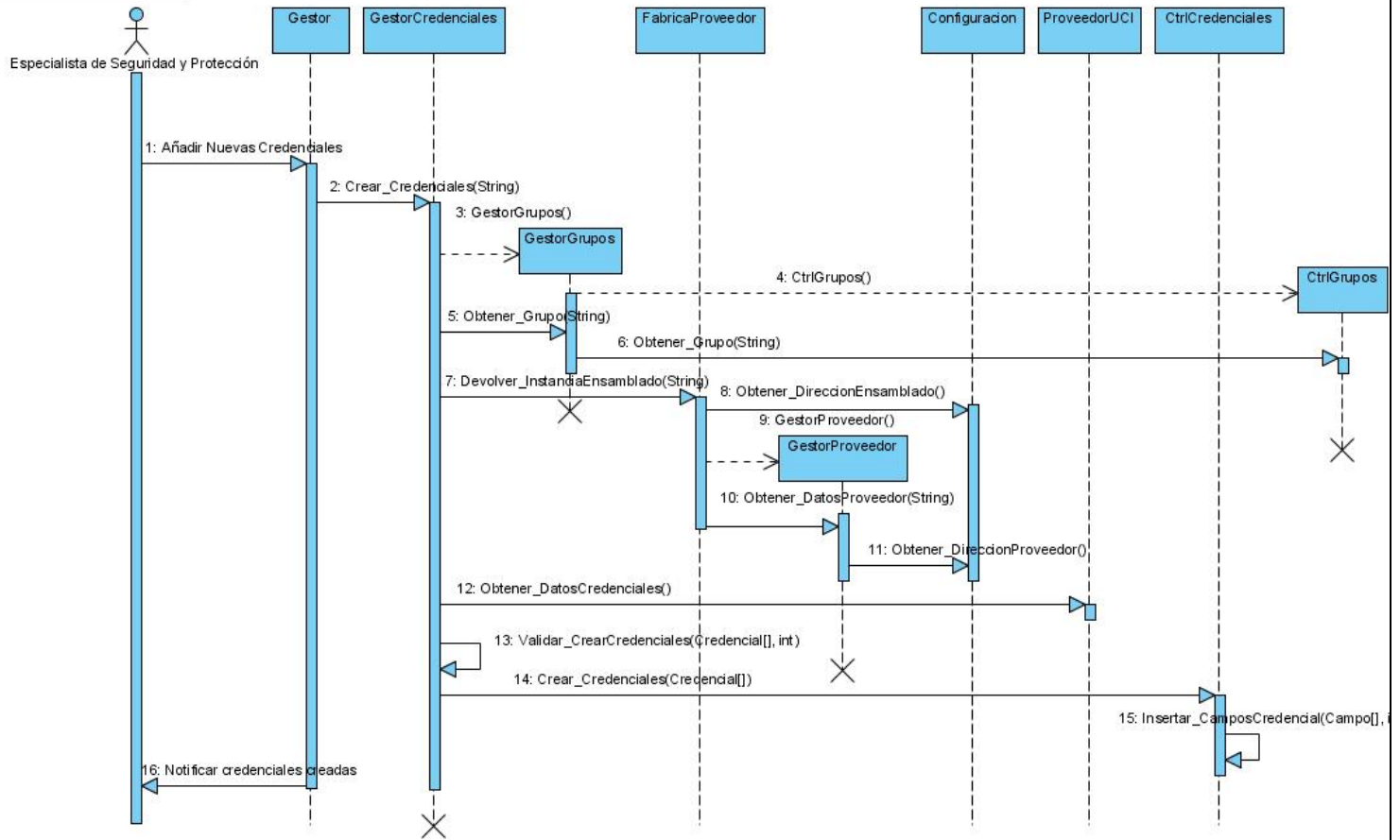


DIAGRAMA DE INTERACCION <<Imprimir Credencial>>

sd Imprimir Credencial

