Universidad de las Ciencias Informáticas

Facultad 1

Sistema de Administración de Tarjetas Inteligentes y Aplicaciones para la Cédula de Identidad Electrónica de la República Bolivariana de Venezuela.

Trabajo de Diploma para optar por el título de

Ingeniero en Ciencias Informáticas

Autor(es): Michel Rafael Pérez Costa.

Luis Fernández Leyva.

Tutor(es): Lic. Reina Mora González.

Lic. Edistio Yoel Verdecia Martínez.

La Habana, junio de 2008.

Hombre es algo más que ser torpemente vivo: es entender una misión, ennoblecerla y cumplirla.

José Martí.

DECLARACIÓN DE AUTORÍA

Declaramos que somos los únicos autores de este trabajo y autorizamos a la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

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

_____ _____

Michel Rafael Pérez Costa Lic. Reina Mora González

_____ _____

Luis Fernández Leyva Lic. Edistio Yoel Verdecia Martínez

AGRADECIMIENTOS

A la Revolución, a Fidel y a Raúl por regalarnos esta gran casa que ha sido la UCI, por la oportunidad de formarnos y permitirnos ser útiles a nuestra patria.

A nuestros tutores Reina y Edistio por ayudarnos siempre, por brindarnos todos sus conocimientos; por hacer posible con su dedicación y esfuerzo el desarrollo de este trabajo.

A todos los integrantes del Proyecto Identidad en especial a Adonis, Landrián, Yosvany, Alina, Erick, Machado y Yurdik por todas las ideas que nos aportaron; por todas las respuestas que encontramos con su ayuda.

A Labrada, Ismar, Abdel, Haidesita y Ezequiel por hacer suyo también este trabajo.

A todos los que se interesaron.

Muchas Gracias.

Luis y Michel.

DEDICATORIA

A mi mama y a mi papa porque todo lo que soy se lo debo a la enseñanza moral, intelectual y física que recibí de ustedes.

A mi hermana por estar ahí para mí.

A mi familia y amigos por confiar en mí.

Luis.

A mi mamá, a Magi, a mima y al viejo Rafa por hacer de mi todo lo que soy hoy, por todo su amor, por sus regaños y besos, por tantos buenos consejos, por confiar siempre en mí, por enseñarme que la familia es lo más importante que tenemos; por todo su apoyo.

A Alejandro y a Oli por ser las dos nuevas luces de mi vida, por hacer que pueda respirar mejor cuando están cerca, por esperarme siempre y a pesar de todo.

A los amigos; los que están y a los que ya no, por cada minuto compartido, por los buenos y malos momentos.

A la “Familia”.

Michel.

RESUMEN

Como parte de las transformaciones que se han venido desarrollando en La República Bolivariana de Venezuela a partir de la llegada del poder de Hugo Rafael Chávez Frías, en el centro de estas transformaciones ha estado el Sistema de Identificación Nacional Venezolano. Estas transformaciones se realizan con el fin de recuperar la confianza de los ciudadanos en dicho sistema y en el documento de identidad, a partir de los logros del Proyecto de Modernización del Sistema de Identificación, Migración y Extranjería en su Fase I, incluido el Pasaporte Electrónico, las autoridades Venezolanas acuerdan la utilización de un documento de identificación soportado en la utilización de tarjetas inteligentes para fortalecer dicho sistema.

En este marco se precisa la creación de un Sistema de Administración de Tarjetas Inteligentes y Aplicaciones para la nueva cédula de identidad electrónica de la República Bolivariana de Venezuela; que permita gestionar su ciclo de vida, las aplicaciones contenidas en la tarjeta y los servicios a los cuales pueden acceder los ciudadanos a través de su utilización; con el objetivo de un aumento de la seguridad y de la confianza en el documento.

Para garantizar el cumplimiento de esta tarea, en el presente trabajo se describe el análisis y diseño de un sistema automatizado que permita integrar, acoplar y mantener actualizados todos los módulos relacionados con el proceso de cedulación, y su nuevo soporte; y que además se integre al Sistema SAIME.

El documento recoge los resultados de la investigación realizada; describiéndose las principales características de los sistemas de gestión de tarjetas inteligentes analizados, se detallan los procesos relacionados con el ciclo de vida de las tarjetas de identificación y se propone el registro automatizado de los mismos; quedando de esta forma concebida, a través de diferentes artefactos, la documentación del sistema propuesto que garantizan poder realizar la implementación del mismo.

ÍNDICE DE CONTENIDOS.

ÍNDICE DE TABLAS.

ÍNDICE DE FIGURAS.

Índice de figuras.

INTRODUCCIÓN

La República Bolivariana de Venezuela, luego del triunfo revolucionario del presidente Hugo Rafael Chávez Frías, se enfrascó en una ardua batalla contra los males económicos, políticos y sociales que la afectaron a lo largo de varias décadas de explotación y saqueo. La corrupción, el desvío de recursos, la fuga de efectivo al exterior y la mutilación de los derechos más básicos de la población eran prácticas comunes en la sociedad venezolana.

Los profundos cambios recogidos en la Constitución Bolivariana presentada por el presidente de la República, se veían afectados, en cierta medida, por la vulnerabilidad existente en todo el Sistema de Identificación Nacional, el cual, caduco y abandonado a merced de los malos manejos de funcionarios corruptos, excluía a un número significativo de personas de contar con un documento legal que les brindara la posibilidad de identificarse como ciudadanos venezolanos.

En el seno de este marco político es que surge la Misión Identidad, la cual brindó la posibilidad a miles de ciudadanos venezolanos de contar con una cédula de identificación y con ella alcanzar los derechos que les habían sido cercenados durante siglos por no contar con una prueba legal que validara su situación ante la sociedad.

El éxito de esta misión demostró que el pueblo y las instituciones de la naciente República Bolivariana estaban listos para iniciar, a una mayor escala, el perfeccionamiento de su sistema de identificación. Para este entonces surge el Proyecto Identidad, encargado de desarrollar el Sistema SAIME (*Servicio Autónomo de Identificación, Migración y Extranjería*), cuyo objetivo primordial es la reestructuración, modernización y automatización de todos los procesos relacionados con las gestiones realizadas por la ONIDEX (*Oficina Nacional de Identificación y Extranjería*), la cual ha iniciado un ambicioso programa de trabajo para insertar las más seguras y modernas tecnologías en todas las áreas relacionadas con el Sistema de Identificación venezolano (Dirección de Informática de la ONIDEX, 2007).

Con el avance de las tecnologías y en aras de satisfacer las necesidades presentes y futuras de la sociedad en materia de identificación, las instituciones encargadas del tema acuerdan que se impone una segunda fase en el sistema que se había venido desarrollando. Esta nueva etapa debe basarse en el empleo de tarjetas de identificación electrónica, las cuales contarán con lo más novedoso en materia de identidad y con múltiples medidas de seguridad. Se distinguirán por poseer un chip que almacenará la información necesaria para comprobar, con alto grado de seguridad, la identidad de su portador y que facilitará el desarrollo de servicios para la Sociedad y el Gobierno Electrónicos.

La nueva cédula contará con un mecanismo de verificación biométrico para garantizar la unicidad en la identificación. Además, tendrá la capacidad de asociar a un solo documento diferentes servicios, minimizando la cantidad de trámites, reduciendo costos administrativos, garantizando la seguridad en las transacciones y el manejo de información, y principalmente fortaleciendo la seguridad nacional.

El empleo de una tarjeta inteligente brinda la posibilidad de contar con múltiples aplicaciones, por lo que además de ser un documento de identificación puede usarse como documento de viaje, tarjeta de salud, tarjeta de crédito, en fin brindar acceso físico o lógico seguro a diferentes recursos. Además, se le pueden establecer diferentes niveles de acceso a la información contenida en ella, así como los roles en que puede operar en dependencia del grado de autorización con que cuente su portador.

Las tarjetas inteligentes que se emplearán en las cédulas de identificación venezolana deben transitar por diferentes estados durante su ciclo de vida, que se inicia en el proceso de fabricación, transita por las fases de producción, pasa los controles de calidad, se entrega al ciudadano y se emplea según los servicios habilitados, hasta que por algún motivo se inhabilite y quede fuera de servicio.

Partiendo de las nuevas posibilidades que brinda este tipo de documento surge entonces la ***Situación Problémica*** a la cual pretendemos dar solución con este trabajo, y que se enuncia a continuación:

El sistema de identidad actual no cuenta con un proceso de administración de tarjetas inteligentes y aplicaciones para la nueva cédula de identidad electrónica de la República Bolivariana de Venezuela, que permita controlar todos los aspectos relacionados con el ciclo de vida de las tarjetas y las aplicaciones que pueden contener.

Derivado de esta situación real que presenta el Sistema de Identificación Nacional de la República Bolivariana de Venezuela nos encontramos frente al siguiente ***Problema Científico:***

¿Cómo gestionar el ciclo de vida y las aplicaciones de la nueva cédula de identidad electrónica de la República Bolivariana de Venezuela?

El presente trabajo plantea como punto de partida la siguiente ***Pregunta científica***:

¿Con el adecuado diseño del Sistema de Administración de Tarjetas Inteligentes y Aplicaciones [1]para la nueva cédula de identidad electrónica de la República Bolivariana de Venezuela se contará con los artefactos necesarios para ejecutar en el tiempo previsto la implementación del sistema?

[1] CAMS, por sus siglas en inglés, Card and Application Management System.

Ideas a Defender:

- Con el análisis y diseño del CAMS se contará con los artefactos necesarios para ejecutar en el tiempo previsto la implementación del sistema para la cédula electrónica de la RBV.
- Un CAMS para la nueva cédula de identidad electrónica de la RBV será un mecanismo para incrementar la seguridad nacional y la confianza de los ciudadanos en el sistema de identidad.

Al concluir este trabajo se espera como **posible resultado:**

Contar con los artefactos necesarios para el posterior desarrollo de un CAMS para la nueva cédula de identidad electrónica de la República Bolivariana de Venezuela que se integre con el Sistema Nacional de Identificación, Migración y Extranjería.

Nuestro ***Objeto de Estudio*** es la ***Transformación del Sistema Nacional de Identificación, Migración y Extranjería de la República Bolivariana de Venezuela*****;** de esta forma podemos definir además el **Campo de Acción** de este trabajo será el ***Sistema de Administración de Tarjetas Inteligentes y Aplicaciones*** asociado a la nueva solución.

Para dar cumplimiento a lo antes planteado tomamos como ***Objetivo Principal***:

Realizar el análisis y diseño de un sistema que permita gestionar el ciclo de vida de la cédula de identidad electrónica de la RBV.

Derivándose a su vez de éste varios ***Objetivos Específicos****:*

- Realizar un estudio sobre el estado del Arte de los CAMS y las Tarjetas Inteligentes.
- Definir los requerimientos funcionales y no funcionales del sistema.
- Elaborar la documentación necesaria para la posterior implementación del CAMS.
- Realizar la integración del CAMS con el resto de los módulos del Sistema de Identidad.

Para dar respuesta a la interrogante presentada en este trabajo, cumplir con lo planteado en la hipótesis propuesta y con los objetivos trazados se plantea el cumplimiento de las siguientes ***Tareas***:

1. Escribir un reporte del estado del arte de los Sistemas de Administración de Tarjetas Inteligentes y Aplicaciones existentes en el mundo.
2. Escribir un reporte del estado del arte del uso de las tarjetas inteligentes con motivos de identificación.
3. Investigar sobre los aspectos teóricos conceptuales y prácticos de las posibles herramientas a utilizar para realizar el Análisis y el Diseño del Sistema informático.
4. Seleccionar de la herramienta a utilizar para realizar el Análisis y el Diseño del sistema informático.

5. Listar y definir los requerimientos funcionales y no funcionales del sistema.
6. Precisar y documentar los casos de uso del sistema.
7. Crear el diagrama de clases a partir de los casos de uso.
8. Detallar el diseño de la base de datos necesaria para la implementación del CAMS y su integración con la base de datos del Sistema SAIME.

Para una mejor comprensión y facilitar el estudio se estructura el presente trabajo en varios capítulos donde se detalla el proceso investigativo realizado, así como el análisis y diseño del Sistema de Administración de Tarjetas Inteligentes y Aplicaciones que se propone.

Capítulo 1. Fundamentación Teórica: Este capítulo muestra un estudio a cerca de las tarjetas inteligentes, sus principales características, y su utilización como documento de identidad electrónico; se analizan algunos ejemplos de los CAMS más representativos en el mercado; en busca de sus características principales, se hace referencia a las tendencias y tecnologías actuales propuestas para el desarrollo de sistemas informáticos.

Capítulo 2. Características del Sistema: Se hace una propuesta de sistema, para ello se estudiaron los principales procesos vinculados al objeto de estudio y al campo de acción de nuestro trabajo, los cuales se muestran a través de casos de uso, los actores que intervienen, sus relaciones y se hace una descripción de cada uno de ellos teniendo en cuenta los requisitos funcionales y no funcionales que debe cumplir el sistema.

Capítulo 3. Análisis y Diseño del sistema: Se confeccionaron y documentaron los principales artefactos que conforman la propuesta; los diagramas de clases de análisis y diseño para cada Caso de Uso del sistema, así mismo se exponen los correspondientes diagramas de interacción y el modelo de datos correspondiente con nuestro sistema; se integra todo al Sistema SAIME.

Capítulo 1 : Fundamentación Teórica

1.1. Introducción

Según la definición de la Real Academia Española de la Lengua, “La identidad es el conjunto de rasgos propios de un individuo o de una colectividad que los caracterizan frente a los demás”, y también “Conciencia de que una persona tiene de ser ella misma y distinta a las demás”. Como se puede apreciar la identidad personal es un concepto importante que toma aún más valor en la actual Sociedad de la Información. De esta forma se entiende la necesidad de establecer los medios y mecanismos más adecuados para que el Estado otorgue esta identidad personal a sus ciudadanos. Otorgar identidad personal a los ciudadanos adquiere una nueva dimensión cuando se trata de establecerla para un uso no presencial en medios telemáticos. Y aunque la identidad siempre es física, es necesario establecer mecanismos y procedimientos electrónicos para verificarla en estos nuevos ámbitos (Diccionario de la Real Academia de la Lengua Española, 2007).

En este capítulo se hace un estudio de algunos *Sistemas de Administración de Tarjetas Inteligentes y Aplicaciones* utilizados por diferentes sistemas de identificación electrónica, se mencionan sus principales características, ventajas y desventajas, así como sus semejanzas y diferencias más significativas. A partir de toda esta información e incorporando otras funcionalidades definidas según las necesidades de las diferentes instituciones interesadas, esperamos confeccionar un diseño general de nuestro sistema.

Dado que la nueva cédula de identidad electrónica es una tarjeta inteligente necesitamos un sistema de administración de tarjetas y aplicaciones por lo que incluimos en este capítulo un estudio sobre éstas y su utilización con motivo de identificación en varios países; además, se exponen las tecnologías usadas para el desarrollo del sistema.

1.2. Situación de la identidad en el mundo

Las tarjetas de identidad son utilizadas, en una u otra forma alrededor del mundo. Su tipo y función varía de un país a otro. Entre los elementos claves de una tarjeta está su número de serie, el cual es utilizado como un mecanismo administrativo para identificar la tarjeta desde su proceso de fabricación y asociar al portador de la tarjeta con un conjunto de actividades en diversas áreas mediante el número de identidad permanente; que también está incluido en la tarjeta, siendo este ultimo el de mayor importancia en la cédula.

En estos momentos en el mundo hay más de 100 países en los que de una forma u otra los ciudadanos deben poseer una cédula de identidad. Existe un grupo de países en los cuales se estudian legislaciones al respecto y hay un fuerte debate en torno a la privacidad (DevelpmentTeam, 2008).

Con el paso del tiempo y el desarrollo tecnológico surgido en el área de la identificación, las tarjetas inteligentes han rebasado los límites de las empresas bancarias y telefónicas convirtiéndose en una de las normas de seguridad más utilizadas por muchas instituciones en todo el mundo para garantizar la identidad de sus portadores, aplicándose incluso como documento de identificación oficial en varios países.

La identificación empleando tarjetas inteligentes varía su forma de empleo dependiendo en gran medida de las leyes que para ello hayan sido trazadas por los diferentes gobiernos. Presentan diseños diferentes pero en su mayoría cuentan con un número de identificación y un grupo de elementos de diferentes niveles de seguridad, estos serán analizados más adelante en el capítulo.

1.2.1. La identificación electrónica en el mundo

Sólo en un pequeño grupo de países, de los que en la actualidad utilizan algún tipo de documento de identificación[2], se ha implementado la utilización de tarjetas inteligentes con tal motivo. Se destacan en este sentido los países pertenecientes a la Unión Europea (UE), los cuales plantean una solución basada en especificaciones comunes que permiten la interoperabilidad entre países, con el fin de que en 2010 los ciudadanos y empresas europeos puedan contar con sistemas de identificación electrónica seguros que les permitan identificarse ante la administración de su propio país o de cualquier otro miembro de la UE (Europes Information Society, 2008).

La implantación de sistemas de identificación nacional basados en la utilización de tarjetas inteligentes ha permitido a diversos estados fortalecer su seguridad nacional y brindar a los ciudadanos e instituciones un mecanismo más seguro de probar su identidad.

A pesar de ser esta nueva forma de identificación de probada efectividad, no es implementada en la generalidad de los países debido a que en la actualidad los recursos necesarios para su funcionamiento son muy costosos y existen pocos proveedores de los mismos.

[2] Ver Anexo1 ¨ Resumen de países que utilizan documentos de identificación en el mundo ¨.

1.3. Tecnologías utilizadas en la identificación

1.3.1. Tarjetas inteligentes

Una tarjeta inteligente (*smart card*), o tarjeta con circuito integrado (TCI), es cualquier tarjeta de tamaño pequeño con circuitos integrados incluidos que permitan la ejecución de cierta lógica programada. Aunque existe un diverso rango de aplicaciones, hay dos categorías principales de TCI: Las tarjetas de memoria que contienen sólo componentes de memoria no volátil y posiblemente alguna lógica de seguridad y las tarjetas microprocesadores que contienen memoria y microprocesador (Smart Cards Alliance, 2008).

El origen de las tarjetas inteligentes se encuentra en Europa a comienzos de los años setenta del siglo XX, alcanzando su mayor auge en los noventa, con la introducción de las tarjetas SIM (acrónimo de *Subscriber Identity Module*, 'Módulo de Identificación del Suscriptor') utilizadas en la telefonía móvil GSM (acrónimo de Global System for Mobile Communications. 'Sistema Global para las Comunicaciones Móviles'). Existen algunas discusiones de quién es el "inventor" original; entre los que se encuentran Juergen Dethloff de Alemania, Kunitaka Arimura de Japón y Roland Moreno de Francia. Una de sus primeras aplicaciones fue el ser utilizadas como tarjetas de crédito y débito por el sector bancario; siendo las firmas internacionales MasterCard, Visa, y Europay quienes publicaron por primera vez un estándar de interoperabilidad para el pago con este tipo de dispositivo (Effing, 2008).

1.3.1.1. Clasificaciones

Las tarjetas inteligentes se pueden clasificar según diversos criterios:

- Según la capacidad de su chip.
- Según la estructura de su sistema operativo.
- Según la interfaz de comunicación.

Pueden existir otras clasificaciones pero en este trabajo utilizaremos fundamentalmente las citadas anteriormente (Effing, 2008).

1.3.1.2. Tipos de tarjetas según la capacidad del chip

- **Memoria:** tarjetas que únicamente funcionan para el almacenamiento de datos. No son capaces de procesar información. Se usan generalmente en aplicaciones sin altos requisitos de seguridad.

- **Microprocesador:** tarjetas con una estructura análoga a la de una computadora (procesador, memoria volátil, memoria persistente). Éstas albergan ficheros y aplicaciones, pudiendo implementar avanzados mecanismos de seguridad para proteger la información contenida en ellas. Suelen usarse en sistemas de identificación y sistemas de pago (monederos electrónicos).
- **Criptográficas:** tarjetas con microprocesador muy avanzadas que cuentan con un coprocesador criptográfico para la ejecución de algoritmos complejos usados para el cifrado de la información y la firma digital de documentos. Estas tarjetas pueden almacenar de forma segura uno o varios certificados digitales, se pueden utilizar para firmar documentos o autenticar al titular de la misma sin que la información "sensible" contenida en su memoria salga de ella.

1.3.1.3. Tipos de tarjetas según la estructura de su sistema operativo

- **Tarjetas de memoria:** Disponen de un elemental sistema operativo limitado a una serie de comandos básicos de lectura y escritura de las distintas secciones de memoria y la protección de la información está condicionada a la presentación de un código secreto.
- **Tarjetas basadas en ficheros:** Estas tarjetas disponen del equivalente a un sistema de ficheros MS-DOS con dos niveles de jerarquía. Hay directorios y ficheros. Tienen un sistema operativo con un conjunto de comandos que le ofrecen las operaciones básicas para el acceso a los datos y la protección de la información.
- **Tarjetas Java:** Es una tarjeta capaz de ejecutar mini-aplicaciones Java. Su sistema operativo es una pequeña máquina virtual Java (JVM) y en ellas se pueden cargar dinámicamente aplicaciones desarrolladas específicamente para este entorno.

1.3.1.4. Tipos de tarjetas según la interfaz de comunicación

- **Tarjetas de Contacto:** Son las que necesitan ser insertadas en un lector para ser accedidas mediante los contactos físicos del chip.
- **Tarjetas sin Contacto:** Son las que se comunican usando radio frecuencia. Para ser accedidas sólo se requiere la proximidad al lector. El chip se encuentra en la parte interna y está conectado a una antena que permite la comunicación con el lector.
- **Tarjetas Dual:** Cuentan con un único chip que se comunica a través de dos interfaces: de contacto y sin contacto. Poseen las características de las mencionadas anteriormente.

- **Tarjetas Híbridas:** Son aquellas que tienen dos chips independientes cada uno con una interfaz diferente: uno con interfaz de contacto y otro con una interfaz sin contacto.

1.3.1.5. Estructura de una tarjeta inteligente

Internamente, el chip de una tarjeta inteligente con microprocesador se compone de:

- **CPU (Central Processing Unit):** el procesador de la tarjeta; suelen ser de 8 bits, a 5 MHz y 5 voltios. Pueden tener opcionalmente módulos hardware para operaciones criptográficas.
- **ROM (Read-Only Memory):** memoria interna, (normalmente entre 12 y 30 KB), en la que se establece el sistema operativo de la tarjeta, las rutinas del protocolo de comunicaciones y los algoritmos de seguridad de alto nivel por software. Esta memoria, como su nombre indica, no se puede reescribir y se inicializa durante el proceso de fabricación.
- **EEPROM:** memoria de almacenamiento, (equivalente al disco duro en un ordenador personal), en el que está grabado el sistema de ficheros, los datos usados por las aplicaciones, claves de seguridad y las propias aplicaciones que se ejecutan en la tarjeta. El acceso a esta memoria está protegido a distintos niveles por el sistema operativo de la tarjeta.
- **RAM (Random Access Memory):** memoria volátil de trabajo del procesador.

1.3.2. Seguridad

La seguridad es una de las propiedades más importantes de las tarjetas inteligentes y se aplica a múltiples niveles y con diferentes mecanismos. Las medidas de protección van desde los elementos empleados en el material de la tarjeta, así como las características usadas en la impresión gráfica que dificultan la reproducción y alteración de las mismas; hasta reglas que se establecen para el acceso a la información contenida en el chip, según sus niveles de confidencialidad.

1.3.2.1. Seguridad física

El conjunto de medidas de seguridad que se pueden implementar para un documento de identidad se dividen en diferentes niveles[3]:

- **Nivel 1:** Perceptibles mediante la vista al observar el documento. No requieren de herramientas especiales, por lo cual no es necesario entrenamiento. Pueden ser de conocimiento público.
- **Nivel 2:** Características escondidas que son visibles mediante equipos simples, luz ultravioleta. No requieren de un entrenamiento especial de los oficiales de seguridad.

[3] Ver Anexo 2¨ Medidas de Seguridad Física de las Tarjetas Inteligentes ¨.

- **Nivel 3:** Características de seguridad que requieren de un entrenamiento al personal y de un equipamiento especial para detectarlas, por ejemplo: un microscopio o lupas especiales.
- **Nivel 4:** Características de un alto nivel de seguridad sobre las que sólo tienen conocimiento el personal requerido y que únicamente pueden verse mediante un equipamiento en un laboratorio especializado.

1.3.2.2. Seguridad lógica

En las tarjetas se implementan distintos niveles de seguridad sobre los ficheros en dependencia de la importancia de la información contenida en ellos. Algunos elementos de seguridad son:

- Especificación de un conjunto de reglas para ejecutar comandos o acceder a datos, las cuales se denominan Condiciones de Acceso.
- Mensajería segura para la autenticación e integridad del intercambio de datos entre la tarjeta y los lectores.
- Protección de la información por claves que deben presentarse a la tarjeta y son verificadas por su sistema operativo.
- Contador de confirmación para evitar agresiones repetidas contra los valores secretos.
- Mecanismos de autenticación mutua de la tarjeta y los terminales para garantizar que sólo elementos autorizados puedan tener acceso a la información.
- Funciones criptográficas que permiten los procesos de firma digital para prevenir repudios.
- Utilización de la infraestructura de llave pública (*PKI*).
- Verificación biométrica por parte de la tarjeta.

1.3.2.3. JavaCard

La tecnología JavaCard combina parte del lenguaje de programación Java con un entorno de ejecución optimizado para Tarjetas Inteligentes y similares. El objetivo de la tecnología JavaCard es llevar los beneficios del desarrollo de software en Java al mundo de las Tarjetas Inteligentes.

El JCRE (acrónimo de *JavaCard Runtime Environment*, Entorno de Ejecución de JavaCard) comprende la máquina virtual de JavaCard (*JCVM*) junto a las clases y servicios definidos en el *Application Programming Interface* (*API*). Sobre este ambiente se ejecutan los applets que se desarrollan. Los applets son las aplicaciones que corren embarcadas en una JavaCard. Dichas aplicaciones interactúan en todo momento con el JCRE utilizando los servicios que éste brinda (Smart Cards Forum., 2003).

1.3.2.4. Seguridad del sistema operativo

La especificación de JavaCard provee ciertos mecanismos para la interacción entre applets residentes en la misma tarjeta, así como elementos que rigen la seguridad de dicha interacción, los cuales son presentados en la siguiente sección.

- **Applet Firewall:** Es una tecnología que refuerza la seguridad más allá de las protecciones que tiene la JCVM por sí misma. Los chequeos que ésta implica se realizan durante la ejecución de la aplicación.
- **Cambios de Contexto:** En todo momento sólo puede haber un contexto activo en la JCVM. Todo el código de bytes es chequeado en tiempo de ejecución para determinar si tiene permiso para acceder al contexto actualmente activo.
- **Contextos Grupales:** Este permite que dos o más applets pertenecientes a un mismo paquete puedan compartir el mismo contexto. También establece todas las instancias de un mismo Applet pueden compartir el mismo contexto.
- **Objetos:** Cuando un objeto es creado queda asociado al contexto activo. El objeto pertenece a la instancia del Applet que lo creó o en caso que fuera un objeto creado por el JCRE éste es el dueño del mismo.
- **Protección del Firewall:** El firewall intenta solucionar varios problemas de seguridad.
- **Errores del desarrollador** o agujeros de seguridad en el diseño de los applets.
- **Acceso a objetos en otros contextos:** El JCRE provee una serie de mecanismos para permitir la comunicación entre objetos que estén en diferentes contextos.
- **Privilegios del JCRE:** El JCRE tiene privilegios que lo habilitan a acceder a cualquier método o campo de cualquier objeto que esté en cualquier contexto.

1.3.2.5. Comunicación Tarjeta-Lector

- Toda comunicación que se realice con una tarjeta es iniciada siempre por el dispositivo externo, esto quiere decir que la tarjeta nunca transmite información sin que se haya producido antes una petición externa. Esto equivale a una relación maestro-esclavo, siendo el terminal el maestro y la tarjeta el esclavo.
- Cada vez que una tarjeta interactúa con un lector, sus contactos se conectan a los del terminal y éste procede a activarlos eléctricamente, a continuación, la tarjeta inicia un reset de encendido y envía una respuesta llamada ATR (*Answer To Reset*) hacia el terminal. Esta

respuesta contiene información referente a cómo ha de ser la comunicación tarjeta-lector, estructura de los datos intercambiados, protocolo de transmisión, etc.

- Una vez que el lector interpreta el ATR procede a enviar la primera instrucción. La tarjeta procesa la orden y genera una respuesta que es enviada hacia el terminal. El intercambio de instrucciones y respuestas acaba una vez que la tarjeta no es energizada.

1.3.3. Biometría

La biometría es el estudio de métodos automáticos para el reconocimiento único de humanos basados en uno o más rasgos conductuales o físicos intrínsecos.

La "biometría informática" es la aplicación de técnicas matemáticas y estadísticas sobre los rasgos físicos o de conducta de un individuo, para “verificar” identidades o para “identificar” individuos. Recurrir a procedimientos y herramientas biométricas constituye en la actualidad uno de los instrumentos más usados en la validación de la identidad. Algunos de los elementos biométricos por los cuales se puede reconocer a un individuo son:

- **Huella**. Las yemas de los dedos tienen una piel corrugada con líneas que forman una especie de surcos de un lado a otro del dedo. El flujo de estos surcos no es continuo y está lleno de terminaciones y bifurcaciones que forman un patrón que es diferente en todas las personas. Este patrón se llama minucia y constituye la base para el reconocimiento de huellas, ya que no cambian con el tiempo. Si antes la toma de huellas se realizaba a través del procedimiento tradicional de entintar los dedos y hacer impresiones en cartones dactilares, ahora los sistemas han evolucionado y se hace de forma digital: la persona pone su dedo en un lector que identifica los puntos clave de la huella de ese individuo, ingresa la información y permite el cotejo posterior de los datos.
- **Iris**. El iris es la parte coloreada del ojo y está compuesto por un tejido fino que tiene la apariencia de líneas radiales y capas, cuando se le examina de cerca. Esto crea un patrón único en cada persona que es el mismo durante toda su vida.
- **Cara**. Tiene la desventaja de que las características de la cara varían por la edad, el maquillaje, el peinado, las gafas, la postura y las condiciones de luz, por lo cual en algunos casos no es útil para verificar la identidad. Los métodos para el reconocimiento del rostro utilizan principalmente estas cuatro técnicas: geometría facial, patrones de la piel, temperatura del rostro y sonrisa.
- **Geometría de la mano**. Los sistemas que manejan la mano como método de identificación toman dos imágenes de esta parte del cuerpo, por arriba y por el lado, así como una de la

palma con la ayuda de un espejo. Para la identificación se verifica el largo y el ancho, así como las diversas curvaturas y detalles de la mano.

1.3.3.1. Validación de la identidad utilizando biometría

En el proceso de autenticación (o verificación) los rasgos biométricos se comparan solamente con los de un patrón ya guardado, este proceso se conoce también como uno-a-uno (1:1). Este procedimiento implica conocer presuntamente la identidad del individuo a autenticar, por lo tanto, dicho individuo debe presentar algún tipo de credencial, la cual puede ser validada o no después de la autenticación biométrica.

Para la función de identificación los rasgos biométricos se comparan con los de un conjunto de patrones ya guardados, este proceso se conoce también como uno-para-muchos (1: N). Este procedimiento implica no conocer la identidad presunta del individuo ya que la nueva muestra de datos biométricos es tomada del usuario y comparada una a una con los patrones ya existentes en el banco de datos registrados. El resultado de este proceso es la identidad del individuo, mientras que en el proceso de autenticación es un valor verdadero o falso.

El procedimiento de autenticación o verificación biométrica es más rápido que el de identificación biométrica, sobretodo cuando el número de usuarios (N) es elevado, esto se debe a que la necesidad de procesamiento y comparaciones es más reducida en el proceso de autenticación. Por esta razón, es habitual usar autenticación cuando se quiere validar la identidad de un individuo desde un sistema con capacidad de procesamiento limitada o se quiere un proceso muy rápido.

1.3.3.2. Tarjetas inteligentes y verificación biométrica

Debido al uso creciente de las tarjetas inteligentes y la existencia de métodos cada vez más seguros y rápidos de identificación biométrica, colocar en estas tarjetas los rasgos de las personas con fines de autenticación e identificación es cada vez más masivo y barato.

La verificación 1:1 del posible titular de una tarjeta inteligente, reside en estos momentos, dentro de la propia tarjeta, lo que asegura que sea cual fuera la característica biométrica utilizada, ésta no saldría nunca de la misma. A este proceso se le conoce como *Match On Card* (en lo adelante MOC, por sus siglas en inglés), lo cual garantiza la seguridad de los datos biométricos resguardándolos de posibles ataques.

En estos casos la tecnología empleada incluye un lector que convierte la característica biométrica de la persona y se la entrega a la tarjeta, la cual verifica, usando los datos que tiene almacenados, si ambas corresponden a la misma persona.

1.3.4. PKI, certificados e identidad digital

La identidad digital es un concepto novedoso y se define como el conjunto de rasgos que caracterizan a una persona en un medio de transmisión digital y tiene la misma validez que la identidad física siempre que se implemente usando medios y métodos correctos. Este tipo de proceso es muy utilizado en la actualidad y está estrechamente relacionado con la utilización de tarjetas inteligentes.

Un certificado digital es una credencial que proporciona información acerca de la identidad de una entidad o persona y la certifica como tal en el ámbito de las relaciones digitales.

El acrónimo PKI deriva de *"Public Key Infrastructure"* (Infraestructura de Clave Pública) y es la forma común de referirse a un sistema complejo necesario para la gestión de certificados digitales y aplicaciones de firma digital.

Una Infraestructura de Clave Pública bien construida debe proporcionar:

- **Autenticidad.** La firma digital tendrá la misma validez que la manuscrita.
- **Confidencialidad** de la información transmitida entre las partes.
- **Integridad.** Detectar si un documento firmado ha sido manipulado.
- **No Repudio** de un documento firmado digitalmente.

1.3.4.1. Criptografía asimétrica

La criptografía asimétrica es el método criptográfico que utiliza un par de claves para el envío de mensajes. Las dos claves pertenecen a la misma persona a la que se ha enviado el mensaje.

Una clave es pública y se puede entregar a cualquier persona, la otra clave es privada y el propietario debe guardarla de modo que nadie tenga acceso a ella. El remitente usa la clave pública del destinatario para cifrar el mensaje, y una vez cifrado, sólo la clave privada del destinatario podrá descifrar este mensaje

1.3.4.2. Firma digital

La firma digital es un método criptográfico que asocia la identidad de una persona al mensaje o documento. En función del tipo de firma, puede además, asegurar la integridad del documento o mensaje.

Una firma digital tiene dos características principales:

- Sólo puede ser generada por el poseedor de la clave privada y puede ser verificada por cualquiera que conozca la clave pública del firmante.
- Es dependiente del documento a firmar, la firma digital de un documento no puede emplearse para firmar otro documento.

El proceso de generación de una firma digital consiste en dos pasos:

- Se emplea una función Hash para generar un resumen, de tamaño fijo, del documento.
- Se cifra el resumen empleando la clave privada del usuario.

1.3.4.3. Certificados digitales

Un certificado digital es un documento mediante el cual una entidad confiable garantiza la vinculación entre la identidad de un sujeto y su clave pública. De esta manera se puede asegurar que una clave pública pertenece a un usuario dado, resolviendo uno de los problemas de seguridad que se pudieran generar del uso de las técnicas antes mencionadas.

Un certificado digital al menos debe contener la siguiente información:

- Identidad del usuario (nombre, apellidos, número identidad permanente, Foto, etc.).
- Clave pública del usuario.
- Período de validez del certificado.
- Identidad de la Autoridad Certificadora (entidad que emite el certificado).
- Firma digital del certificado generada por la Autoridad Certificadora.

1.3.5. Tarjetas inteligentes con motivos de identificación

Las tarjetas inteligentes como mecanismos de identificación son utilizadas actualmente en varios países del mundo, en los cuales su forma de uso es variable, dependiendo en gran medida de las leyes que para ello hayan sido trazadas por los diferentes gobiernos.

En la mayoría de los casos presentan diseños diferentes, adaptados a las necesidades propias de los diferentes países, en su generalidad cuentan con un número de identificación, el cual es utilizado como un mecanismo administrativo para asociar al portador de la tarjeta con una serie de actividades en diversas áreas.

1.3.5.1. Utilización en el mundo

La implantación de sistemas de identificación nacional basados en la utilización de tarjetas inteligentes ha permitido que se reúnan una serie de experiencias, las cuales constituyen un buen punto de partida para aquellos países que pretendan aplicar esta tecnología. En este epígrafe analizaremos algunos de los ejemplos más significativos[4] (Enrrique Vasquez Gallo, 2007).

1.3.5.2. DNI electrónico en España [5]

El Real Decreto 1553/2005, de 23 de diciembre, regula la expedición del Documento Nacional de Identidad y sus certificados de firma electrónica. La tarjeta de identidad de España es una tarjeta de tamaño ID-1, construido de policarbonato que incorpora un chip con información digital. Los datos mostrados son: en el frente, la información relativa a la tarjeta y el ente emisor de la misma, nombres y apellidos; en su reverso la fecha de nacimiento, nombres y apellidos, etc., país, firma, fechas de emisión y expiración.

La tarjeta puede ser usada de diferentes maneras, para identificarse presencialmente o de forma electrónica a través de los certificados contenidos en ella, así como puede también ser usada para la firma electrónica de documentos. Puede ser usada en el Sector Privado para: acceder a sitios WEB, firma de contratos, verificación de autenticidad de documentos y en el *Sector Público* para la declaración anual de impuestos, interacción con la Administración. Pública para la obtención de formularios, servicios en línea, obtener registros criminales.

La tarjeta contiene un certificado de firma electrónica cuya vigencia es de 2,5 años o hasta la renovación física de la tarjeta criptográfica. El plazo normal de validez de las tarjetas es 5 años, 10 años o permanente según la edad del solicitante. La renovación de los certificados de usuario, con o sin renovación de la tarjeta, implica el cambio de claves.

1.3.5.3. DNI electrónico en Finlandia

Finlandia fue el primer país europeo en emitir tarjetas de identidad electrónica a finales de 1999. La tarjeta FINEID lleva impreso en el frente la información relativa a la tarjeta y el ente emisor de la misma y en el reverso: identificación de la entidad emisora, datos de la tarjeta y del ciudadano en formato OCR, código de barras. Sin embargo, salvo el nombre, los otros datos personales no se almacenan electrónicamente en la tarjeta.

4 Ver Anexo 3 ¨ Sistemas nacionales de identificación que utilizan tarjetas inteligentes ¨.
5 Ver Anexo 4 ¨ DNIe España.

Cada tarjeta tiene dos certificados, uno para autenticación y cifrado y el otro para firma electrónica. Cada uno se utiliza con un PIN diferente. Si el solicitante lo desea, puede incluir información médica en la tarjeta FINEID, con lo cual no necesita tener una tarjeta sanitaria separada (denominada KELA).

A pesar de que la tarjeta FINEID lleva en funcionamiento desde 2000 y que con ella pueden acceder los ciudadanos a gran variedad de servicio (el PRC ha establecido la marca para identificarlos fácilmente) y las empresas, su crecimiento está siendo muy lento. Por ejemplo, desde 2000 hasta mediados de 2003 sólo se habían emitido 16.000 tarjetas. (Como referencia, la población total del país en 2003 era de 5,2 millones de personas).

Además de los certificados incluidos en las tarjetas de identidad, el *Population Register Centre* (PRC) emite certificados a los ciudadanos para utilizar en tarjetas Visa Electron (estas tarjetas, que dan los mismos servicios que la de identidad y además servicios bancarios, se solicitan en una sucursal bancaria) y certificados para utilizar con la tarjeta SIM de un teléfono móvil (el ciudadano debe comprar la tarjeta SIM al operador móvil, y registrarla en el departamento de policía).

1.3.5.4. DNI electrónico en Estonia [6]

En enero de 2002, aprobado por el parlamento, comenzó la emisión de tarjetas de identidad electrónica. Son tarjetas de tamaño ID1 construidas de policarbonato, que cumplen con las normas ISO 7810, 7816 e ICAO.

Chip 16k, criptografía RSA 2048 bits. La tarjeta lleva impreso en el frente el nombre del ciudadano, código de identificación nacional, fecha de nacimiento, sexo, ciudadanía, número de serie de la tarjeta, fecha de validez, fotografía y la firma digital; en el reverso: lugar de nacimiento, fecha de emisión, permiso de residencia u otras informaciones si es necesario.

Cada tarjeta contiene dos certificados cualificados con códigos PIN diferentes: uno se usa para autenticación y cifrado, el otro, para firmas digitales con validez legal equivalente a la firma manuscrita. Los certificados son válidos durante 1100 días (aproximadamente tres años). La única información personal que consta en los certificados es el nombre del titular y su número de identificación nacional único, ambos considerados de acceso público en Estonia.

El certificado de autenticación contiene, además del nombre y número de identificación del titular, una dirección de correo electrónico asignada por la Administración con el formato "nombre.apellido_NNNN@eesti.ee". Esta dirección está pensada para comunicaciones entre ciudadano y Administración, aunque puede usarse también entre particulares. El titular puede cifrar

[6] Ver Anexo 5 DNIe Estonia.

sus correos electrónicos y/o firmarlos digitalmente con la clave correspondiente a su certificado de autenticación, aunque sin el compromiso legal que implica el certificado de firma.

La tarjeta puede ser utilizada para servicios públicos y privados, banca electrónica, pago de impuestos.

1.4. Sistemas de Administración de Tarjetas Inteligentes y Aplicaciones

Con el avance de las tecnologías de la información y la necesidad creciente de seguridad en el traspaso y manipulación de la misma, aparecen en el mundo sistemas de identificación de usuarios y procesos cada vez más exigentes en cuanto a la verificación real de la identidad de su portador. Con el transcurso de los años han sido utilizados varios mecanismos que van desde la verificación más simple empleando códigos de acceso o contraseñas hasta una de las más reciente y aún novedosa: las tarjetas inteligentes. Estas tarjetas están dotadas de modernos chips en los cuales se almacena información y cuentan con complejos procesos de acceso que garantizan la seguridad de los datos registrados. Entre los múltiples usos que ellas tienen se destaca su utilización como tarjetas de identificación, en las cuales el chip cuenta con toda la información necesaria para corroborar si su portador es quien dice ser o si puede realizar una operación determinada.

A partir de la utilización de tarjetas inteligentes en una institución, ya sea gubernamental o no, surge la necesidad de tener cierto control sobre las mismas, así como conocer los servicios a los cuales tienen acceso o solicitan tener sus propietarios. Tomando esta situación como punto de partida, es que mucho productores han creado los Sistemas de Administración de Tarjetas y Aplicaciones (CAMS) y los Sistemas de Administración de Tarjetas (CMS), los cuales, en la mayoría de los casos, son dedicados a negocios específicos, aunque los más avanzados son adaptables a diferentes instituciones a través de reglas de uso que pueden ser definidas a la hora de ponerlos en funcionamiento.

1.4.1. Definiciones

Un **CAMS** (Sistemas de Administración de Tarjetas y Aplicaciones) es una pieza de software que actúa como motor central en un sistema de emisión de tarjetas inteligentes. Su función principal consiste en garantizar la gestión de las tarjetas emitidas, ofreciendo información sobre las diferentes etapas en las que se encuentran durante su ciclo de vida, y permitiendo la administración de las aplicaciones que contendrán.

CMS: (Sistemas de Administración de Tarjetas): en la mayoría de los casos, son dedicados a negocios específicos, aunque los más avanzados son adaptables a diferentes instituciones a través de reglas de uso que pueden ser definidas a la hora de ponerlos en funcionamiento. Su función es prácticamente la misma que la de los CAMS y por tanto sus características son muy similares, sólo que a diferencia de los primeros, éstos no implementan la administración de aplicaciones. Pueden formar parte de sistemas de identidad con funcionalidades muy básicas

Esta diferencia esencial trae consigo la necesidad de un estudio del ambiente de desarrollo y despliegue del futuro sistema de administración que se aplicará para el control de las tarjetas inteligentes, ya que el primero brindaría, además de las funciones de gestión de las tarjetas, un mecanismo de control de las aplicaciones que se integren a las tarjetas, sin embargo el segundo sería una solución bastante acertada a la hora de implementar sistemas de identificación con tarjetas inteligentes un poco más sencillos y con funciones mucho más básicas, en las cuales las tarjetas no se utilicen para tener acceso a múltiples aplicaciones.

CAMS	CMS
Cuenta con módulos especializados (recepción de tarjetas vírgenes, personalización y entrega al usuario).	
Pieza de Software	
Administración y monitoreo del ciclo de vida de las tarjetas: se controla el estado; de los definidos por el sistema, se encuentra una tarjeta determinada, se le asignan o eliminan servicios a los que puede acceder, se gestionan las solicitudes de los usuarios derivadas de su utilización.	
Son utilizados en proyectos de mayor envergadura donde una misma tarjeta puede acceder a un conjunto variable de aplicaciones y servicios.	Se implantan de forma independiente sólo en aquellas soluciones donde la utilización de las tarjetas es limitada a servicios más sencillos.

Tabla 1.1 Comparación de CAMS y CMS.

A continuación caracterizaremos algunos de los CAMS y CMS existentes en el mercado, con el objetivo de evaluar sus características, ventajas y desventajas, y poder modelar mejor el Sistema vinculado a la solución de cédula electrónica de la República Bolivariana de Venezuela.

1.4.2. Sistemas y soluciones analizados

1.4.2.1. SafeNet, MyID Card Management System

Sistema que se utiliza para emitir, administrar y apoyar el uso las tarjetas inteligentes en las empresas que las utilizan como credenciales.

Características:

- Basado en la Web. Hace fácil despliegue y uso en toda la empresa.
- Completamente personalizable. Se adapta a las políticas de seguridad corporativa propias de la institución donde se utilice y sus reglas de negocio.
- Administración desde la web de todas las aplicaciones y usos de las tarjetas o tokens de identificación, lo que agiliza los trámites sobre las mismas.
- Permite que desde una red pública se pueda tener acceso al servidor sin que esto implique un peligro para la seguridad de la información (Cifrado VPN).
- PKI independiente.
- Gestión de credenciales digitales: maneja las claves, los certificados y datos almacenados en las tarjetas o fichas.
- Reduce la necesidad de uso de soporte debido a PINs olvidados: incluye soporte y gestión para resolución de problemas de este tipo.
- Emisión, administración y soporte de tarjetas inteligentes y tokens de identificación (iKey) USB en todas las áreas de la empresa.

El despliegue y gestión de tarjetas inteligentes y tokens de identificación digitales puede ser una tarea muy engorrosa si no se cuenta con las herramientas necesarias para ello. SafeNet CMS reduce la complejidad asociada al despliegue de un sistema de identificación de tarjetas inteligentes pues brinda una serie de servicios para ello, como son:

- Edición electrónica y personalización de las tarjetas o tokens: las organizaciones tienen un seguro sistema basado en la Web para generar, emitir y cargar las credenciales digitales de un servidor central o distribuido a muchos usuarios.
- Administrar el despliegue de tarjetas y tokens USB: CMS SafeNet da a los usuarios de la empresa la facultad de solicitar y renovar fácilmente sus credenciales digitales personales. Un usuario accede a la SafeNet CMS publicado a través de un sitio Web y recibe una lista de opciones de menú que están disponibles de acuerdo a las políticas de la empresa. La solicitud de certificados, cambio de contraseña de la tarjeta y la renovación de certificados, etc. están disponibles a través de la Web.
- SafeNet CMS Help Desk: conjunto de herramientas para el soporte a las tarjetas y usuarios del sistema que permite realizar el monitoreo de las tarjetas en uso, bloqueo y desbloqueo en los casos necesarios; ya sea por exceso de intentos de entrada con contraseñas incorrectas o en

caso de anuncio de pérdida. Además prevé la emisión de nuevos objetos en caso de pérdidas o deterioros, así como el recambio de contraseñas en caso de olvido. (Safe Net, 2008)

1.4.2.2. RSA, Card Manager

Esta compañía presenta sistemas e infraestructuras de gestión de tarjetas inteligentes para gobiernos y empresas de todo el mundo, por lo que constituye uno de los productores más difundidos en el mercado y a su vez más comprados por las instituciones que necesitan este tipo de servicios.

RSA Card Manager, es un sistema de gestión de tarjetas que permite a las entidades usuarias gestionar el ciclo de vida de las credenciales de identificación almacenadas en cualquier dispositivo con chip inteligente, incluidas tarjetas inteligentes y generadores de clave USB. En un terreno más amplio, RSA Card Manager crea una plataforma para que las empresas y gobiernos pongan en marcha soluciones de gestión de accesos e identidades (IAM) basadas en tarjetas inteligentes.

Esta solución cubre todo el ciclo de vida de las credenciales de tarjeta inteligente, como la emisión, sustitución y cancelación de tarjetas y credenciales, así como ofrece facilidades para la administración de tarjetas de identificación y de pequeñas aplicaciones.

Para facilitar aún más la gestión de las credenciales, el software RSA ® Card Manager, incorpora un portal de administración que permite a los usuarios realizar varias operaciones como desbloquear tarjetas, solicitar credenciales temporales en caso de pérdida y tramitar renovaciones de certificados digitales. Además proporciona un alto grado de flexibilidad, ya que se integra fácilmente a varios sistemas de este tipo que ya se utilizan en el mundo.

El sistema cuenta con los módulos que se describen a continuación:

- El RSA Smart Cards System; tienen la capacidad de desplegar el subsistema PIV (tarjetas para la Verificación de Identidad Personal), que se ocupa de los accesos tanto físicos como lógicos a los diferentes recursos.
- RSA Card Manager actúa como el subsistema de gestión y emisión de tarjetas PIV que permite a los clientes gestionar el ciclo de vida de las credenciales y las tarjetas inteligentes.
- RSA Card Manager y RSA Certificate Manager permiten implantar un subsistema eficaz de control de accesos que también hace posible almacenar credenciales en una tarjeta inteligente de forma segura, aprovechar los certificados digitales para controlar los accesos lógicos y gestionar el uso habitual de certificados digitales.

La solución de gestión de tarjetas inteligentes de RSA Security está basada en una alianza estratégica con Intercede Group, fabricante líder de software para tarjetas inteligentes y gestión de

identidades. Como parte de esta alianza, RSA Security compró la tecnología MyID™ de Intercede, una de las más utilizadas a nivel mundial, lo que le permitirá ofrecer un alto grado de integración entre RSA Card Manager y otras soluciones de la compañía para gestión de accesos e identidades. (RSA Security, 2008)

1.4.2.3. ActivID™ Card Management System

ActivID Card Management System proporciona una solución completa y flexible para la gestión de la emisión y administración, necesarios para el éxito en el despliegue de tarjetas inteligentes. ActivID CMS gestiona las tarjetas, así como los datos digitales incluyendo credenciales PKI y certificados relacionados con ellas a lo largo de todo su ciclo de vida.

Principales características:

Administración y configuración:

- Arquitectura flexible que garantiza la seguridad a través grupos y roles.
- Gestión de solicitudes a través de centros de llamadas.
- Flujos de trabajo adaptables a múltiples escenarios de despliegue.
- Disponibilidad de realización de auditorias que recogen las actividades y eventos del sistema para la expedición de informes de funcionamiento.
- Emisión de Tarjetas Inteligentes.
- Soporte plenamente distribuido (supervisado o libre-asistido) por lotes, de servicios y modelos de emisión.
- Soporte seguro y distribuido después de la emisión y expedición de tarjetas de la gestión de su ciclo de vida, sin problemas de distribución de claves.
- Cuenta con plena capacidad para foto y una tarjeta de captura de impresión.

Asistencia a los usuarios:

- Recuperación de datos y credenciales fácilmente.
- Administración de contenidos: suspender/reanudar y revocar certificados PKI, añadir/quitar y actualizar datos de la tarjeta siempre que sea necesario durante la vida de la misma.
- Múltiples servicios de la tarjeta: en línea, fuera de línea, desbloqueo de PIN, revocación y reciclaje de las tarjetas inteligentes.

Beneficios:

Seguridad:

- Proporciona una amplia seguridad antes y después de la emisión de las tarjetas; utilización de ActivIdentity, tecnología adoptada por el Departamento de Defensa de EE.UU. para emitir 4,3 millones de tarjetas de acceso común a personal militar.

Interoperabilidad:

- Facilita el despliegue, aprovecha las inversiones existentes y minimiza la obsolescencia por el apoyo a la más amplia gama de las tarjetas inteligentes, directorios, bases de datos, autoridades de certificación, sistemas de suministro de identidad, y sistemas de control de acceso físico.
- Usabilidad:
- Facilita la administración con una intuitiva interfaz Web y la delegación de capacidades. El asistente de autoservicio optimiza la experiencia del usuario final y minimiza llamadas de asistencia.
- Capacidad de Despliegue:
- Minimiza el tiempo y el coste de integrar y configurar. Producto comercial con la utilidad de configuración completa. (ActivIdentity, 2008)

1.4.2.4. Siemens Identity Management and Smart Cards

Esta solución puede ser utilizada en diverso propósitos:

- Cifrado (VPN)
- Acceso remoto (a través de LAN).
- La firma de los documentos electrónicos.
- Acceso a PCs.
- Acceso a las aplicaciones existentes.
- Control de acceso a aplicaciones Web.
- Control de acceso a edificios.
- Control del tiempo y la captación de datos.
- Pago seguro.

Ofrece una completa e integrada gama de servicios que comprenden personalización de tarjetas inteligentes, sistemas operativos de tarjeta inteligente, gestión de contraseña y una interfaz para

aplicaciones de PKI. Garantiza la seguridad e integridad de los recursos que maneja a través del uso de roles que controlan el acceso de los usuarios.

Beneficios:

Aumento de la seguridad física:

- Sólo las personas autorizadas se les permite el acceso a sitios específicos, como los edificios, laboratorios, salas de operación, centros de computación o estacionamientos.
- Los titulares de la tarjeta tratando de obtener acceso no autorizado a los puntos de acceso pueden ser fácilmente identificados.

Aumento de la lógica de seguridad:

- Sólo las personas autorizadas tienen acceso a estaciones de trabajo y servidores, y la información sensible.
- Las firmas digitales que se ofrecen, se pueden utilizar en todas las aplicaciones.
- Utiliza cifrado VPN.
- Flujos de trabajo más eficientes. (Siemens, 2008).

1.4.2.5. Tabla resumen [7]

Características	Sistemas analizados			
	Safe Net	RSA Security	ActivId	Siemens
Control de Inventario de tarjetas vírgenes.	x	x	x	x
Personalización de tarjetas vírgenes.	x	x	x	x
Administración del ciclo de vida de las tarjetas.	x	x	x	x
Gestión a través de la Web.	x	x	x	x
Sistema personalizable.	x	x	x	
Infraestructura de Llave Pública (PKI).	x		x	x
Tokens de identificación (iKey) USB.	x	x		
Opera con varios tipos de tarjetas.			x	
Integración a sistemas similares en funcionamiento		x		

Tabla 1.2 Resumen simplificado de características de los sistemas analizados.

[7] Ver Anexo 6 ¨ Resumen de las Características de Sistemas de Administración de Tarjetas Inteligentes ampliada ¨.

1.5. Tendencias tecnológicas; metodologías y herramientas

1.5.1. Metodología utilizada

El Proceso Racional Unificado (RUP, por sus siglas en inglés *Rational Unified Process*) es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado (UML), constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos. RUP es en realidad un refinamiento realizado por Rational Software del más genérico Proceso Unificado. Proporciona una guía en el orden de las actividades de un equipo, dirige las tareas individuales de los desarrolladores, especifica qué productos deberían ser desarrollados y ofrece criterios para monitorear y medir los productos y actividades del proyecto.

RUP es una metodología que se caracteriza por:

- Proporcionar una guía para ordenar las actividades de un equipo.
- Dirigir las tareas de cada desarrollador por separado y del equipo como un todo.
- Especificar los artefactos que deben desarrollarse.
- Ofrecer criterios para el control y la medición de los productos y actividades del proyecto.

En fin, RUP es una forma disciplinada de asignar tareas y responsabilidades (quién hace qué, cuándo y cómo). Pretende implementar las mejores prácticas en Ingeniería de Software:

- Desarrollo iterativo.
- Administración de requisitos.
- Uso de arquitectura basada en componentes.
- Control de cambios.
- Modelado visual del software.
- Verificación de la calidad del software.

1.5.2. Lenguaje de modelación visual

El Lenguaje Unificado de Modelado (UML, por sus siglas en inglés, *Unified Modeling Language*) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; aún cuando todavía no es un estándar oficial, está apoyado en gran manera por el OMG (*Object Management Group*). Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema de software. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocios y funciones del sistema, y

aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables**.**

1.5.3. Microsoft.NET como plataforma de desarrollo

Microsoft.NET es el conjunto de nuevas tecnologías en las que Microsoft ha estado trabajando durante los últimos años con el objetivo de obtener una plataforma sencilla y potente para distribuir el software en forma de servicios que puedan ser suministrados remotamente y que puedan comunicarse y combinarse unos con otros de manera totalmente independiente de la plataforma, lenguaje de programación y modelo de componentes con los que hayan sido desarrollados.

El ambiente de .NET es una infraestructura sobre la que se reúne todo un conjunto de lenguajes y servicios que simplifican enormemente el desarrollo de aplicaciones. Mediante esta herramienta se ofrece un entorno de ejecución altamente distribuido, que permite crear aplicaciones robustas y escalables. Organiza toda la funcionalidad del sistema operativo en un espacio de nombres jerárquico de forma que a la hora de programar resulta bastante sencillo encontrar lo que se necesita.

Los principales componentes de este entorno son:

- Lenguajes de desarrollo.
- Biblioteca de clases de .NET.
- CLR (Common Language Runtime).

Ventajas:

- Código administrado**:** El CLR realiza un control automático del código para que este sea seguro, es decir, controla los recursos del sistema para que la aplicación se ejecute correctamente.
- Interoperabilidad multilenguaje**:** El código puede ser escrito en cualquier lenguaje compatible con .NET ya que siempre se compila en código intermedio o Microsoft Intermediate Lenguaje (MSIL).
- Compilación just-in-time**:** El compilador JIT (Just In Time, nombre que recibe ese tipo de compilación porque se realiza en tiempo de ejecución), incluido en el Framework, compila el código intermedio (MSIL) generando el código máquina propio de la plataforma. Se aumenta así el rendimiento de la aplicación al ser específico para cada plataforma.
- Despliegue**:** Por medio de los ensamblados resulta mucho más fácil el desarrollo de aplicaciones distribuidas y el mantenimiento de las mismas. El Framework realiza esta tarea de

forma automática mejorando el rendimiento y asegurando el funcionamiento correcto de todas las aplicaciones.

1.5.4. Lenguaje de programación C#

C# es un lenguaje de programación orientado a objetos desarrollado por Microsoft y estandarizado como parte de su plataforma .NET. Aunque para la plataforma .NET es prácticamente posible programar en cualquier lenguaje, el C# es el lenguaje de propósito general diseñado por Microsoft para ser utilizado en ella, por lo que programar usando C# es mucho más sencillo e intuitivo que hacerlo con cualquiera de los otros.

Su sintaxis básica deriva de C/C++ y utiliza el modelo de objetos de la plataforma .NET el cual es similar al de Java aunque incluye mejoras derivadas de otros lenguajes. C# fue diseñado para combinar el control a bajo nivel de lenguajes como C y la velocidad de programación de lenguajes como Visual Basic.

Características principales:

- **Sencillez:** C# elimina muchos elementos que otros lenguajes incluyen y que son innecesarios en .NET. Por ejemplo:
- El código escrito en C# es autocontenido, lo que significa que no necesita de ficheros adicionales al propio fuente, como ficheros de cabecera.
- El tamaño de los tipos de datos básicos es fijo e independiente del compilador, sistema operativo o máquina para la cual se compile, lo que facilita la portabilidad del código.
- **Orientación a componentes:** La propia sintaxis de C# incluye elementos propios del diseño de componentes que otros lenguajes tienen que simular mediante construcciones más o menos complejas. Es decir, la sintaxis de C# permite definir cómodamente propiedades (similares a campos de acceso controlado), eventos (asociación controlada de funciones de respuesta a notificaciones) o atributos (información sobre un tipo o sus miembros).
- **Eficiente:** Como principio, en C# el código incluye numerosas restricciones para asegurar su seguridad y no permite el uso de punteros. Sin embargo, y a diferencia de Java, en C# es posible saltarse dichas restricciones manipulando objetos a través de punteros, con sólo marcar regiones de código como inseguras (modificador unsafe). En estos bloques pueden usarse los punteros de forma similar a cómo se hace en C++, lo que cual resulta vital en situaciones donde se necesite eficiencia y mayor velocidad de procesamiento.

1.5.5. Oracle como sistema gestor de base de datos

Oracle es considerado uno de los Sistemas Gestores de Bases de Datos más completos en la actualidad. Representa una opción factible y potente para la información que se maneja en sistemas que generan un gran volumen de información, al presentar un alto rendimiento. Es altamente escalable permitiendo grandes demandas de usuarios y manteniendo una alta capacidad de adaptarse a cambios bruscos de demanda.

El Oracle 10gR2 permite la utilización de los Clúster de Aplicaciones Reales (RAC) con la tecnología de disco compartido. Los recursos, servidores y almacenamiento pueden ser administrados como una entidad única dentro del ambiente del clúster. A medida que se agregan recursos, el Clúster de Aplicaciones Reales puede utilizarlos, esto asegura un costo total de propiedad más bajo, de esta forma no se necesita comprar nuevo hardware con los requerimientos necesarios.

Otras de sus principales características son su gran capacidad de almacenamiento y de réplica, máxima seguridad, administración simplificada, soporte de transacciones y facilidades en las tareas de recuperación y respaldo.

1.6. Conclusiones

En este capítulo hemos hecho un análisis a profundidad tanto de las tarjetas inteligentes y su utilización con motivo de identificación, como de los Sistemas de Administración de Tarjetas Inteligentes y Aplicaciones más utilizados, de esta forma hemos podido llegar a importantes conclusiones acerca del sistema que pretendemos modelar en este trabajo.

Para la elaboración del sistema se plantea la utilización de un conjunto herramientas de trabajo que facilitarán a la aplicación un entorno sencillo, confiable y seguro; como son el gestor de base de datos Oracle, un ambiente visual basado en la tecnología .NET, teniendo como lenguaje de programación el C# y siguiendo en todo momento la metodología de desarrollo de software RUP y utilizando el lenguaje UML para la modelación visual del problema.

Capítulo 2 : Características del Sistema

2.1. Introducción

En este capítulo se describe la propuesta de solución CAMS que se encargará del control del ciclo de vida de la cédula de identidad electrónica de la República Bolivariana de Venezuela, para ello se identifican los procesos del negocio relacionados con el objeto de estudio. Debido a la poca estructuración de estos procesos, se requiere definir los conceptos que se pueden agrupar en un Modelo de Dominio que permita diseñar un sistema correcto. Además, se determinan los requisitos funcionales y no funcionales con los que debe cumplir la solución y se especifican los Casos de Uso y sus relaciones con los actores.

2.2. Objeto de estudio

2.2.1. Objetivos estratégicos de la organización

La Oficina Nacional de Migración y Extranjería (ONIDEX) es un órgano adscrito al Ministerio del Poder Popular para Relaciones Interiores y Justicia (MPPIJ) a través del cual el Estado Bolivariano garantiza el derecho a la identidad de todos los ciudadanos, la regulación del flujo migratorio y el control de extranjeros en la República Bolivariana de Venezuela.

Con la introducción de la nueva cédula de identidad electrónica, dicha organización tiene como objetivo fortalecer la seguridad del Sistema Nacional de Identificación; centralizando el proceso de emisión de la cédulas de identidad e implementando nuevas mediadas de seguridad que en la actualidad no se aplican debido a que el material utilizado para la confección de los documentos actuales no lo permite.

El nuevo documento de identidad trae cambios en algunos de los procesos actuales relacionados con la cedulación, surgiendo la necesidad de que el Sistema de Nacional de Identificación cuente con un subsistema que brinde la posibilidad de gestionar y administrar todos los aspectos relacionados con la nueva cédula de identidad: operaciones realizadas con ella, gestione su ciclo de vida y servicios a las cuales tendrá acceso el usuario mediante su utilización.

2.2.2. Flujo actual de los procesos involucrados en el campo de acción

El campo de acción y objeto de estudio de nuestro trabajo están estrechamente relacionados con el proceso de Identificación, dicho proceso está determinado por tres tipos de trámite de cedulación.

Cada tipo de trámite está definido por un conjunto de pasos por los cuales tiene que transcurrir para ser concluido satisfactoriamente.

Tipo de Trámites de Cedulación		
Original	**Duplicado**	**Renovación**
La persona no ha sido cedulada antes por el sistema. Se captan sus datos e imágenes en el sistema y se le confecciona una nueva cédula.	La persona ha sido cedulada antes por el sistema, el tiempo de caducidad de la cédula no ha expirado y es necesario una reimpresión o reexpedición de la cédula por deterioro o extravío de la original. Se hace una copia de la cédula manteniendo intactos los datos anteriores pero con renovación de imágenes	La persona ha sido cedulada por el sistema y el tiempo de caducidad de la cédula ha expirado o cuando la persona realiza alguna actualización de la información contenida en el documento. El resultado final consiste en la actualización de los datos en el sistema y la confección de la cédula con los datos actualizados.
Pasos necesarios para los tipos de trámites de cedulación[8]		
Captación de datos, Captación de imágenes, Impresión Planilla de Control, Aprobación de Datos, Aprobación de Documentos, Aprobación Imágenes, Asignación de Número de Cédula, Impresión de Cédula y Entrega de Cédula.	Captación de datos, Captación de imágenes, Impresión Planilla de Control, Aprobación Imágenes, Impresión de Cédula y Entrega de Cédula.	Captación de Datos, Captación de Imágenes, Impresión Planilla de Control, Aprobación de Datos, Aprobación de Documentos, Aprobación Imágenes, Impresión de Cédula y Entrega de Cédula.

Tabla 2.1 Tipos de trámites de cedulación.

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

El proceso de cedulación en la República Bolivariana de Venezuela en un principio se diseñó sólo para suplir las necesidades de identificación que tenía la sociedad venezolana, su principal objetivo fue hacer más sencilla la obtención de un documento que permitiera a los venezolanos y venezolanas identificarse; se desplegaron entonces por todo el país oficinas y móviles de cedulación en los cuales se desarrollaba todo el proceso, incluso el almacenamiento y personalización de los documentos.

[8] Ver Anexo 7: Pasos necesarios para los tipos de trámite de cedulación.

Este gran levantamiento se desarrolló en el Marco de la Misión Identidad [9](Abril de 2004) y permitió que miles de ciudadanos; olvidados y excluidos jurídica y socialmente, participaran en el importante proceso electoral que se desarrollaba en el país y que culminó con la victoria del Referendo Constitucional propuesto por el Presidente Hugo Chávez.

Por la urgencia y la necesidad de un documento de identificación para los miles de ciudadanos que carecían de él, se utilizó para la confección de las cédulas el papel; un material que propicia las falsificaciones y deterioro de las mismas.

Con la introducción de un documento de identidad electrónico, los procesos asociados a la cedulación cambian: el proceso de impresión no se puede realizar usando la tecnología actual (el cuerpo de la tarjeta está compuesto por policarbonato, este sólo puede ser impreso usando láser), la incorporación de un chip (cambia la forma en que los datos son almacenados, cambia que datos son visibles y cuales no). La garantía de la unicidad de la identidad y de la eficacia de los procesos asociados hace que la nueva cédula puede ser utilizada para que se accedan a servicios que brindan diferentes instituciones públicas o privadas; mejorando la eficiencia de la comunicación entre el estado y los ciudadanos.

La seguridad y control de todos los procesos relacionados con la tarjeta, empieza a tomarse como uno de los principales factores a tener en cuenta, en temas de seguridad, el resultado final es tan fuerte como tan débil es uno de sus eslabones, por tal motivo se hace necesario contar con algunos elementos que no están presentes en el proceso de cedulación actual en el cual se detentan las siguientes limitantes:

- **No existe un control centralizado de los insumos.** El papel con que se imprimen las cédulas se almacena en las oficinas y no se controla en un inventario central, lo que podría provocar algún tipo de descontrol del mismo; por ejemplo el papel sustraído en un robo, puede ser utilizado con fines ilegales para la suplantación de la identidad.
- **El proceso de personalización de las cédulas se desarrolla de forma descentralizada** en las diferentes oficinas y utilizando para ello impresoras comunes, en las cuales no es posible llevar a cabo el proceso de grabado a láser de los datos en la cubierta de la tarjeta ni en el interior del chip.

[9] Misión Identidad: en el periodo entre abril y junio de 2004 se cedularon 5.076.660 venezolanos de ellos 675.398 no tenía cédula.

- **Los insumos son manipulados por demasiadas personas:** No existe una escala de responsabilidades que proteja el nivel de acceso al material de impresión y no se registran las operaciones realizadas sobre éste por los operarios.

Para desarrollar un sistema de identificación basado en la utilización de tarjetas inteligentes es necesaria la integración de diferentes subsistemas y equipamiento externo y para lograr la comunicación entre ellos debe existir alguna forma de conexión segura y de fácil acoplamiento que generalmente brinda un sistema de gestión intermedio. Además de esta funcionalidad, se utiliza para la administración de todo el ciclo de vida de las tarjetas y los servicios que se podrán brindar a través de ellas y **con el cual no contamos.**

2.3. Objeto de automatización

2.3.1. Descripción de los procesos que serán objeto de automatización

El proceso de cedulación, de forma general, cuenta con una serie de pasos o procesos asociados a él diseñados para garantizar la emisión de la actual cédula de identificación[10]. Con la implementación de un nuevo documento de identificación a través del uso de las tarjetas inteligentes; surge la necesidad de reestructuración de estos procesos, así como la inclusión de otros, necesarios para que el Sistema de Identificación Nacional funcione de forma correcta y segura.

La ocurrencia de estos procesos en cada uno de los subsistemas que forman parte del CAMS implica cambios de estado en las tarjetas. Estos representan características de la tarjeta en un momento específico de tiempo. Los cambios de estado son registrados, permitiendo mantener el control de las tarjetas. En la mayoría de los casos los procesos se realizan mediante la relación e integración de varios subsistemas, esta relación será posible a través de interfaces, protocolos de comunicación y canales seguros para el intercambio de información.

A continuación se describen los procesos propuestos como objeto de automatización que de una forma u otra forman parte del CAMS.

2.3.1.1. Proceso de Control de Inventario de Tarjetas

Proceso mediante el cual se registran las operaciones realizadas sobre los lotes de tarjetas que se mantienen dentro de un contexto seguro con acceso limitado (Subsistema de Inventario), entre esas

[10] Cédula impresa directamente en papel, presenta pocas medidas de seguridad y ninguna capacidad de almacenamiento o procesamiento de información.

operaciones está: el ingreso de nuevas tarjetas, las reclamaciones al fabricante en caso de que las mismas se encuentren defectuosas, las solicitudes de lotes nuevos. Debe brindar información sobre lotes completos o tarjetas individuales.

2.3.1.2. Proceso de Asignación de Lotes de Tarjetas a Puntos de Impresión

Proceso que permite asignar a un punto de impresión lotes de tarjetas listas para personalización y el listado de las personas listas para cedulación. Este proceso puede notificar una de dos acciones la primera que la máquina de impresión aceptó el lote y esta lista para la personalización y la segunda que la máquina no está lista para recibir las tarjetas, notificando un error al operario y rechazando el lote. Todas las operaciones de este proceso deben quedar registradas.

2.3.1.3. Proceso de Personalización de Tarjetas

Mediante este proceso en las tarjetas inicializadas se graban los datos de una persona que fueron captados a través del trámite de cedulación, algunos va a la cubierta de la misma (son los que quedan visibles) y otros dentro del chip .Este proceso además incluye las siguientes acciones:

- Se asocia un número de identificación a las tarjetas y se notifica el cambio de estado de éstas.
- La persona pasa a ser cedulada.
- Los certificados digitales utilizados durante la personalización son registrados como “Activos”.

2.3.1.4. Proceso de Envío a Oficina

Se inicia dentro del Subsistema de Inventario y ocurre cuando se asignan a las diferentes oficinas lotes de tarjetas que han sido personalizadas y se encuentran listas para ser entregadas a los usuarios; culmina cuando las oficinas notifican que han recibido correctamente dichos lotes.

El recibo de un lote por parte de las oficinas puede provocar un cambio de estado en las entidades de tarjetas que lo forman. En el Subsistema de Inventario debe registrarse este cambio por lo que se inicia un evento que es recibido por este indicándole o no dicho cambio.

2.3.1.5. Proceso de Entrega al usuario

A través de él los ciudadanos obtienen sus cédulas de identidad electrónica, en este momento las tarjetas dejan de ser controladas por el inventario de las oficinas y pasan a su vida útil como

documento de identificación. Es en este paso donde se realiza la primera comprobación de identidad, o sea se toman las huellas digitales del ciudadano y se comparan con las que se encuentran almacenas en el chip de la cédula, este proceso ocurre completamente dentro de la tarjeta y arroja un resultado real de si la persona es quien dice ser (MOC). Una vez que se produce la entrega se inicia un evento que recibe el Subsistema de Inventario y realiza un cambio de estado en la tarjeta además y en el estado de las personas que reciben las cédulas.

2.3.1.6. Proceso de Reemplazo de Tarjetas

Posibilita reemplazar tarjetas reportadas como perdidas, robadas o con mal funcionamiento. Cuando se manifiesta alguna de estas variantes se revocan los Certificados Digitales adicionándolos a la Lista de Revocación de Certificados y se notifica un cambio de estado en la tarjeta reportada y en la persona que manifiesta el problema. Además, esto implica que necesariamente tengan que realizarse nuevamente el Proceso de Personalización y Entrega al Usuario. Está asociado a los procesos de Renovación y Duplicado del Documento de Identidad.

2.3.1.7. Proceso de Bloqueo/Desbloqueo de Tarjetas

Consiente en bloquear tarjetas reportadas como perdidas, robadas o entregadas a una oficina por alguna persona; asegurando que no sean utilizadas por personas no autorizadas, esto implica que dicha tarjeta y sus certificados digitales quedan temporal o definitivamente inhabilitados y que se notifique este cambio a todas las instituciones que prestan servicios a través de ésta. Además permite desbloquear las tarjetas en caso de que se notifique por la persona o institución que solicitó inicialmente su bloqueo.

2.3.1.8. Proceso de Cambio de PIN [11]

Brinda la posibilidad al ciudadano de cambiar el PIN de su tarjeta de una manera segura y sin tener que recurrir al lugar donde la recibió (Oficina de Entrega), este proceso se puede realizar de varias formas: una de ellas sería proveer al ciudadano de un software que le permita cambiar su PIN, otra sería incluir en el portal web de la institución una interfaz que le permita al ciudadano realizar esta acción.

[11] Del acrónimo de Personal Identification Number, “Número de Identificación Personal”.

2.3.1.9. Proceso de Administración de Certificados Digitales

Mantiene el control sobre la creación y cambios de estado de los Certificados Digitales emitidos por la AC y su vinculación con las cédulas de identidad electrónica de los ciudadanos. ***Este proceso no es responsabilidad del Sistema Nacional de Identificación, sino de la entidad que el Gobierno defina a través de SUCERTE.***

2.3.1.10. Proceso de Administración de Servicios y Aplicaciones

Posibilita mantener el control de los servicios que puede brindar una institución determinada como pueden ser: el monedero electrónico, seguro médico, ficha estudiantil, y el acceso de ésta a secciones de almacenamiento de información dentro de la tarjeta, además de permitir adicionar o eliminar aplicaciones a la misma. Para este proceso se deben definir las interfaces necesarias para que en el futuro el usuario pueda actualizar las aplicaciones sin la intervención del ente emisor.

2.3.2. Descripción de los sistemas automatizados que existen en la empresa

En la República Bolivariana de Venezuela existe actualmente el Sistema SAIME el cual está compuesto por módulos y subsistemas y se encarga de la identificación ciudadana, migración y control de extranjeros de acuerdo a las legislaciones nacionales y extranjeras vigentes.

Este sistema ha transitado por diferentes etapas de desarrollo:

- La primera etapa fue la creación de la infraestructura necesaria que permitiera suplir las necesidades de identificación, y agilizar los trámites migratorios que podían realizar los ciudadanos venezolanos y extranjeros.
- La segunda etapa plantea la incorporación de un nuevo documento de identificación electrónica; con este documento surge la necesidad de ampliar algunos de los subsistemas desarrollados con anterioridad, e incluir otros que garanticen el control de la nueva cédula.
- En una tercera etapa se debe poder controlar las operaciones que el ciudadano pueda realizar portando la misma.

En la siguiente tabla se relacionan los subsistemas que forman parte del Sistema SAIME:

Sistema de Oficinas Regionales e Integrales.	
Subsistema de Oficina Regional.	**Subsistema de Oficina de Migración.**
Módulo de Pasaporte Andino.	Módulo de Aeropuerto.
Módulo de Pasaporte electrónico.	Módulo de Puerto.
Módulo de Cedulación.	Módulo de Punto Fronterizo.
Módulo de Extranjería.	Módulo de administración.
Módulo de Datos Filiatorios.	
Módulo de Irregularidades AFIS.	
Módulo de Administración.	

Sistema de Cede Central		
Subsistema de Sede Central	**Subsistema de Administración**	**Subsistema de Personalización**
Módulo de Identificación.	Modulo de Administración Global.	Módulo de Impresión y Control de Pasaporte.
Módulo de Prontuario.	Módulo de Administración Sede. Central.	Módulo Inventario Documentos.
Módulo de Migración.	Módulo de Administración Oficinas.	Módulo de CPID.
Módulo de Irregularidades.		
Módulo Extranjería		

Portales			
Portal Web SAIME.	Portal Web ONIDEX.	Portal Web Centro de llamadas.	Aplicación de Atención Ciudadana para Oficina.

Tabla 2.2 Subsistemas que forman parte del Sistema SAIME.

2.3.3. Descripción de los módulos que se relacionan directamente con el campo de acción del presente trabajo

2.3.3.1. Módulo de Cedulación

Módulo encargado del proceso de cedulación de los ciudadanos. Brindó la posibilidad a cientos de venezolanos y venezolanas de contar con un primer documento que los identificara ante la sociedad y el estado. Se implementó bajo el basamento de las leyes de identificación nacional y logró automatizar todos los procesos que se realizaban por el estado en este sentido de forma manual. Es el encargado de mantener el control sobre los datos personales de cada ciudadano, así como de los trámites asociados al proceso central.

2.3.3.2. Módulo de Irregularidades AFIS

Se encarga de la verificación de la identidad de las personas a través de la comparación biométrica que puede ser de uno a uno para validar identidad de una persona, o de uno a muchos en caso de que se utilice para la identificación de una persona entre muchas.

2.3.3.3. Módulo de Administración Global

Permite al administrador del sistema crear y eliminar usuarios, asignar o eliminar roles y permisos a éstos. Además, permite controlar todas las operaciones realizadas por cada usuario durante su interacción con el sistema.

2.3.3.4. Módulo Inventario de Documentos

Es de suma importancia. En un inicio se creó con el fin de almacenar solamente los pasaportes que se personalizan en el CPID (Centro de Personalización e Impresión de Documentos). En la actualidad se registran en él todos los documentos con los cuales se realizan operaciones dentro del Sistema SAIME. Es el que permite mantener el control de cada documento, así como las operaciones realizadas sobre éstos; a través de un mecanismo de estados por los cuales atraviesan durante su ciclo de vida.

2.3.3.5. Módulo de CPID

Es el encargado de la personalización e impresión de los documentos utilizados por el Sistema SAIME para cada uno de los trámites que así lo requiera.

2.4. Propuesta de Sistema

El CAMS es la relación de varios subsistemas que permiten controlar el ciclo de vida de las tarjetas, administrar los servicios y aplicaciones de las mismas, además debe ser el encargado de mantener actualizados los datos de los subsistemas relacionados con el proceso de cedulación electrónica. Nuestra propuesta se basa en cómo integrar, acoplar y mantener actualizados todos estos subsistemas.

El sistema debe contar con un componente que permita acoplar de forma segura todos los subsistemas que están relacionados con el ciclo de vida de las tarjetas, de esta forma surge la idea de crear un *Entity Management System* (*EMS*) cuya función fundamental sería la gestión de las entidades y subsistemas que se controlan dentro del Sistema SAIME. Las tarjetas inteligentes que se utilizarán como cédula de identificación son un ejemplo de estas entidades.

El CAMS tendría como su componente central al EMS; pero este específicamente basaría su campo de acción en las tarjetas inteligentes que se utilizarán como cédula y en los subsistemas que intervienen en el control del ciclo de vida de las mismas.

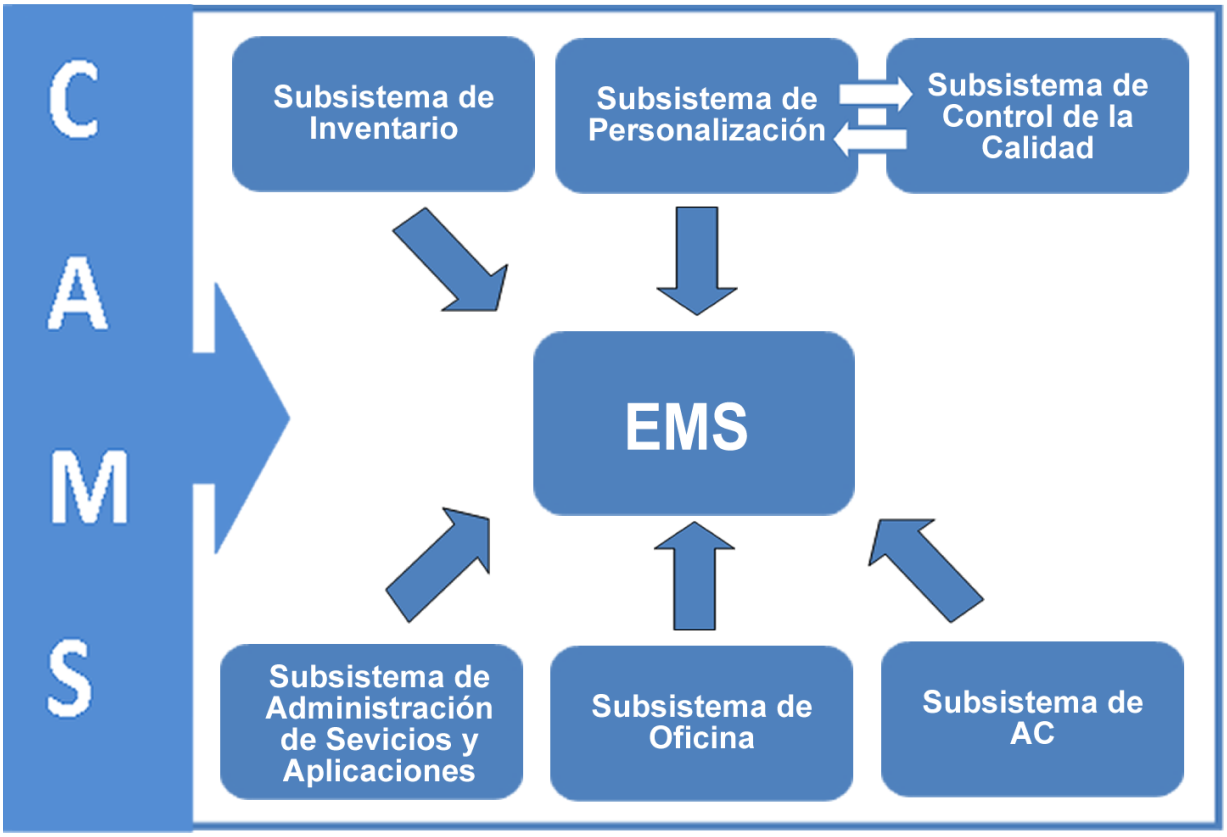

Figura 2.1 Estructura del CAMS.

En el EMS se crean los subsistemas que formarán el CAMS, se definen los estados por los que transita una tarjeta en cada uno de estos, las reglas que provocan cambios de estado y los eventos a los cuales podrán estar subscritos dichos subsistemas. Además, quedan registradas las operaciones que se realizan sobre las tarjetas en cada uno de los subsistemas y se establecen los mecanismos que permiten mantener actualizados a todos los componentes del CAMS.

Las tarjetas durante su ciclo de vida, transitan por diferentes estados. Los cambios de estos estados son registrados en el EMS el cual mantiene actualizados a los componentes restantes. Los estados que proponemos en cada subsistema para las tarjetas están representados en la siguiente figura:

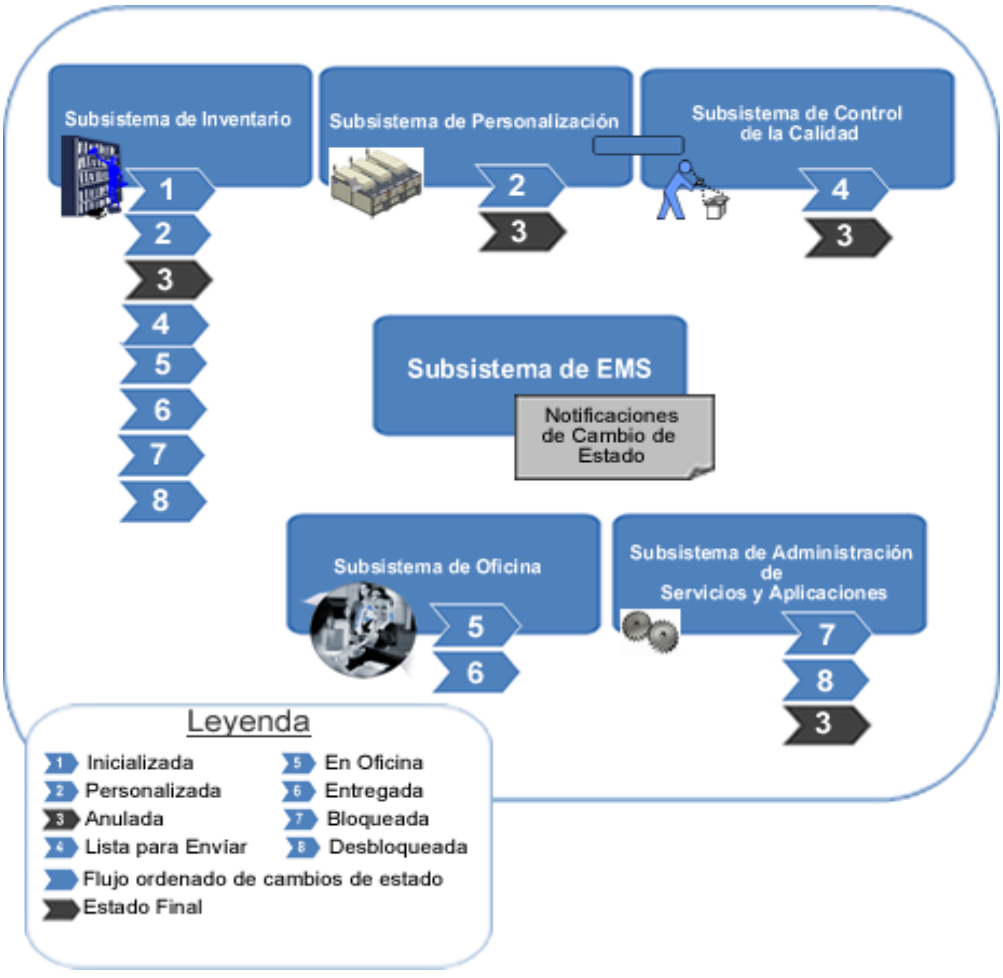

Figura 2.2 Estados de las Tarjetas en los diferentes subsistemas.

Además, el CAMS debe contar con un subsistema que permita administrar las aplicaciones y los servicios que se brindarán mediante la utilización de las tarjetas como medio de identificación, al cual llamaremos Subsistema de Administración de Servicios y Aplicaciones.

A continuación se especifican los subsistemas que forman parte del CAMS y algunas de las operaciones que se realizan en ellos que proponemos se registren para garantizar el control de todo el ciclo de vida de las tarjetas inteligentes.[12]

Subsistemas	Operaciones
Subsistema de Inventario.	▪ Inserción en el sistema de lotes de tarjetas inicializadas. ▪ Reclamación de lotes de tarjetas inicializadas al proveedor. ▪ Anulación de lotes de tarjetas o tarjetas individuales. ▪ Envío a Oficina de lotes de tarjetas. ▪ Baja de lotes de tarjetas anuladas
En este subsistema luego de cada operación debe quedar registrado el estado actual de cada tarjeta.	
Subsistema de Personalización.	▪ Personalización de lotes de tarjetas. ▪ Anulación de lotes de tarjetas o tarjetas individuales.
Subsistema de Control de la Calidad.	▪ Aprobación de Personalización de lotes de tarjetas o tarjetas individuales. ▪ Anulación de lotes de tarjetas o tarjetas individuales.
Subsistema de Oficinas.	▪ Recepción de lotes de tarjetas o tarjetas individuales. ▪ Anulación de lotes de tarjetas o tarjetas individuales. ▪ Entrega de cédula de identidad al ciudadano.
Subsistema de Administración de Servicios y Aplicaciones.	▪ Anulación de Tarjetas. ▪ Bloqueo de Tarjetas ▪ Desbloqueo de Tarjetas.
En este subsistema además se registran otras operaciones que no inciden directamente en el estado de las tarjetas pero si son necesarias para que estas puedan brindar todas las utilidades que permite el hecho de contar con un chip como: ▪ Anulación de Certificados. ▪ Administración de Aplicaciones y Servicios. ▪ Expedición de Reportes a entidades que prestan servicios a través de las tarjetas y a funcionarios que los requieran. ▪ Servicios de Cambio de Datos como el PIN a los usuarios.	
El CAMS además está compuesto por los siguientes subsistemas pero estos no inciden directamente en los estados de la tarjeta por lo que solamente los describimos brevemente:	

[12] Ver Anexo 8: Descripción de las operaciones que deben registrar los subsistemas que forman el CAMS

Subsistemas	Descripción
Subsistema de EMS.	Este subsistema es el encargado de registrar las operaciones realizadas sobre las entidades que se controlan dentro del Sistema SAIME independientemente de donde hayan ocurrido las mismas, cuenta para ello con una interfaz de comunicación que permite el acople de diferentes subsistemas que necesitan estar relacionados para garantizar el monitoreo del ciclo de vida de dichas entidades. Las características antes mencionadas hacen de este subsistema una pieza fundamental del Sistema de Administración de Tarjetas y Aplicaciones que proponemos ya que nos brinda la posibilidad de mantener retroalimentados a todos los subsistemas que lo forman y nos permitiría mantener las trazas de todo el ciclo de vida de las tarjetas inteligentes utilizadas como cédula.
Subsistema de la Autoridad Certificadora.	Este es un subsistema provisto por la Autoridad Certificadora creada por el Estado Venezolano, el cual es el encargado de la gestión de los Certificados Digitales y las Llaves que van incorporados en las cédulas de identidad electrónica

Tabla 2.3 Subsistemas que forman parte del CAMS.

2.5. Modelo de Dominio

Después de haber hecho un análisis profundo, se llegó a la conclusión de que no se modelaría el negocio a través del diagrama de casos de uso del negocio, esto es provocado por la baja estructuración de los procesos de negocios; por tal motivo es que basaremos este trabajo en un modelo de dominio; el cual es la representación de conceptos y sus relaciones. Un modelo de dominio es utilizado para esclarecer cómo funciona el entorno en el cual está enmarcado el problema.

En nuestro modelo se muestran las relaciones que existen entre los principales conceptos que componen el CAMS, destacándose los distintos subsistemas y los aspectos organizativos que deben ser tomados en cuenta para su desarrollo.

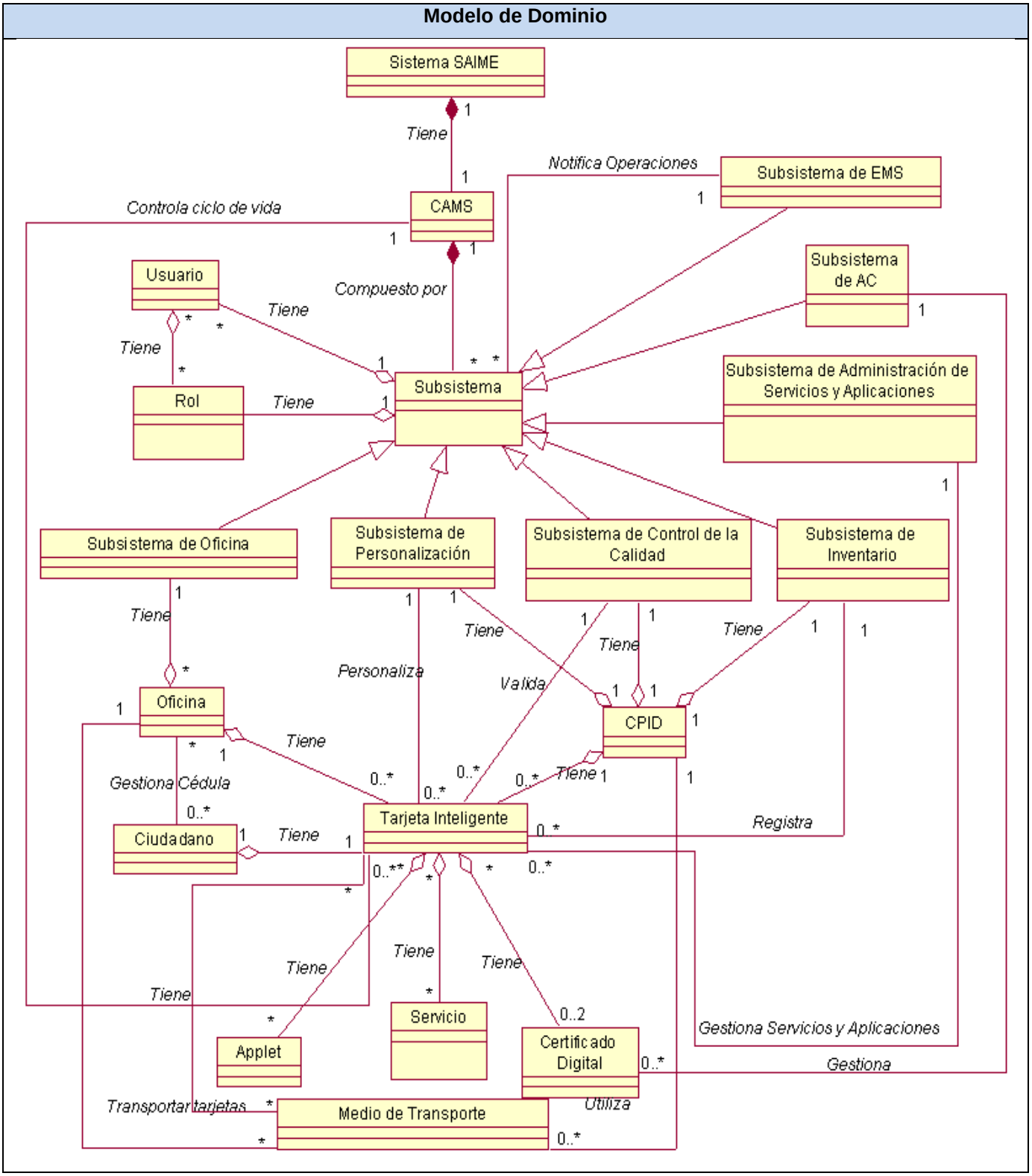

Figura 2.3 Diagrama del Modelo de Dominio.

A continuación se identificarán los conceptos más significativos que se utilizarán en el modelo de dominio:

Término	Concepto
Sistema SAIME	Es el Sistema del Servicio Autónomo de Identificación, Migración y Extranjería, en él se desarrollan un conjunto de procesos, que se implementan a través de varias aplicaciones.
CAMS	Sistema de Administración de Tarjetas Inteligentes y Aplicaciones, es una composición de subsistemas que funcionan de forma coordinada para garantizar el control del ciclo de vida de las tarjetas así como de las aplicaciones y servicios de las mismas.
Usuario	Define un concepto administrativo para garantizar seguridad en el sistema, a través de él, un funcionario del sistema podrá autenticarse y dependiendo de sus roles, podrá efectuar determinadas operaciones. Posee un estado de activación y datos de la persona vinculada.
Rol	Es un concepto relacionado con la autorización de los usuarios u operadores del sistema para utilizar las funcionalidades que este brinda, y a través de la administración se le asigna o revoca en dependencia de lo que deba hacer en ese momento.
Subsistema	Concepto utilizado para generalizar todos aquellos subsistemas que componen al CAMS.
Subsistema de Oficina	Es el subsistema encargado de registrar las operaciones que se realizan sobre la cédula en la Oficina.
Subsistema de Personalización	Es el subsistema encargado de registrar las operaciones de personalización que se realizan sobre la Tarjeta Inteligente.
Subsistema de Control de la calidad	Es el subsistema encargado de registrar las operaciones de Aprobación de Personalización de lotes de tarjetas o tarjetas individuales y Anulación de lotes de tarjetas o tarjetas individuales.
Subsistema de Inventario	Es el subsistema encargado de registrar las operaciones de inserción, reclamación, anulación, envío a oficinas y dar de baja a lotes de tarjetas en el inventario.
Subsistema de Administración de Servicios y Aplicaciones	Es el subsistema encargado de registrar las operaciones de bloqueo desbloqueo de tarjetas así como de sus aplicaciones y servicios.

Término	Concepto
Subsistema de AC	Este es el subsistema encargado de la gestión de los Certificados Digitales y las Llaves que van incorporados en las cédulas de identidad electrónica.
Subsistema de EMS	Este es el subsistema encargado de registrar las operaciones realizadas sobre las entidades que se controlan dentro del Sistema SAIME.
Oficina	Es la estación de trabajo donde se registra que existe una o varias aplicaciones y donde los usuarios pueden trabajar con el sistema y brindar un conjunto de servicios.
Ciudadano	Persona que requiere uno o varios de los servicios que ofrece la institución. Ej. Una cédula.
Tarjeta Inteligente	Es cualquier tarjeta de pequeño tamaño con circuitos integrados incluidos que permitan la ejecución de cierta lógica programada.
CPID	Centro de Personalización e Impresión de Documentos, es aquí donde se confecciona la nueva cédula, la cual es llevada a las oficinas por lotes en los medios de transporte.
Medio de transporte	Incluye todos los medios e infraestructuras implicados en el movimiento de las cédulas.
Certificado Digital	Es un documento digital mediante el cual un tercero confiable (una autoridad de certificación) garantiza la vinculación entre la identidad de un sujeto o entidad y su clave pública.
Servicio	Servicios que brindan las empresas a través de las tarjetas ejemplo Monedero Electrónico, tarjeta de salud.
Applet	Es una máquina de estados que sólo procesa los comandos recibidos a través del dispositivo lector enviando y respondiendo con códigos de estado y datos.

Tabla 2.4 Conceptos del modelo de dominio.

2.6. Especificación de los requisitos de software

2.6.1. Definición de los requisitos funcionales

El CAMS está compuesto por varios subsistemas que interactúan entre sí permitiendo el control de todo el ciclo de vida de la tarjeta. A continuación se listan los requisitos funcionales para cada uno de los subsistemas.

2.6.1.1. Subsistema de EMS

RFEMS 1. Crear subsistemas que formarán parte del CAMS.

RFEMS 1.1. Definir los estados por los que atraviesa una entidad dentro del subsistema.

RFEMS 1.2. Definir grafo de sucesión de estados.

RFEMS 1.3. Definir reglas para la transición de estados.

RFEMS 1.4. Definir las operaciones a realizar sobre la entidad.

RFEMS 1.5. Subscribir Subsistemas a eventos ya existentes.

RFEMS 1.6. Eliminar subscripciones a eventos de un Subsistema.

RFEMS 1.7. Subscribir Subsistemas a nuevo eventos.

2.6.1.2. Subsistema de Inventario

RFINV 1. Notificar eventos de operaciones que se realicen sobre una tarjeta o grupos de éstas dentro del Inventario.

Eventos:

- Operaciones realizadas sobre lotes de tarjetas y/o tarjetas individuales que impliquen cambio de estado en éstas.
- Movimientos de lotes de tarjetas y/o tarjetas individuales hacia otros subsistemas.

2.6.1.3. Subsistema de la Autoridad Certificadora

RFAC 1. Que se le hagan solicitudes de información por parte de otros subsistemas del CAMS.

RFAC 2. Recibir notificación de eventos ocurridos que impliquen cambio de estado en los certificados.

Solicitudes de Información:

- Certificados digitales y llaves.

Eventos:

- Cambio de estado de Certificados Digitales.
  1. Asignado.
  2. En Uso.
  3. Revocado.
  4. Suspendido.

2.6.1.4. Subsistema de Personalización

RFPER 1. Gestionar órdenes de personalización.

RFPER 1.1 Crear órdenes de personalización.

RFPER 1.2 Personalizar órdenes de personalización.

RFPER 2 Solicitar información al Subsistema de la Autoridad Certificadora.

- Certificados Digitales.

RFPER 3 Notificar eventos a otros subsistemas del CAMS.

- El cambio de estado de persona(s): de Listas para Cedular a Ceduladas.
- El cambio de estado de una tarjeta o lotes de estas: de Inicializadas a Personalizadas o Anuladas.

El cambio de estado de Certificados Digitales: de Asignados a Revocados o Suspendidos.

2.6.1.5. Subsistema de Oficina

RFOF 1 Notificar ***eventos*** sobre operaciones realizadas sobre las tarjetas dentro de este subsistema que implican cambio de estado de éstas a otros subsistemas del CAMS.

Eventos:

- El cambio de estado de una tarjeta o lotes de estas. Cuando se recibe en la Oficina ocurre un cambio de Enviada a Oficina a En Oficina y una ves que se entrega al ciudadano debe ser notificado otro cambio de En Oficina a Entregada.
- El cambio de estado de Certificados Digitales.

2.6.1.6. Subsistema de Administración de Servicios y Aplicaciones

Este subsistema es el encargado de Interactuar con las diferentes instituciones que brindarán servicios a los ciudadanos a través de la tarjeta, así como será el que controlará las acciones que se podrán realizar sobre las aplicaciones de las tarjetas. Debe permitir:

RFASA 1 Bloquear / Desbloquear tarjetas y servicios.

RFASA 2 Activación / Desactivación de aplicaciones.

RFASA 3 Notificar eventos a las Instituciones que brindan servicios a través de la tarjeta.

RFASA 4 Notificar eventos a otros subsistemas.

RFASA 5 Crear una notificaciones de servicios.

RFASA 6 Eliminar notificaciones de servicios.

2.6.2. Requisitos no funcionales

El CAMS formará parte de la solución de software brindada para automatizar los procesos relacionados con la identificación de los ciudadanos en la República Bolivariana de Venezuela por tal

motivo deberá, en primer lugar, cumplir con una serie de requisitos no funcionales definidos de forma general para todas las aplicaciones que componen dicha solución**.** Además, al ser un sistema gestor de todo el ciclo de vida de las tarjetas inteligentes y al estar diseñado para brindar servicios, principalmente a otros subsistemas, necesita cumplir con requisitos propios que le permitan realizar dicha función y todas las operaciones relacionadas con la misma, los cuales se definen a continuación.

2.6.2.1. Apariencia o interfaz externa

- El sistema sólo brindará interfaces de comunicación con subsistemas externos o hardware por lo que no se tendrá en cuenta el diseño de la Apariencia o Interfaz externa.
- De ser necesaria alguna interfaz, ésta debe cumplir con lo definido en el documento “Proyecto Identidad. Especificación de Requerimientos”.

2.6.2.2. Diseño e implementación

- Las Interfaces de Comunicación con los diferentes subsistemas externos o hardware deben estar implementadas de forma tal que permitan al CAMS comunicarse con diferentes tecnologías que pueden ser: Web Services, applets, DLL u otras.

2.6.2.3. Usabilidad

- Las Interfaces que se definan para la comunicación deben contar con reglas de acoplamiento de fácil implementación y estar bien documentadas.

2.6.2.4. Rendimiento

- El tráfico de información debe considerarse durante las 24 horas del día.
- Disponibilidad para soportar hasta treinta y tres mil (33000) procesos de impresión de cédulas diarias. Este aspecto deben cumplirlo todos los subsistemas relacionados con el proceso de cedulación.

2.6.2.5. Portabilidad

- El software estará construido con código totalmente portable para implementaciones libres de la plataforma .NET; aunque la dependencia a **drivers** de dispositivos externos utilizados por la aplicación no permite la migración inmediata. Muchos de los programas asociados son software propietario.

2.6.2.6. Seguridad

- Garantizar el registro de las acciones realizadas por los usuarios sobre una tarjeta o grupo de éstas.
- Garantizar canales seguros para el intercambio de información entre los diferentes subsistemas externos y hardware que se comunicarán durante el ciclo de vida de la tarjeta.
- Brindar un mecanismo de seguridad que permita controlar el acceso a regiones de almacenamiento de información dentro de la tarjeta.
- Garantizar la validación de la identidad de los usuarios al acceder a los datos contenidos en el chip; ya sean propietarios de la tarjeta o instituciones que brindan servicios a través de ésta.

2.6.2.7. Legales

- Este software será propiedad del Ministerio del Poder Popular para las Relaciones de Interior y Justicia, entregándose hasta el nivel de código fuente del sistema.
- La solicitud de Certificados Digitales sólo se hará a las Autoridades Certificadoras avaladas por la Superintendencia de Servicios de Certificación Electrónica (SUCERTE).

2.6.2.8. Confiabilidad

- Debe recuperarse en el menor tiempo posible en caso de producirse una falla en algunos de los componentes de software o hardware de los cuales depende el funcionamiento del sistema.
- Se realizarán salvas periódicas de la información en otros dispositivos y lugares.
- Confidencialidad: La información manejada por el sistema estará protegida de acceso no autorizado y divulgación.
- Integridad: La información manejada por el sistema será objeto de cuidadosa protección contra la corrupción y estados inconsistentes, de la misma forma será considerada igual a la fuente o autoridad de los datos.

2.6.2.9. Disponibilidad

- A los usuarios autorizados se les garantizará el acceso a la información definidos para su nivel de acceso.
- Los dispositivos o mecanismos utilizados en la implementación de las políticas de la seguridad no ocultarán o retrasarán a los usuarios para obtener los datos deseados en un determinado momento.

2.6.2.10. Interfaz interna

- Proveer interfaces de comunicación con diferentes Hardware.
- Proveer interfaces de comunicación con Subsistemas externos.

2.7. Modelo del sistema

La modelación de sistemas muestra la forma en que el sistema debe funcionar y facilita la comprensión de las relaciones entre las diversas actividades y el impacto que tienen entre sí. Contextualiza los procesos como parte de un gran sistema cuyo objetivo es responder a una necesidad específica del cliente.

2.7.1. Definición de los actores del sistema

Los actores del sistema son entidades externas que de alguna forma interactúa con él.

Actores	Justificación
Administrador del CAMS	Es el encargado de toda la gestión de subsistemas: crearlos, eliminarlos, gestionar su máquina de estados y sus eventos.
Personalizador de Documentos	Es el encargado de gestionar las órdenes de personalización de las tarjetas, puede crear órdenes de personalización y personalizar las mismas.
Responsable de Inventario	Es el encargado de toda la gestión con los lotes de tarjetas en el inventario.
Subsistema de Administración de Servicios y Aplicaciones	Es el encargado de interactuar con las diferentes instituciones que brindarán servicios a los ciudadanos a través de la tarjeta, así como será el que controlará las acciones que se podrán realizar sobre las aplicaciones de las tarjetas y que hacen las notificaciones a los otros subsistemas del CAMS.
Subsistema de Oficina	En este subsistema se realiza todo el proceso de entrega al ciudadano de la tarjeta de identificación y se hacen las notificaciones a los otros subsistemas del CAMS.
Subsistema de Personalización	En él se realizan las operaciones de personalización de las tarjetas de identificación y se hacen las notificaciones a los otros subsistemas del CAMS.
Subsistema	Los subsistemas que forman parte del CAMS

Actores	Justificación
Responsable de Entrega de Cedulación	Es el encargado de la entrega de la tarjeta de identificación al ciudadano.
Administrador de Tarjetas y Servicios	Es el encargado de gestionar las tarjetas de identificación y los servicios que esta brinda.
Administrador de Aplicaciones	Es el encargado de la gestión de las aplicaciones contenidas en la tarjeta de identificación.
Notificador de Servicios	Es el encargado de notificar los servicios que puede brindar una empresa determinada.
Empresa que brinda Servicios	Empresa que puede brindar servicios que podría tener la tarjeta de identificación.

Tabla 2.5 Actores del sistema.

2.7.2. Descripción de Casos de Uso del Sistema [13]

Un caso de uso es un fragmento de funcionalidad del sistema que proporciona al usuario un resultado importante. Los casos de uso representan los requisitos funcionales, en su conjunto constituyen el modelo de casos de uso que describe la funcionalidad total del sistema.

2.7.2.1. Paquete de Administración del CAMS

Diagrama casos de uso del sistema para la Administración del CAMS.

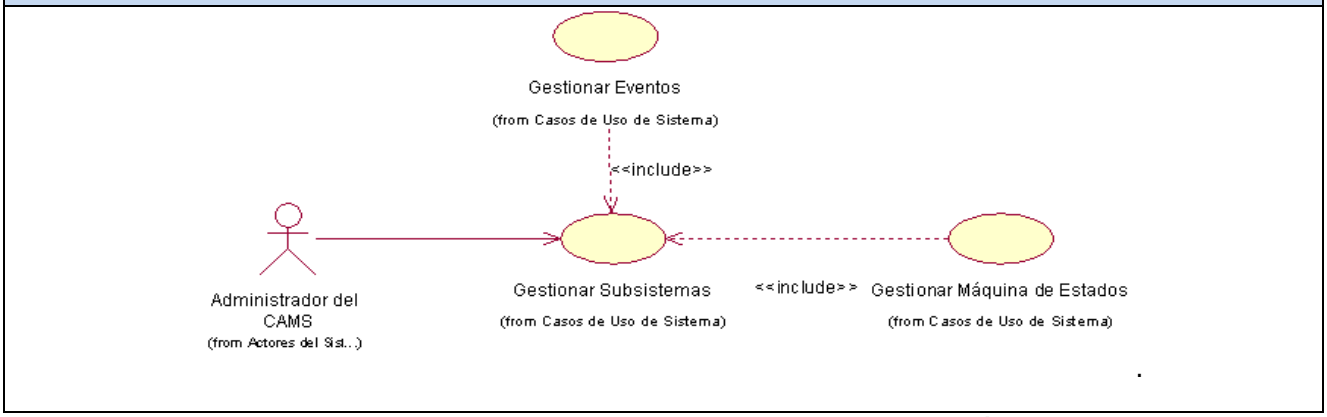

Figura 2.4 Diagrama casos de uso del sistema para la Administración del CAMS.

[13] Ver Anexo 19 Descripción de extendida de los Casos de Uso del Sistema.

2.7.2.2. Paquete de Inventario

Diagrama casos de uso del sistema para el Inventario de Tarjetas.

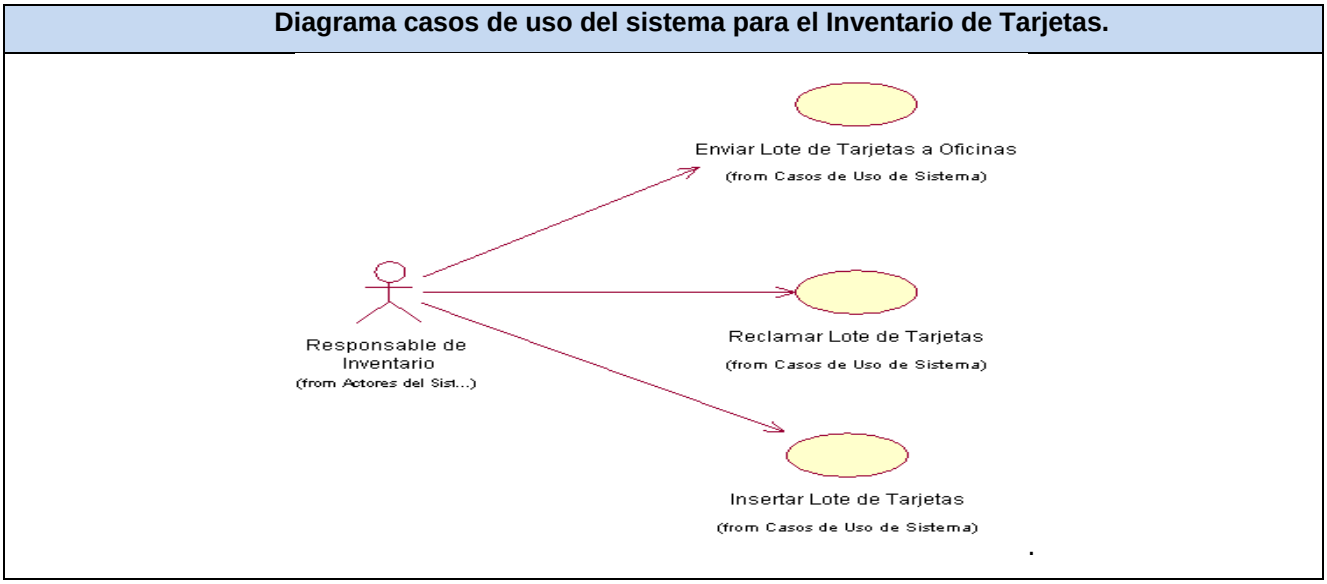

Figura 2.5 Diagrama casos de uso del sistema para el Inventario de Tarjetas.

2.7.2.3. Paquete de Autoridad Certificadora

Diagrama casos de uso del sistema para la notificación de Operaciones a la AC.

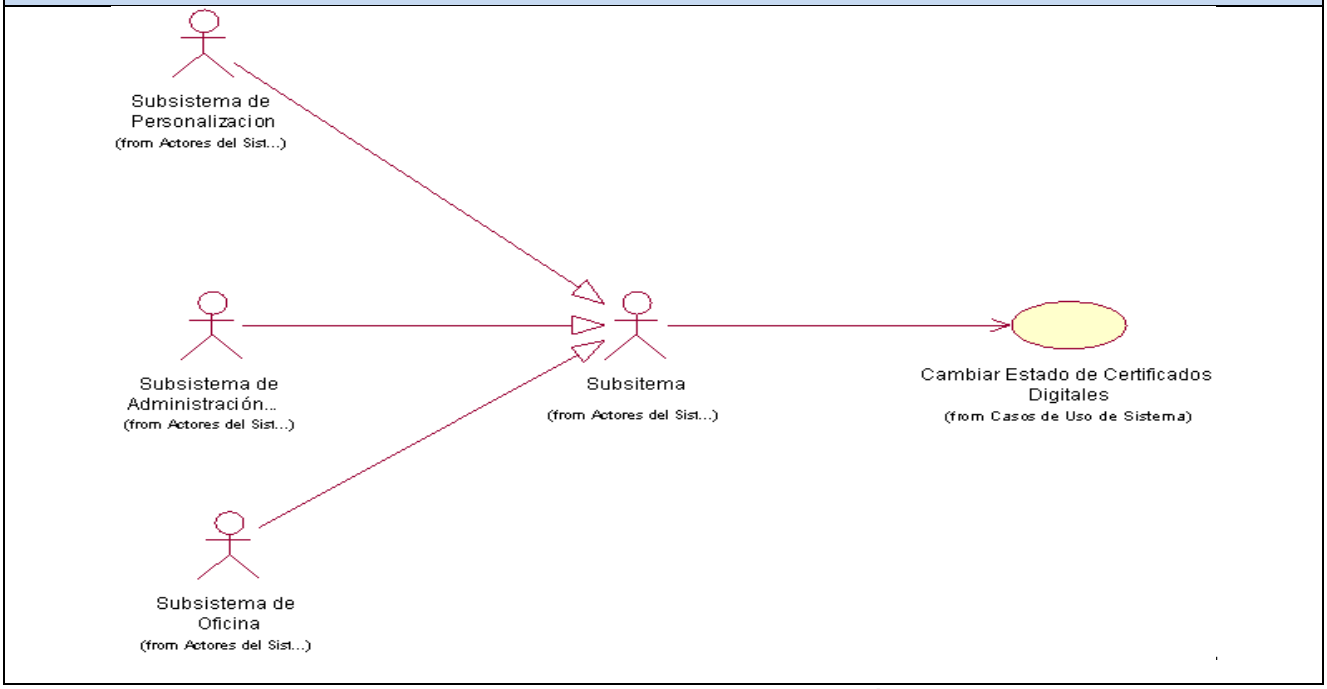

Figura 2.6 Diagrama casos de uso del sistema para la notificación de Operaciones a la AC.

2.7.2.4. Paquete de Personalización

Diagrama casos de uso del sistema para la Personalización de Tarjetas.

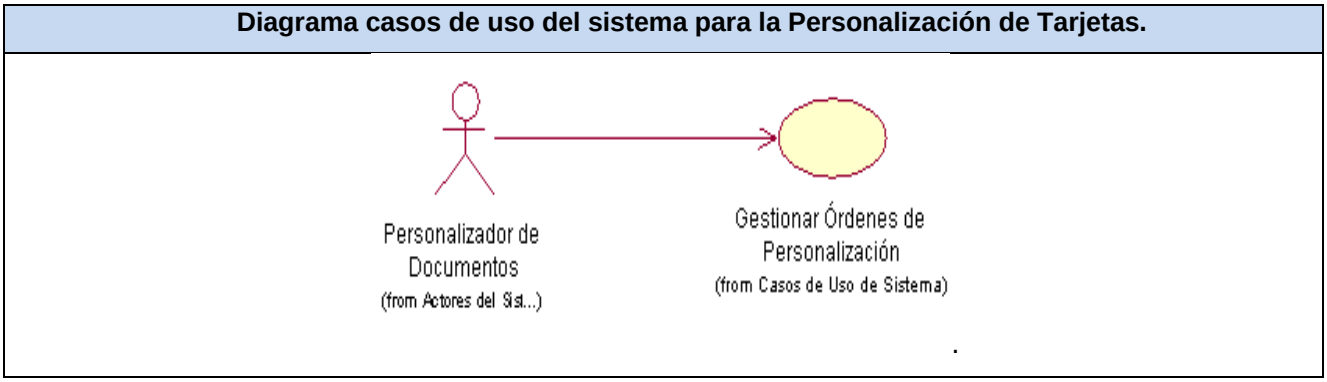

Figura 2.7 Diagrama casos de uso del sistema para la Personalización de Tarjetas.

2.7.2.5. Paquete de Oficina

Diagrama casos de uso del sistema para la Oficina.

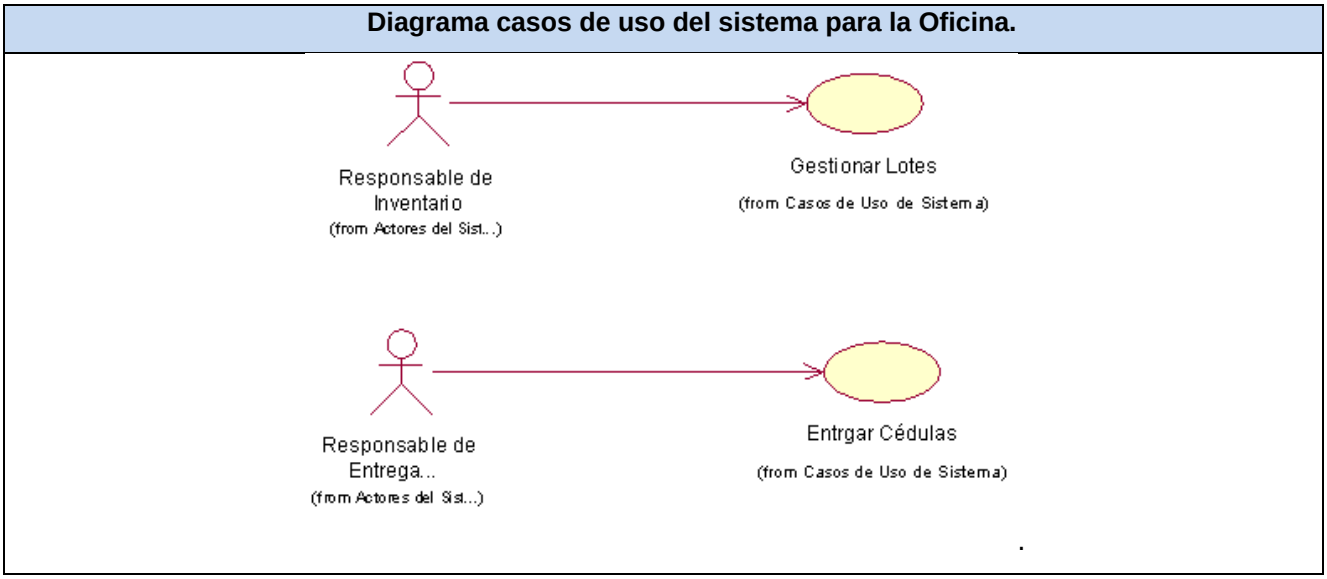

Figura 2.8 Diagrama casos de uso del sistema para la Oficina.

2.7.2.6. Paquete de Administración de Servicios y Aplicaciones

Diagrama casos de uso del sistema para la Administración de Servicios y Aplicaciones.

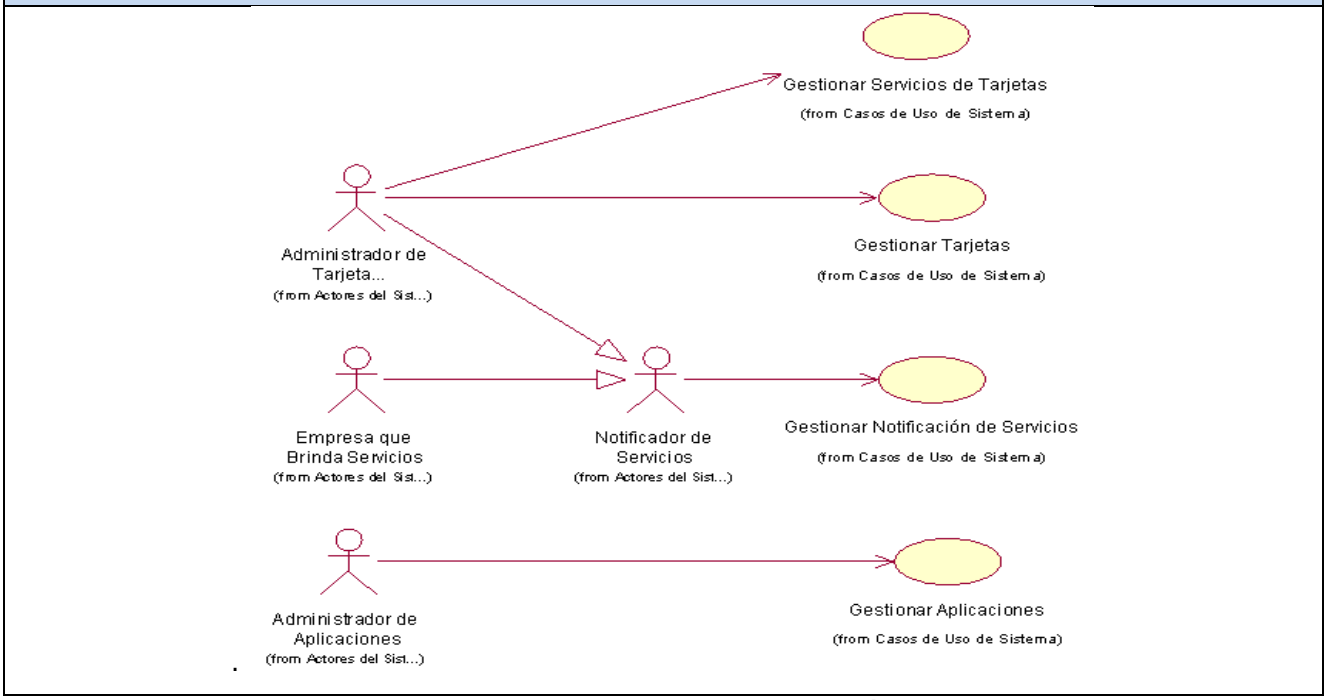

Figura 2.9 Diagrama casos de uso del sistema para la Administración de Servicios y Aplicaciones.

2.8. Estimación de esfuerzo

La estimación de esfuerzo es una de las tareas más importantes en la gestión de un proyecto de software. En la actualidad existen técnicas que permiten realizar esta labor tales como: COCOMO II y Puntos de Casos de Uso.

La estimación mediante el análisis de Puntos de Casos de Uso es un método propuesto originalmente por Gustav Karner de Objectory AB, y posteriormente refinado por muchos otros autores. Se trata de un método de estimación del tiempo de desarrollo de un proyecto mediante la asignación de "pesos" a un cierto número de factores que lo afectan, para finalmente, contabilizar el tiempo total estimado para el desarrollo de un proyecto.

2.8.1. Cálculo de esfuerzo

El primer paso para la estimación consiste en el cálculo de los Puntos de Casos de Uso sin ajustar. 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.

- **Factor de Peso de los Actores sin Ajustar (UAW)**

Este valor se calcula mediante un análisis de la cantidad de Actores presentes en el sistema y la complejidad de cada uno de ellos. La complejidad de los Actores 14se establece teniendo en cuenta en primer lugar si se trata de una persona o de otro sistema, y en segundo lugar, la forma en la que el actor interactúa con el sistema.

Actores del sistema	Tipo de actor	Factor de Peso
Administrador del CAMS.	Complejo.	3
Personalizador de Documentos.	Complejo.	3
Responsable de Inventario.	Complejo.	3
Subsistema de Administración de Servicios y Aplicaciones	Simple.	1
Subsistema de Oficina.	Simple.	1
Subsistema de Personalización.	Simple.	1
Responsable de Entrega de Cedulación.	Complejo.	3
Administrador de Tarjetas y Servicios.	Complejo.	3
Administrador de Aplicaciones.	Complejo.	3
Notificador de Servicios.	Complejo.	3

2.6 Factor de Peso de los actores del sistema.

UAW es igual a la suma de la multiplicación del tipo de actor por su factor de peso.

UAW = 7 x 3 + 3 x 1

UAW = 24.

- **Factor de Peso de los Casos de Uso sin ajustar (UUCW)**

Este valor se calcula mediante un análisis de la cantidad de Casos de Uso presentes en el sistema y la complejidad de cada uno de ellos. La complejidad de los Casos de Uso [15]se establece teniendo en cuenta la cantidad de transacciones efectuadas en el mismo, donde una transacción se entiende como

[14] Ver Anexo 9 Criterios para determinar la complejidad de los actores del sistema.

[15] Ver Anexo 10 Capítulo 2 Criterios para determinar la complejidad de los casos de uso del sistema.

una secuencia de actividades atómica, es decir, se efectúa la secuencia de actividades completa, o no se efectúa ninguna de las actividades de la secuencia.

- **Tipos de Casos de Uso y su peso**

Casos de uso	Tipo de Caso de Uso	Factor de Peso
Gestionar Subsistemas.	Complejo	15
Gestionar Máquinas de Estado.	Complejo	15
Gestionar Eventos.	Complejo	15
Insertar Lote de Tarjetas.	Simple	5
Reclamar Lote de Tarjetas.	Simple	5
Enviar Lote de Tarjetas a Oficina.	Simple	5
Cambiar Estado de Certificados Digitales.	Simple	5
Crear Certificados Digitales y Llaves	Simple	5
Gestionar Órdenes de Personalización.	Complejo	15
Gestionar Lotes.	Complejo	15
Entregar Cédulas.	Medio	10
Gestionar Servicios de tarjetas	Complejo	15
Gestionar Tarjetas	Complejo	15
Gestionar Notificación de Servicios.	Complejo	15
Gestionar Aplicaciones	Complejo	15

Tabla 2.7 Factor de Peso de los Casos de Uso del Sistema.

UUCW es igual a la suma de la multiplicación del Caso de Uso por su factor de peso.

UUCW = 9 x 15 + 1 x 10 + 5 x 5

UUCW = 135 + 35

UUCW = 170.

Al ser **UAW**=24 y **UUCW**=170 finalmente, los Puntos de Casos de Uso sin ajustar resultan:

UUCP = **UAW** + **UUCW** sustituyendo tenemos:

UUCP = 24 + 170

UUCP = 194.

- **Cálculo de Puntos de Casos de Uso ajustados.**

UCP = **UUCP** x **TCF** x **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 **TCF** es el coeficiente que se calcula mediante la cuantificación de un conjunto de factores que determinan la complejidad técnica del sistema. Cada uno de los factores se cuantifica con un valor de 0 a 5, donde 0 significa un aporte irrelevante y 5 un aporte muy importante.

Factor	Descripción	Peso	Valor asignado
T1	Sistema distribuido	2	4
T2	Objetivos de performance o tiempo de respuesta	1	1
T3	Eficiencia del usuario final	1	1
T4	Procesamiento interno complejo	1	4
T5	El código debe ser reutilizable	1	5
T6	Facilidad de instalación	0.5	4
T7	Facilidad de uso	0.5	4
T8	Portabilidad	2	2
T9	Facilidad de cambio	1	3
T10	Concurrencia	1	5
T11	Incluye objetivos especiales de seguridad	1	5
T12	Provee acceso directo a terceras partes	1	4
T13	Se requieren facilidades especiales de entrenamiento	1	3

Tabla 2.8 Factor de Complejidad Técnica.

El Factor de complejidad técnica se calcula mediante la siguiente ecuación:

TCF = 0.6 + 0.01 x Σ (Peso i x Valor asignado i)

TCF = 0.6 + 0.01 ***** 47.

TCF = 1.07.

- **Factor de ambiente (EF)**

Las habilidades y el entrenamiento del grupo involucrado en el desarrollo tienen un gran impacto en las estimaciones de tiempo. Estos factores son los que se contemplan en el cálculo del Factor de ambiente.

Factor	Descripción	Peso	Valor asignado
E1	Familiaridad con el modelo de proyecto utilizado.	1.5	2
E2	Experiencia en la aplicación.	0.5	2
E3	Experiencia en trabajo orientado a objetos.	1	4

Factor	Descripción	Peso	Valor asignado
E4	Capacidad del analista líder.	0.5	5
E5	Motivación.	1	5
E6	Estabilidad de los requerimientos.	2	3
E7	Personal a tiempo completo.	-1	4
E8	Dificultad del lenguaje de programación.	-1	2

Tabla 2.9 Factor Ambiente.

El Factor de ambiente se calcula mediante la siguiente ecuación:

EF =1.4 - 0.03 x Σ (Peso i x Valor asignado i).

EF = 1.4 – 0.03 x 14.5

EF = 0.932.

Al ser **UUCP** = 194, TCF = 1.07, EF = 0. 932 y

UCP = UUCP x TCF x EF

UCP = 194 x 1.07 x 0. 932

UCP = 193.5

El esfuerzo en horas-hombre viene dado por:

E = UCP x CF

Donde:

E: Esfuerzo

UCP: Puntos de Casos de Uso ajustados (calculado anteriormente).

CF: Factor de conversión (para este tipo de proyecto 20 horas-hombre/Punto de Casos de Uso)

Al tener **UCP** = 193.5, **CF** = 20 horas-hombre/Punto de Casos de Uso y

E = UCP x CF

E = 193.5 x 20 horas-hombre/Punto de Casos de Uso.

E = 3870 horas-hombre.

- **Esfuerzo por flujo de trabajo**

Actividad	Porcentaje	Esfuerzo
Análisis	10%	967.5
Diseño	20%	1935
Implementación	40%	3870
Prueba	15%	1451.25
Sobrecarga	15%	1451.25
Total	100%	9675

Tabla 2.10 Esfuerzo por flujo de trabajo.

El proyecto requiere de 9675 horas-hombre para su desarrollo. Trabajando 8 horas diarias en un mes de 24 días laborables se obtiene aproximadamente 192 horas mensuales, es decir, el proyecto realizándolo una persona tendría una duración de 50 meses aproximadamente.

En el caso del CAMS este tiempo sería reducido por la incorporación a su desarrollo de un equipo de trabajo de un número variable de personas.

- **Costos y beneficios**

Los beneficios económicos que se obtienen con el desarrollo del CAMS son intangibles; pues su objetivo principal es dotar al Sistema SAIME de una herramienta para la gestión de las tarjetas inteligentes y sus aplicaciones; no es un software independiente que pueda comercializarse.

Durante su desarrollo se prevé la creación de un sistema para la gestión de entidades genéricas; adaptado en este caso a las que se controlan dentro del Sistema SAIME. Dicho componente puede adaptarse a múltiples sistemas ya que su diseño genérico lo permite, por lo que sí puede ser comercializado y reportar ganancias.

El sistema no incluye costos pues no ha sido necesario realizar gastos para adquirir equipamiento técnico, ni en salarios para desarrolladores; la fuerza de trabajo con la que se cuenta consiste en estudiantes que no son remunerados y el equipamiento es proporcionado por la UCI.

2.9. Conclusiones

En este capitulo ha quedado confeccionada una propuesta de sistema para la administración de las tarjetas inteligentes y aplicaciones de la cédula de identidad electrónica de la República Bolivariana de Venezuela, para ello se analizaron los procesos actuales relacionados con el campo de acción de nuestro trabajo, puntualizando los principales conceptos y sus relaciones a través de un modelo de dominio, han quedado expuestas; mediante diagramas de casos de uso, las principales funcionalidades que deberán estar presentes en el CAMS, además se plantearon los requisitos funcionales y no funcionales que debe cumplir el sistema.

Capítulo 3 : Análisis y diseño del sistema

3.1. Introducción

El presente capítulo describe como debe ser el sistema, realizando su análisis y diseño basado en los requisitos de software detectados con anterioridad. Se muestran los diagramas de clases del análisis y las relaciones presentes entre las mismas. Del diseño se presentan los diagramas de clases y de interacción correspondientes a cada caso de uso del sistema. Además, se expone el modelo de datos propuesto y se describen sus tablas.

3.2. Análisis

El análisis consiste en obtener una visión del sistema. Responde a la pregunta: ¿Qué debe hacer el sistema? Es descrito en el lenguaje de los desarrolladores y analiza a profundidad los requisitos funcionales. Esboza de forma clara cómo llevar a cabo el desarrollo del sistema, se incluyen las funcionalidades significativas para la arquitectura. Es la primera aproximación al diseño.

3.2.1. Modelo de clases del análisis

El modelo de clases del análisis está estructurado por clases y paquetes estereotipados que proporcionan la organización de la vista interna del sistema. Es utilizado por los desarrolladores para comprender cómo debe ser diseñado e implementado el sistema.

3.2.1.1. Paquete de Administración del CAMS

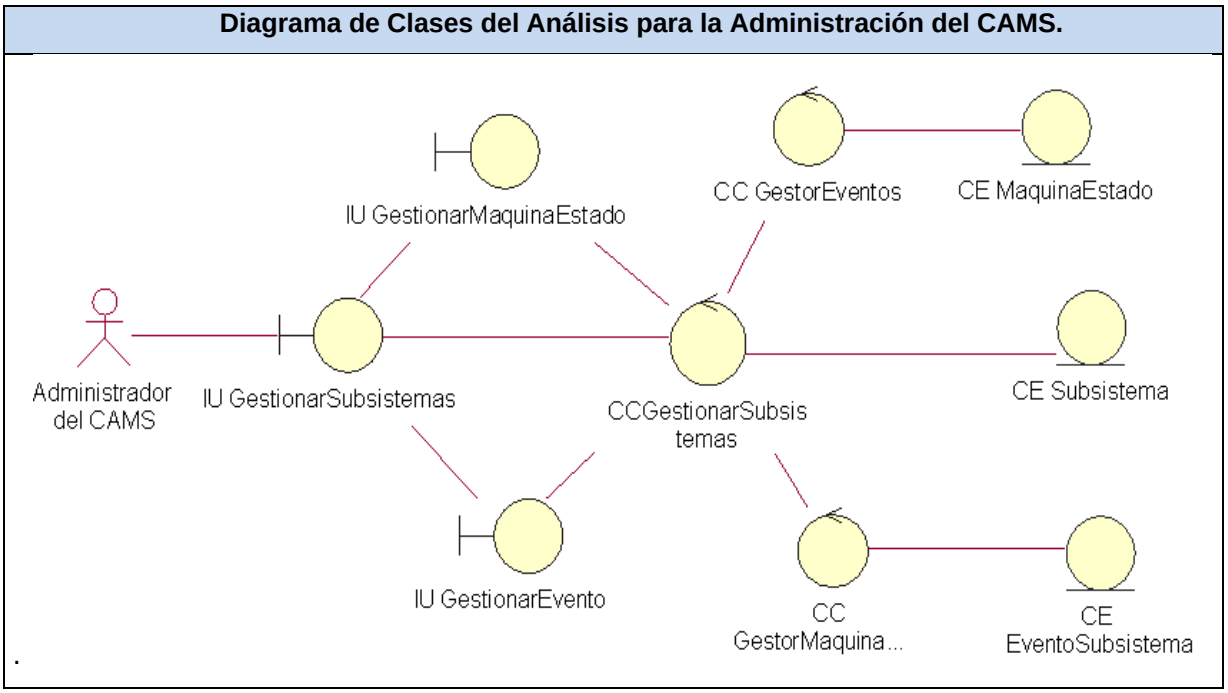

Figura 3.1 Diagrama de Clases del Análisis para la Administración del CAMS.

3.2.1.1. Paquete de Inventario

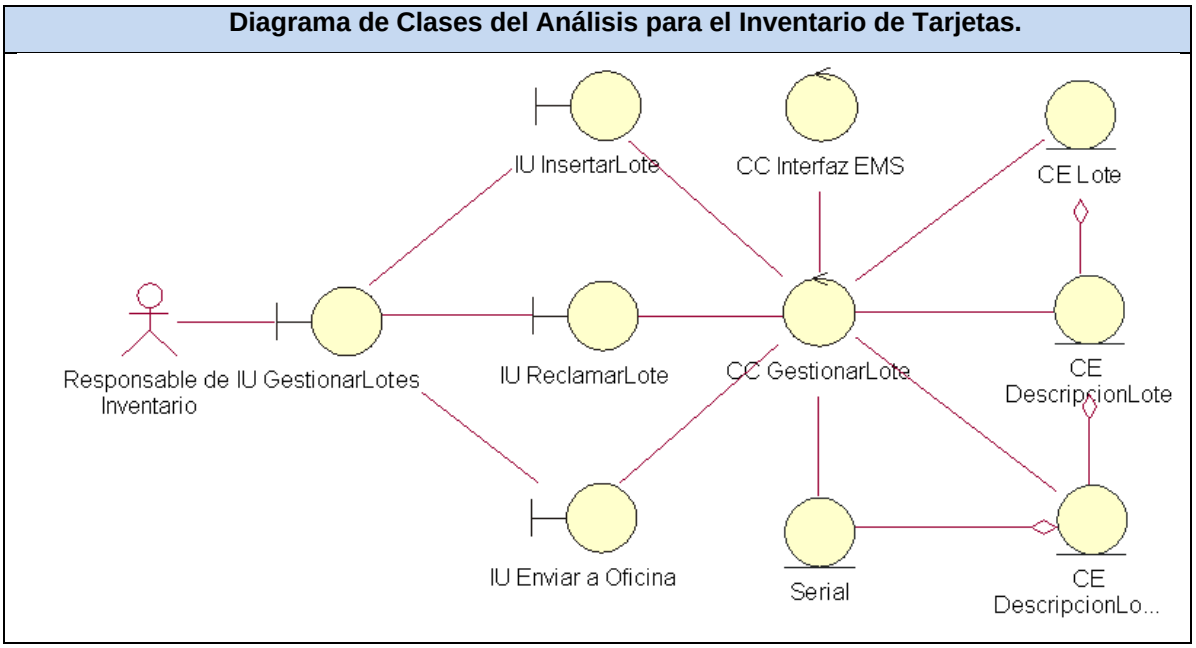

Figura 3.2 Diagrama de Clases del Análisis para el Inventario de Tarjetas.

3.2.1.2. Paquete de Autoridad Certificadora

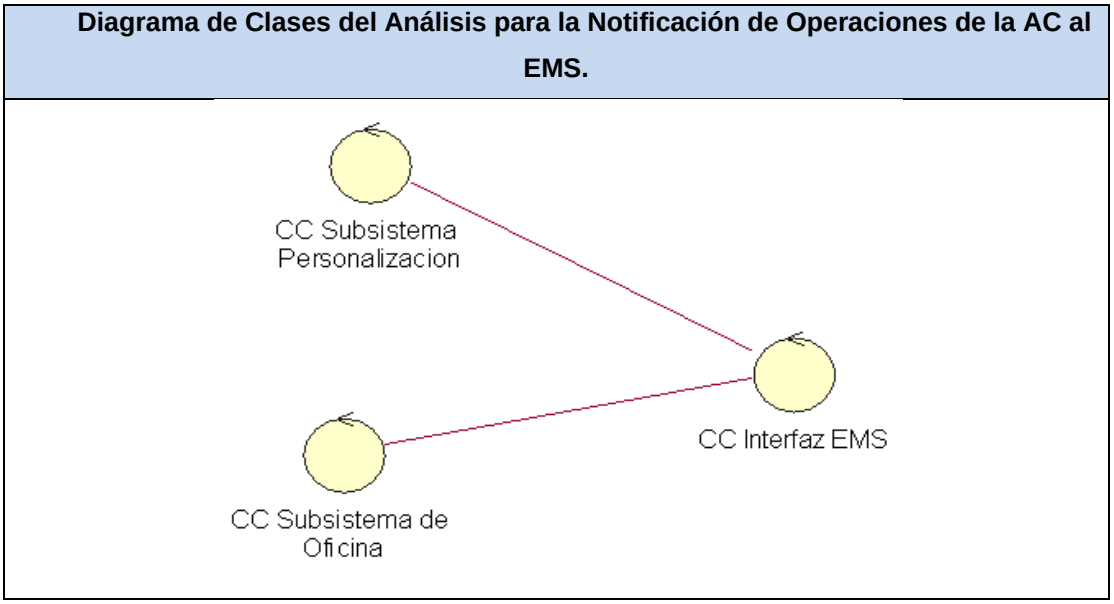

Figura 3.3 Diagrama de Clases del Análisis para la Notificación de Operaciones de la AC al EMS.

3.2.1.3. Paquete de Personalización

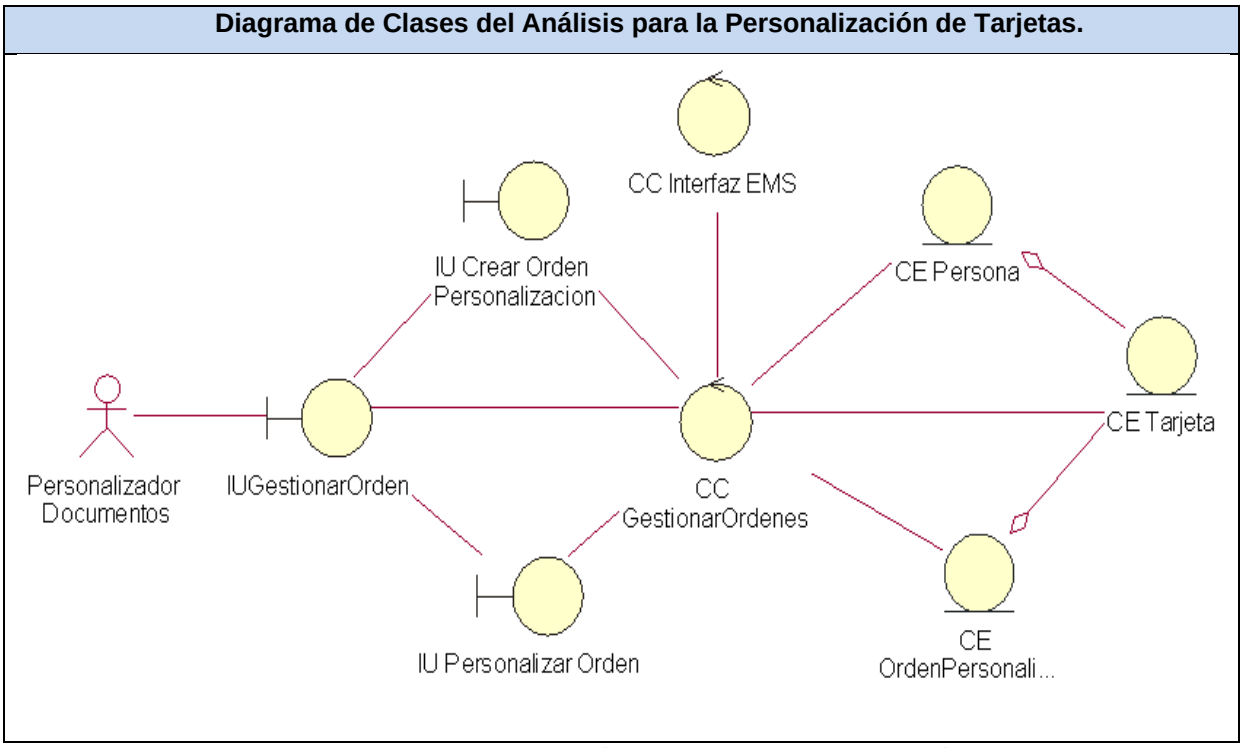

Figura 3.4 Diagrama de Clases del Análisis para la Personalización de Tarjetas.

3.2.1.4. Paquete de Oficina

Diagrama de Clases del Análisis para la Oficina de Entrega.

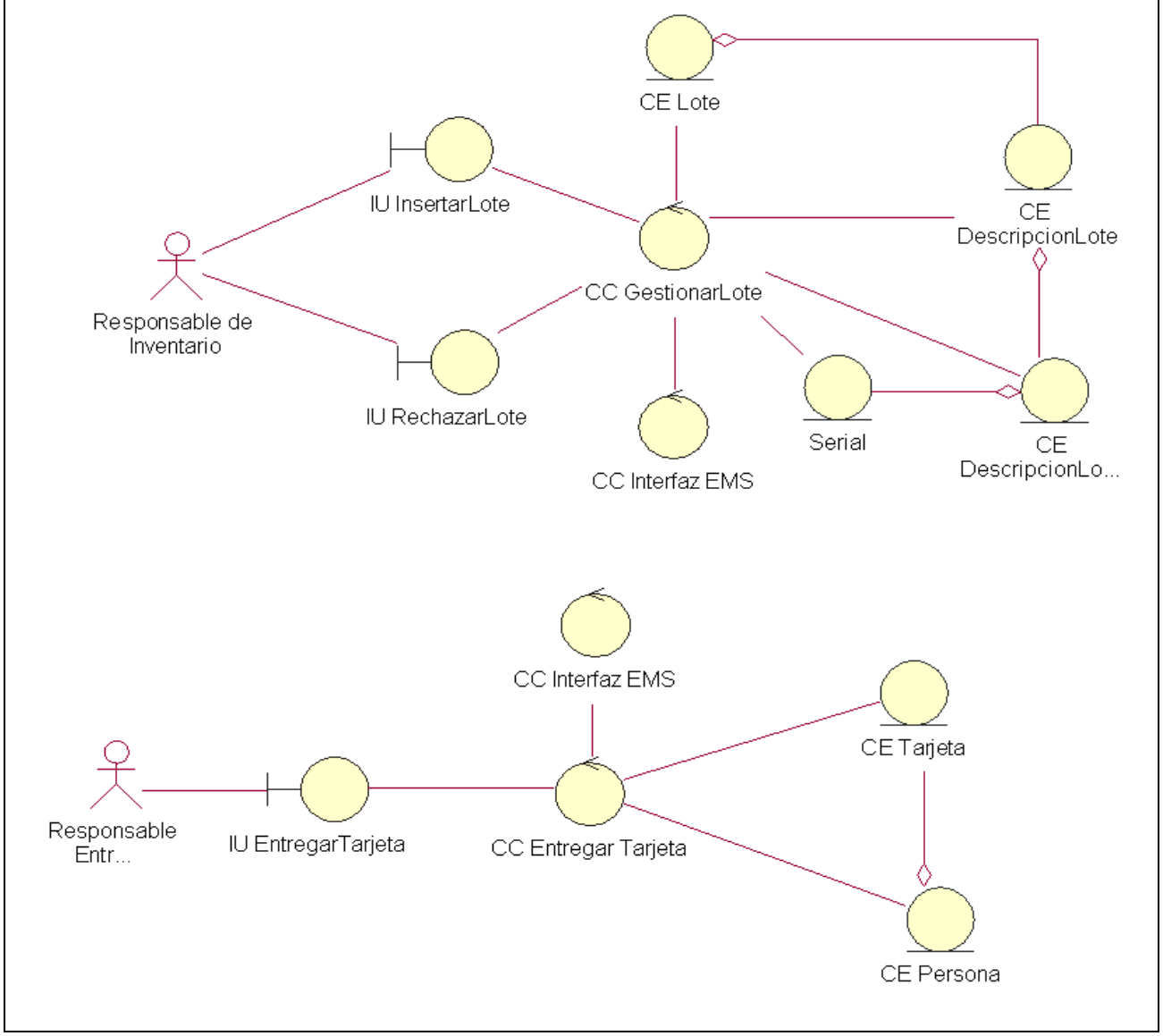

Figura 3.5 Diagrama de Clases del Análisis para la Oficina de Entrega.

3.2.1.5. Paquete de Administración de Servicios y Aplicaciones

Diagrama de Clases del Análisis para la Administración de Servicios y Aplicaciones.

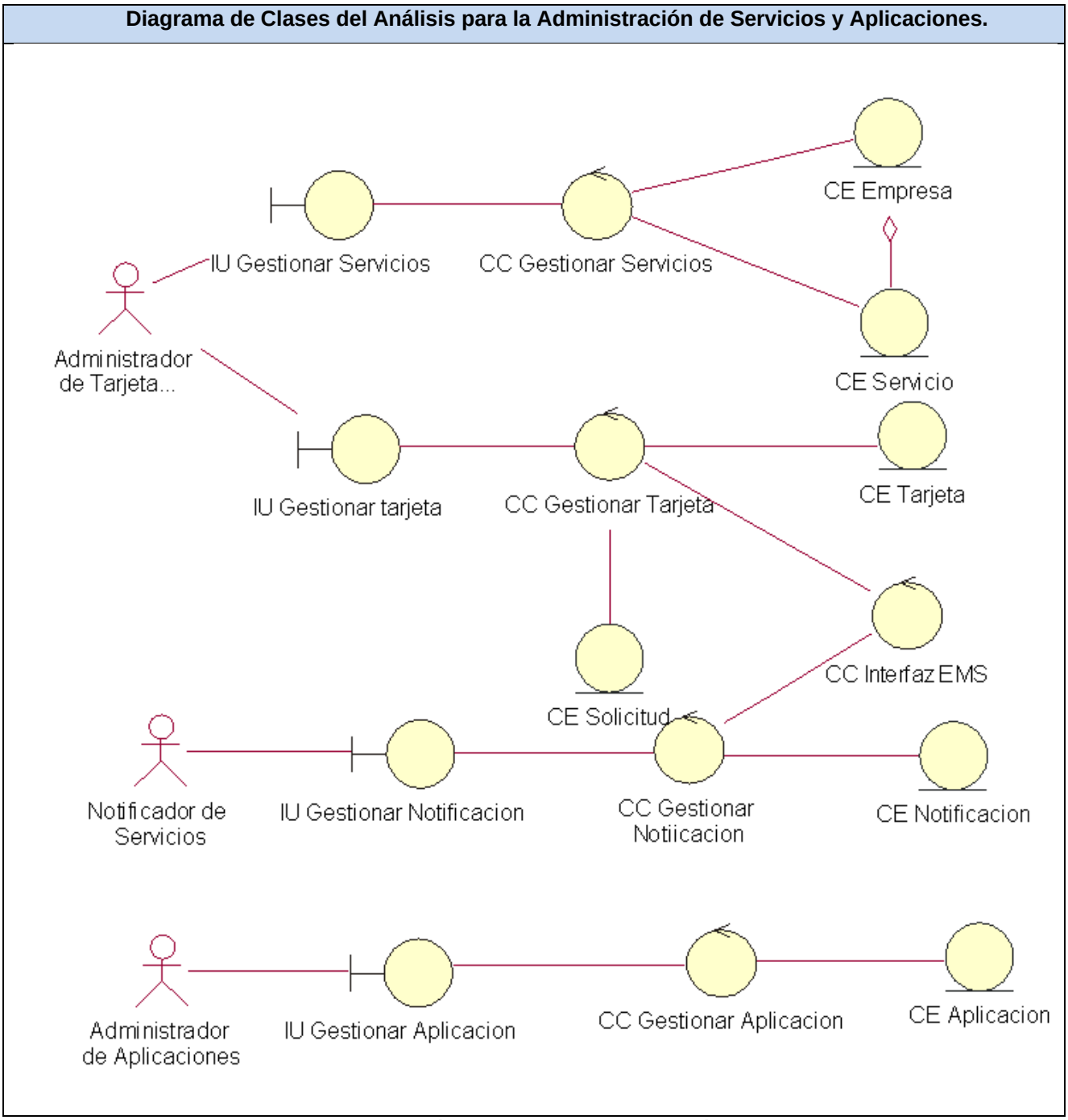

Figura 3.6 Diagrama de Clases del Análisis para la Administración de Servicios y Aplicaciones.

3.3. Diseño

El diseño es un refinamiento del análisis que tiene en cuenta los requisitos no funcionales, expone CÓMO el sistema cumple sus objetivos. Debe ser suficiente para que el sistema pueda ser implementado sin ambigüedades. Se modela la arquitectura para que soporte los requisitos no funcionales y las restricciones que se le suponen.

3.3.1. Descripción de la arquitectura

La propuesta de arquitectura de los componentes del CAMS está basada en la implementación del patrón Modelo Vista Controlador (MVC). Este patrón permite un desacople entre las capas de la aplicación; marcando una separación entre la vista o interfaz de usuario, el modelo de diseño; que permite dar solución a los problemas que debe resolver el sistema así como la gestión de los datos de la aplicación u otras funcionalidades y la capa que realiza el control y acople entre las dos anteriores.

Entre sus principales ventajas se encuentra la posibilidad de diseñar interfaces totalmente indiferentes a cómo se van a gestionar sus eventos, esto permite que se divida el trabajo entre diferentes especialistas del equipo de desarrollo.

En un nivel inferior se encuentra la Capa de Acceso a Datos (CAD), la cual es generada por una herramienta comercial adquirida por el proyecto llamada TierDeveloper v4.0, esta capa es usada por el controlador para la interacción con la base de datos; abstrayendo a ésta de la forma que se realizan dichas operaciones.

Para la implementación de la solución proponemos el uso del Framework Común versión 2.0. Este paquete fue construido en las primeras fases de desarrollo del Sistema SAIME, y su objetivo es brindar un conjunto de clases bases, y funcionalidades comunes para los distintos módulos de dicho sistema.

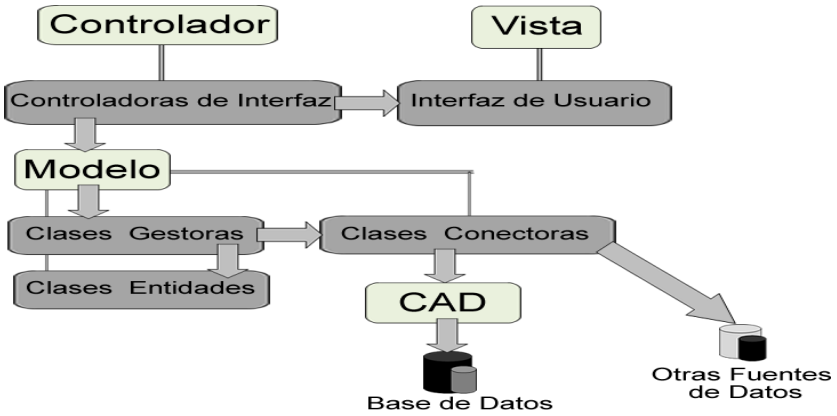

Figura 3.7 Propuesta de Arquitectura.

3.3.2. Diagramas de Clases de Diseño

Los diagramas de clases del diseño muestran las definiciones de los objetos que serán implementados, las interacciones entre ellos y las responsabilidades asignadas a cada uno.

Diagrama de Clases Simplificado: Clases Interfaz del Subsistema EMS[16].

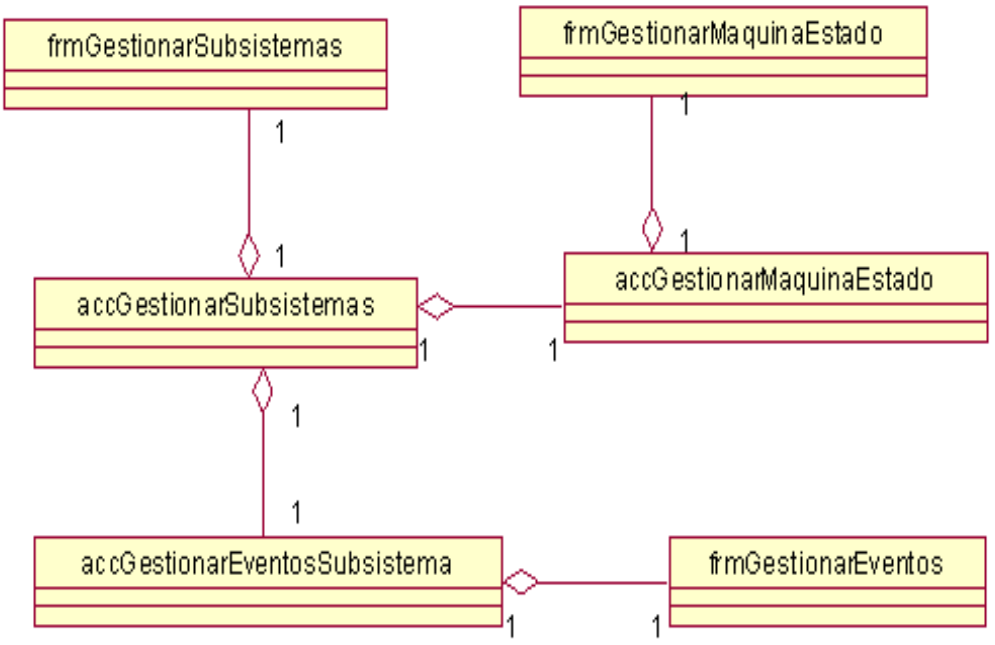

Figura 3.8 Diagrama de Clases Interfaz del Subsistema EMS.

[16] Ver Anexo 11 Descripción de las clases para la Interfaz de Usuario del Subsistema EMS.

Diagrama de Clases Simplificado: Clases Controladoras del Subsistema EMS.[17]

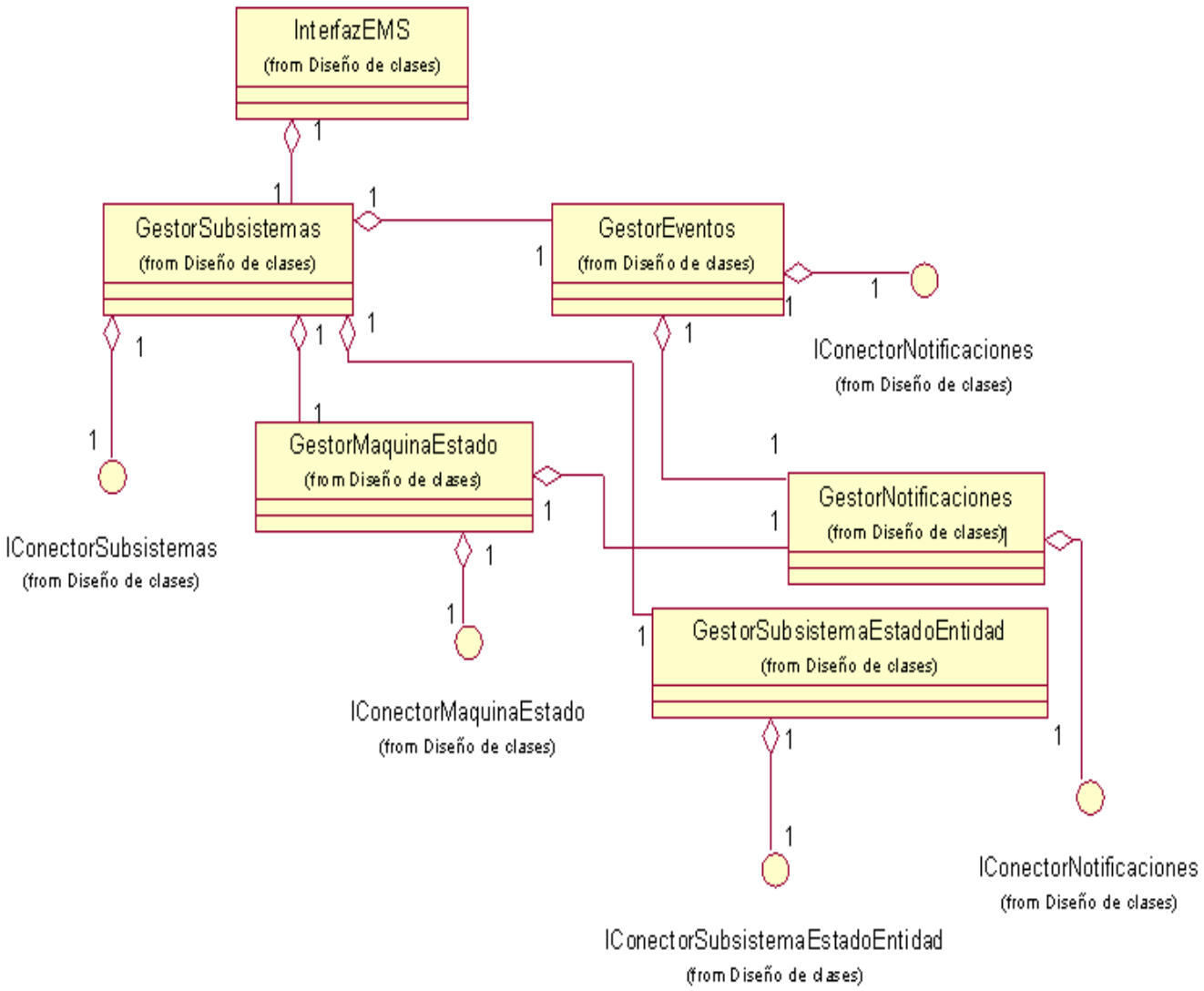

Figura 3.9 Diagrama de Clases Controladoras del Subsistema EMS.

[17] Ver Anexo 12 Descripción de las clases controladoras del Subsistema EMS.

Diagrama de Clases Simplificado: Clases Entidades del Subsistema EMS (Parte 1).[18]

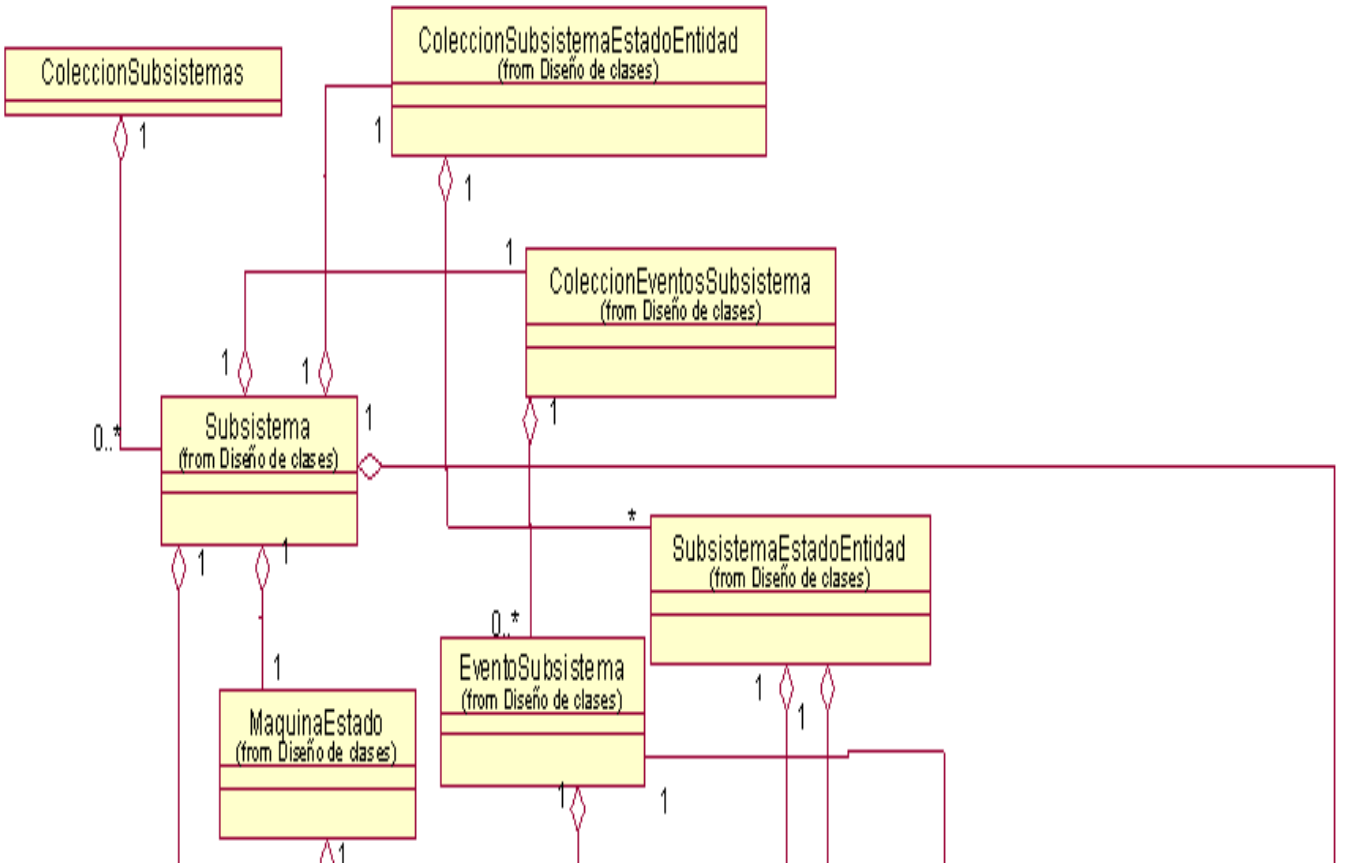

Figura 3.10 Diagrama de Clases Entidades del Subsistema EMS (Parte 1).

[18] Ver Anexo 13 Descripción de las clases entidades del Subsistema EMS.

Diagrama de Clases Simplificado: Clases Entidades del Subsistema EMS (Parte 2).

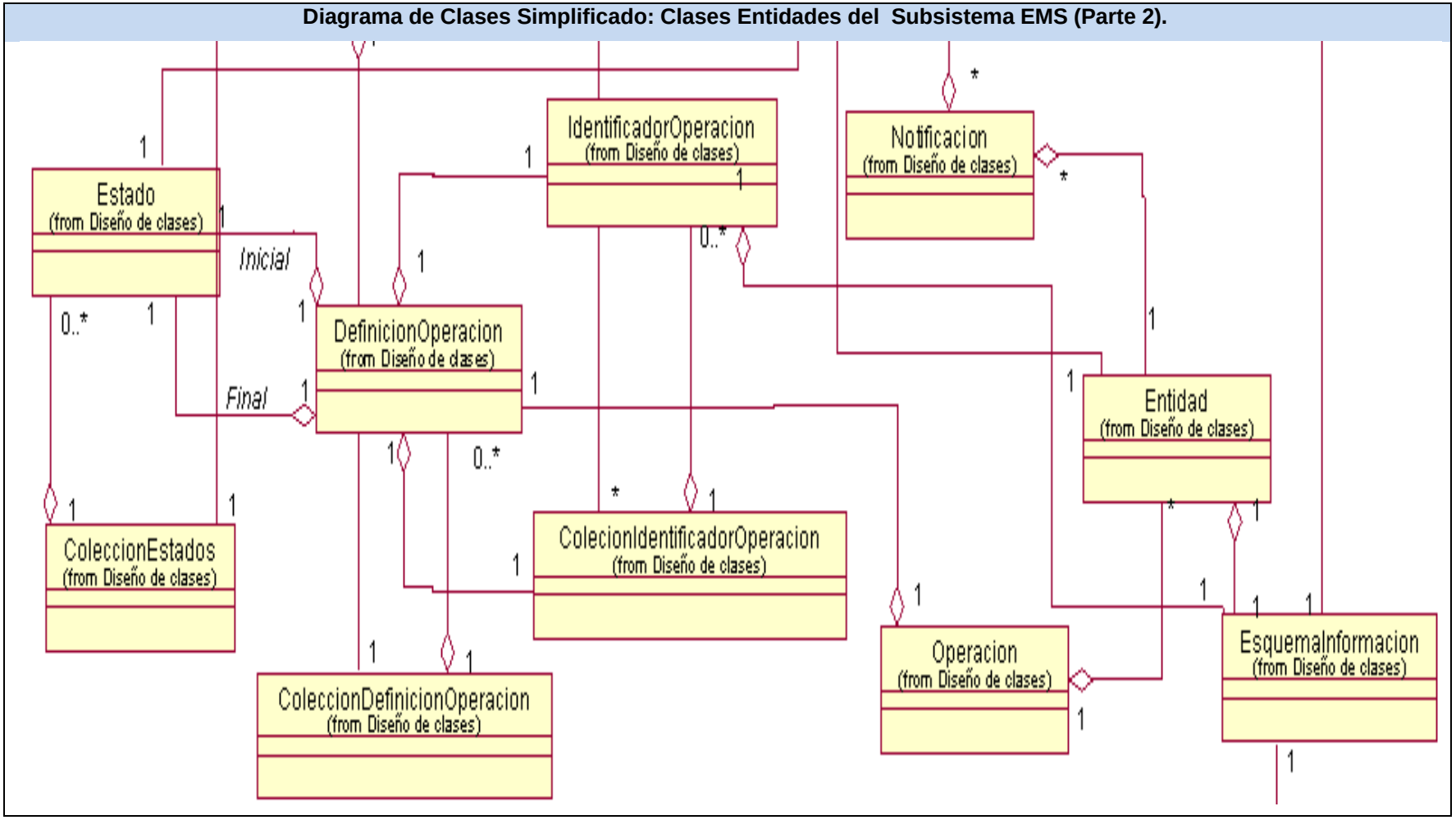

Figura 3.11 Diagrama de Clases Entidades del Subsistema EMS (Parte 2).

Diagrama de Clases Simplificado: Clases Entidades del Subsistema EMS (Parte 3).

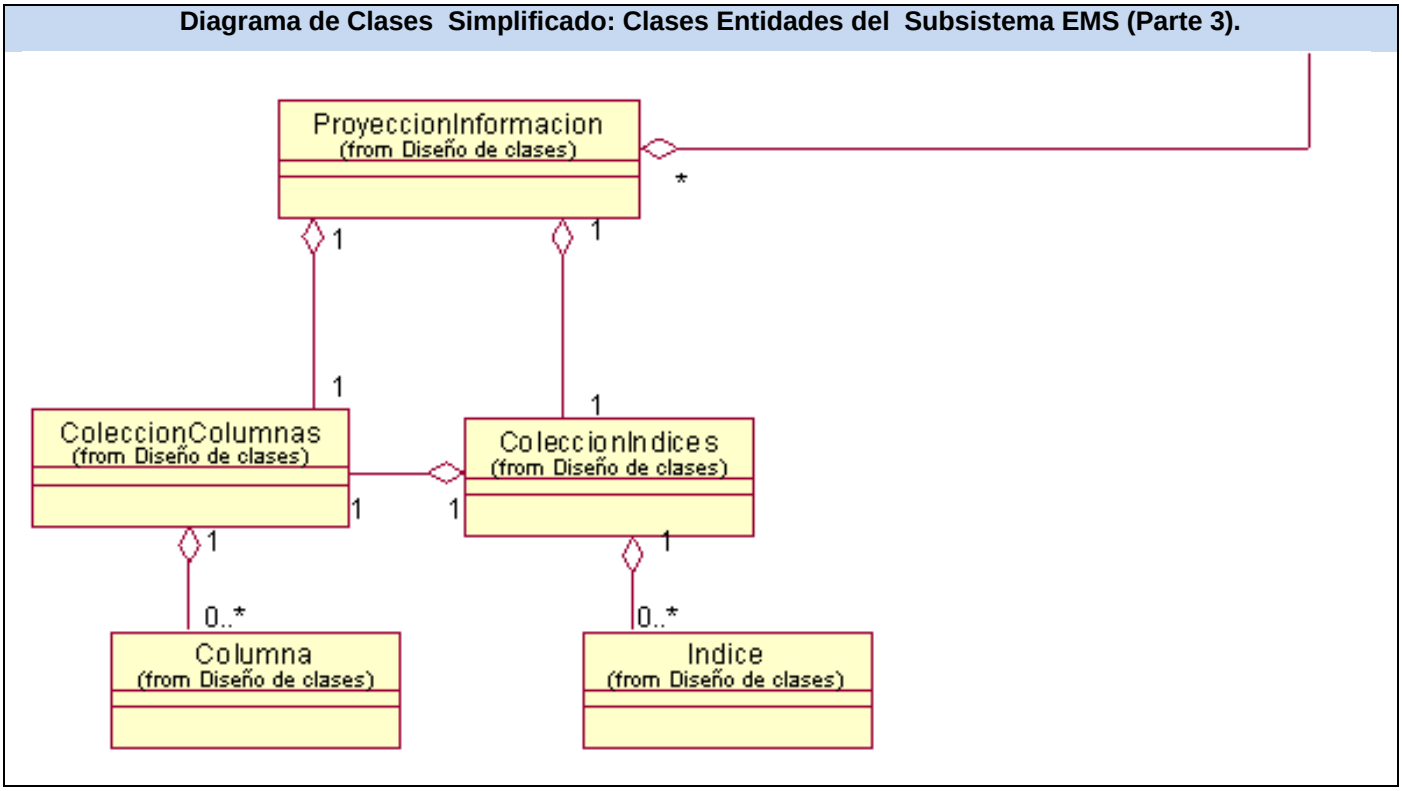

Figura 3.12 Diagrama de Clases Entidades del Subsistema EMS (Parte 3).

Diagrama de Clases Simplificado: Interfaz del Subsistema de Administración de Servicios y Aplicaciones. [19]

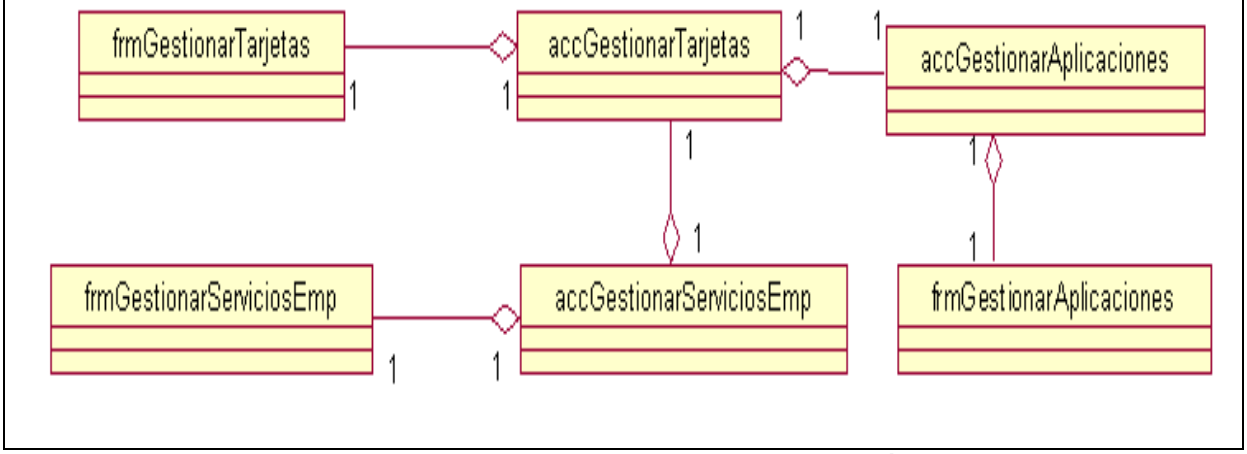

Figura 3.13 Diagrama de Clases Interfaz del Subsistema de Administración de Servicios y Aplicaciones.

[19] Ver Anexo 14 Descripción de las clases para la Interfaz de Usuario del Subsistema de Administración de Servicios y Aplicaciones.

Diagrama de Clases Simplificado: Clases Controladoras del Subsistema de Administración de Servicios y Aplicaciones.[20]

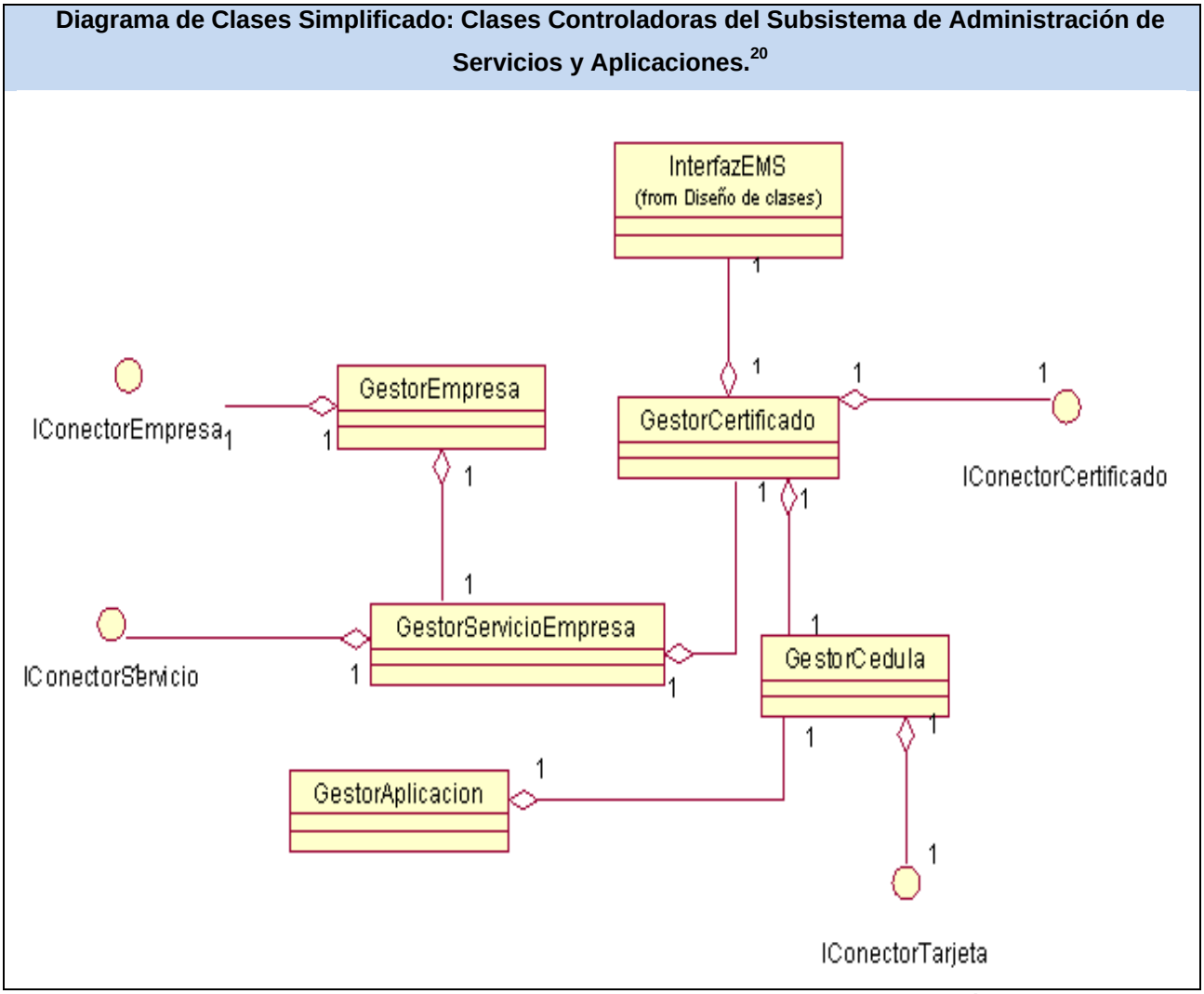

Figura 3.14 Diagrama de Clases Controladoras del Subsistema de Administración de Servicios y Aplicaciones.

[20] Ver Anexo 15 Descripción de las clases controladoras del Subsistema de Administración de Servicios y Aplicaciones.

Diagrama de Clases Simplificado: Clases Entidades del Subsistema de Administración de Servicios y Aplicaciones. [21]

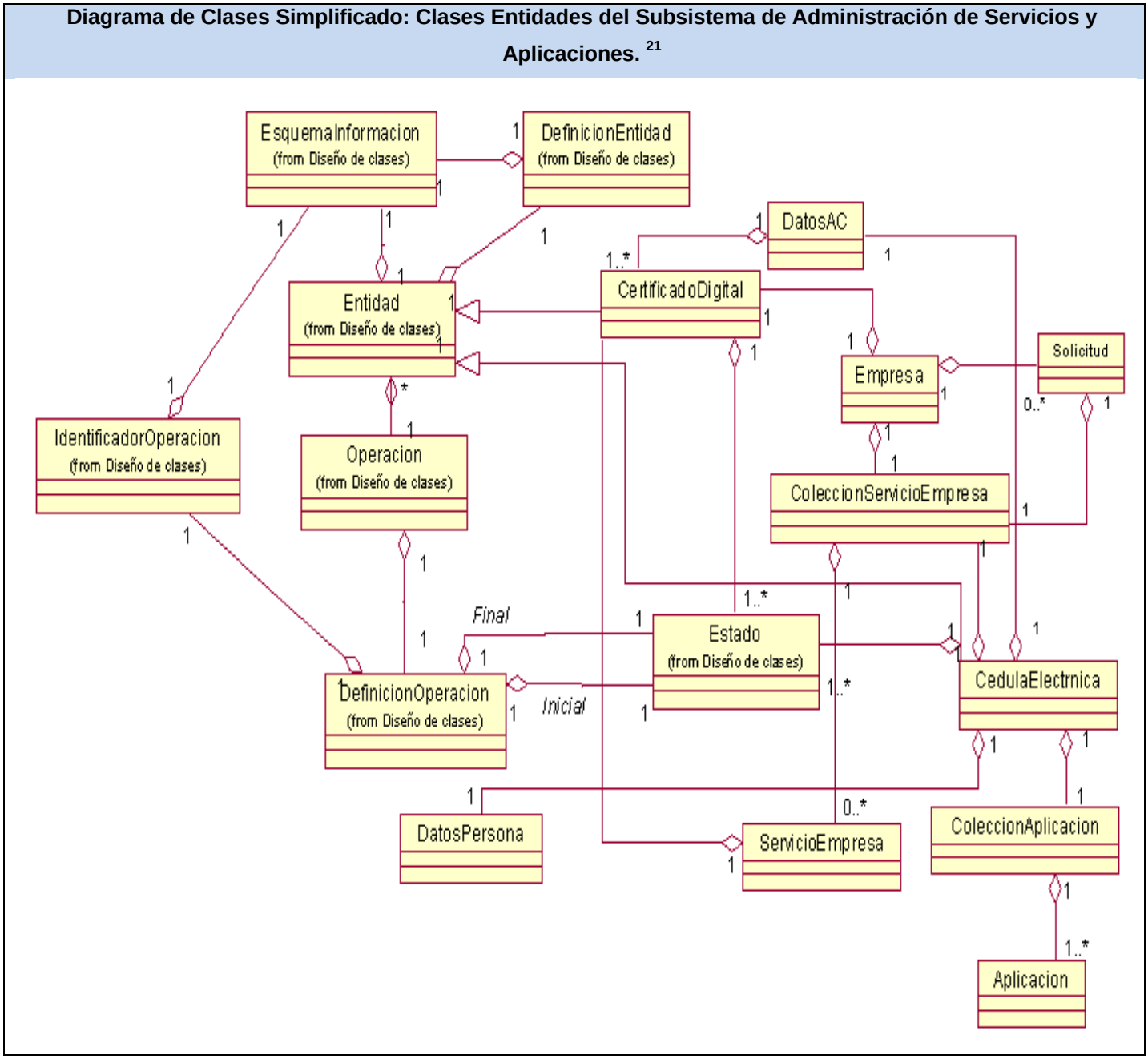

Figura 3.15 Diagrama de Clases Entidades del Subsistema de Administración de Servicios y Aplicaciones.

[21] Ver Anexo 16 Descripción de las clases entidades del Subsistema de Administración de Servicios y Aplicaciones.

3.3.3. Diagramas de interacción

Los diagramas de interacción se utilizan para estructurar los aspectos dinámicos de un sistema, conllevan a modelar instancias concretas o prototípicas de clases interfaces, componentes y nodos, así como los mensajes enviados entre ellos. Ilustran un comportamiento, describen la forma en que grupos de objetos colaboran para obtener un objetivo final y presenta una visión externa del sistema.

Diagrama de Secuencia CU Gestionar Notificación

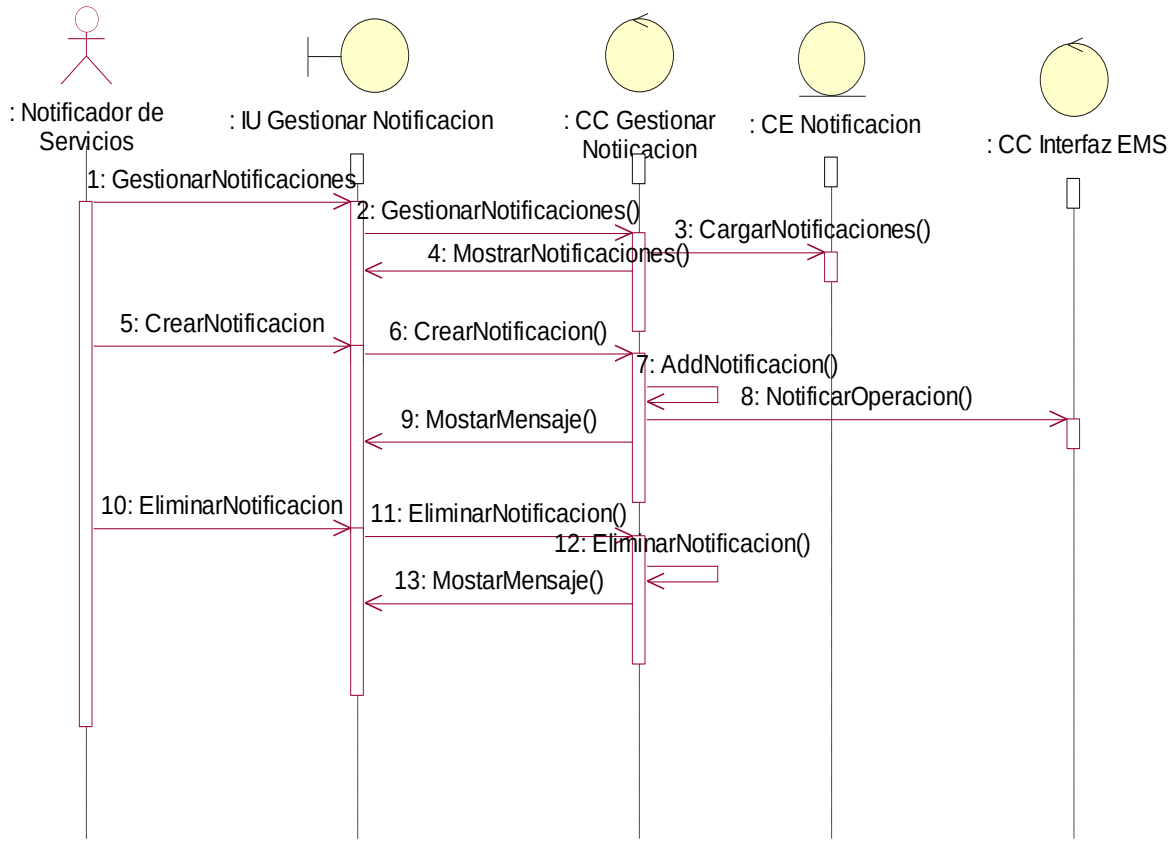

Figura 3.16 Diagrama de Secuencia CU Gestionar Notificación.

Diagrama de Secuencia CU Gestionar Subsistemas.

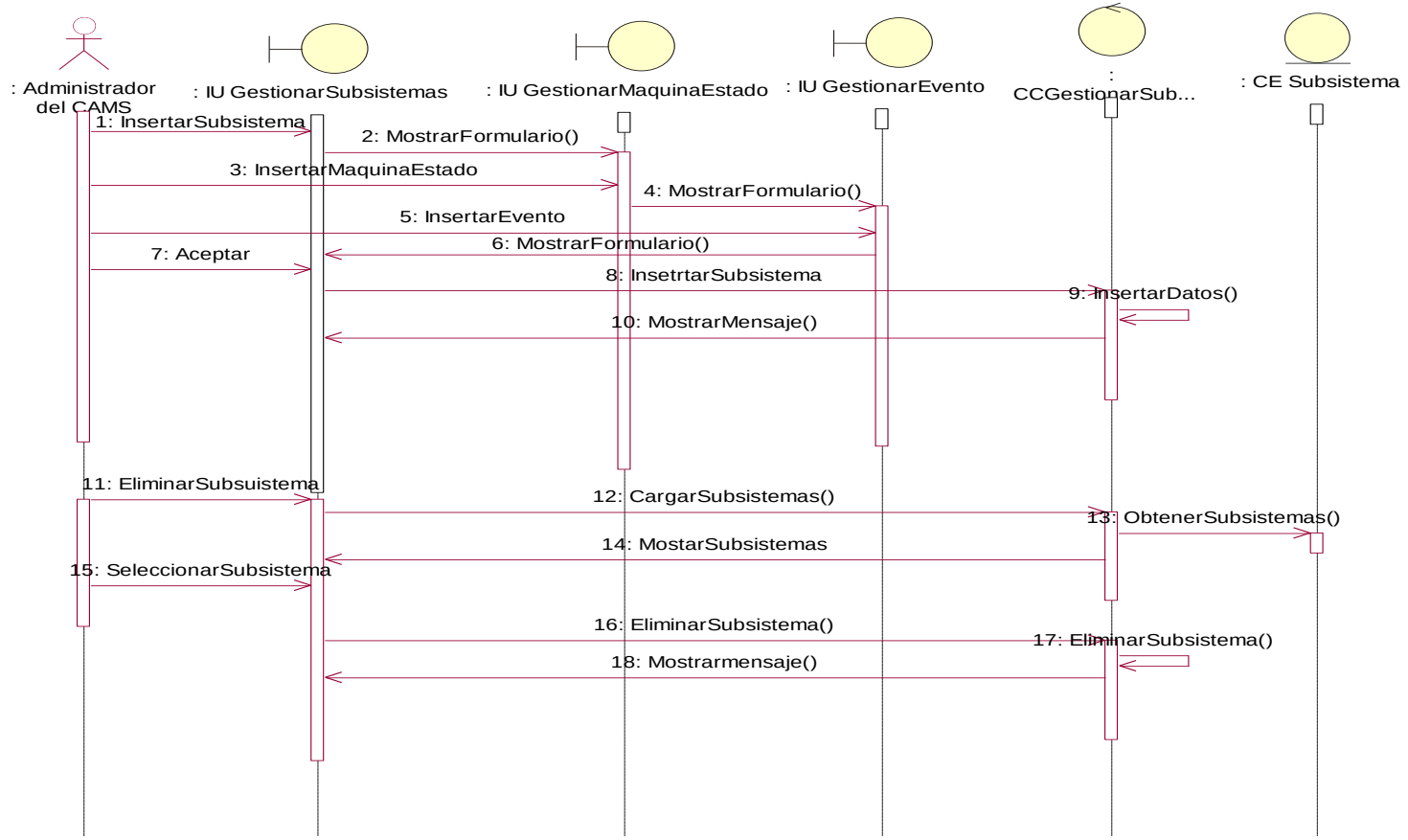

Figura 3.17 Diagrama de Secuencia CU Gestionar Subsistemas.

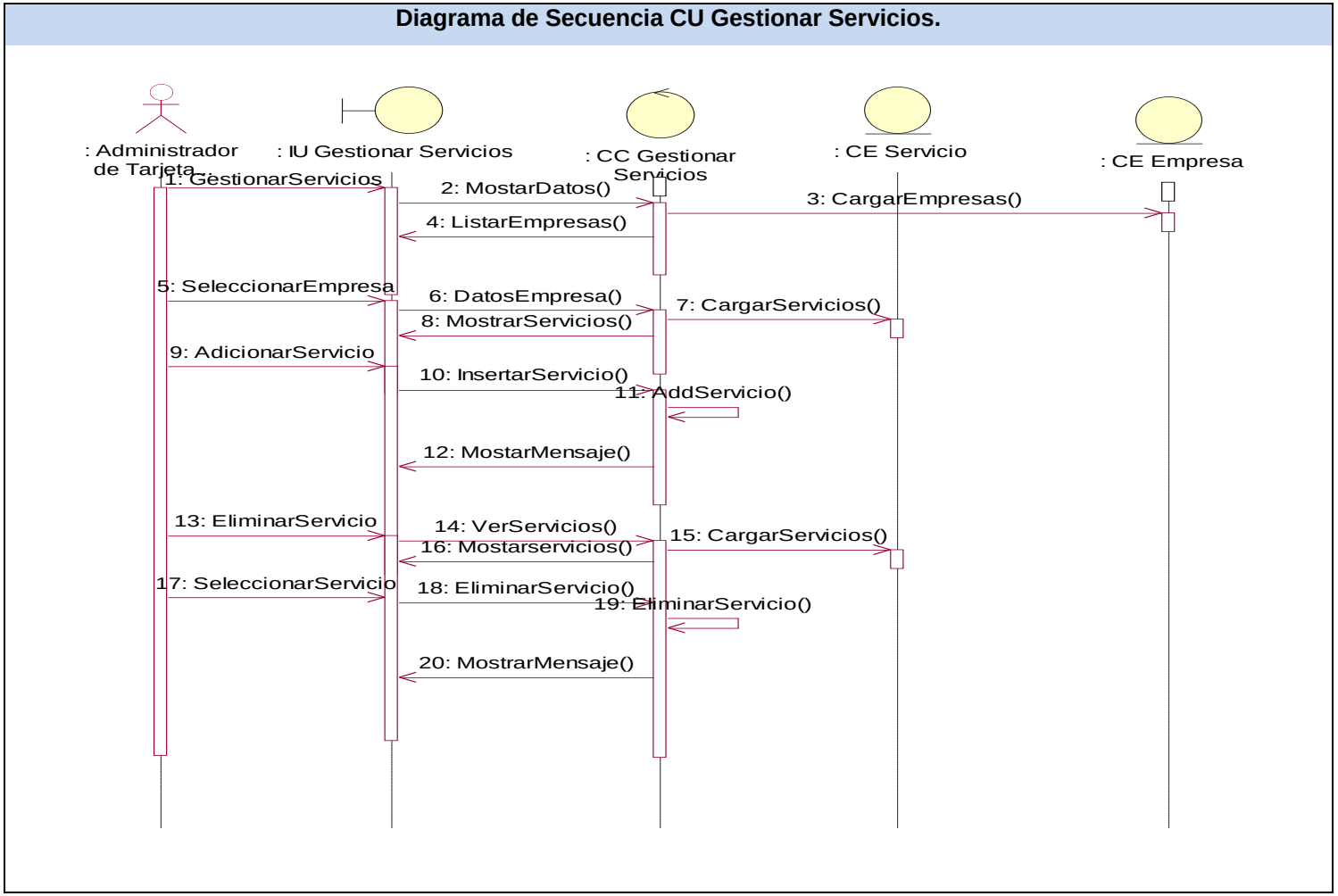

Figura 3.18 Diagrama de Secuencia CU Gestionar Servicios.

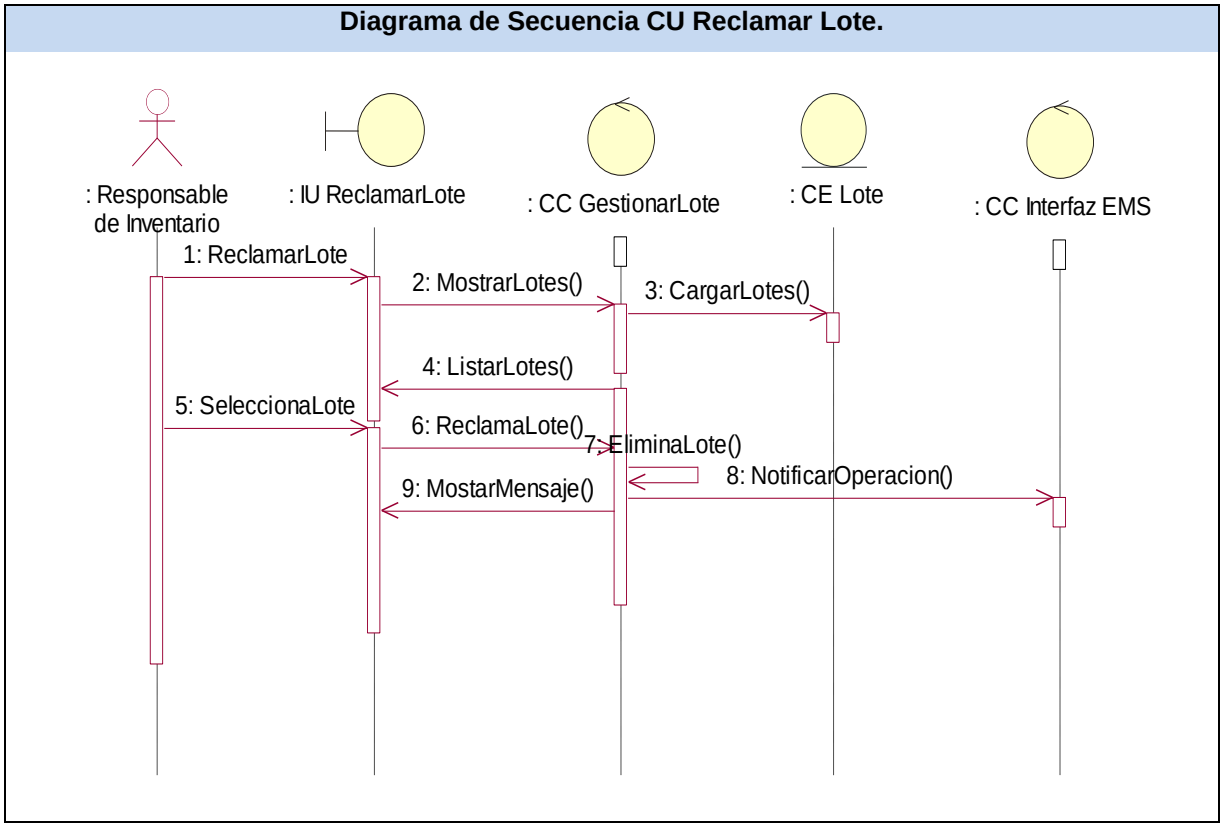

Figura 3.19 Diagrama de Secuencia CU Reclamar Lote.

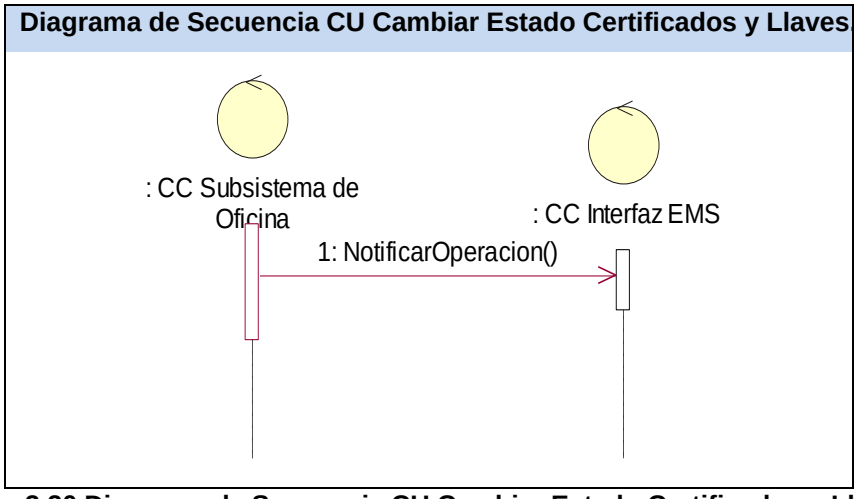

Figura 3.20 Diagrama de Secuencia CU Cambiar Estado Certificados y Llaves.

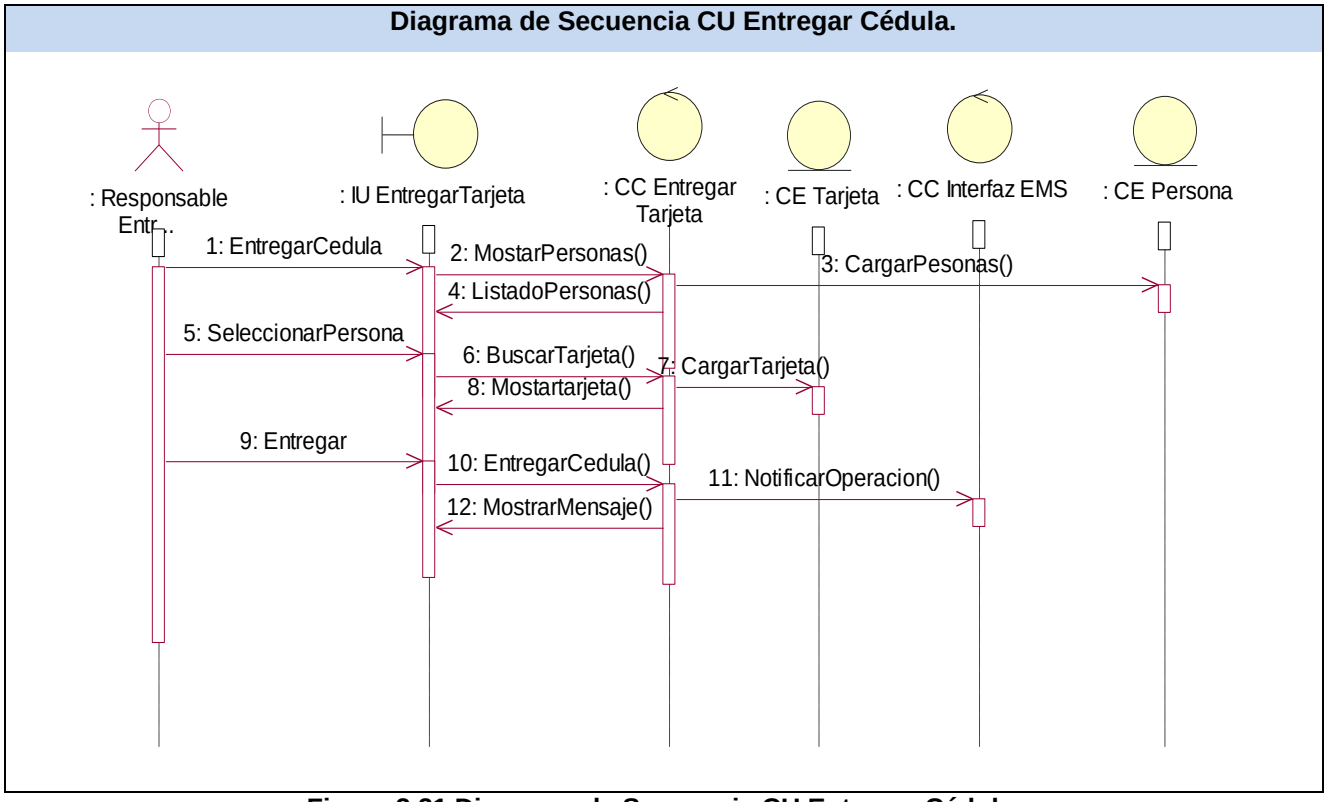

Figura 3.21 Diagrama de Secuencia CU Entregar Cédula.

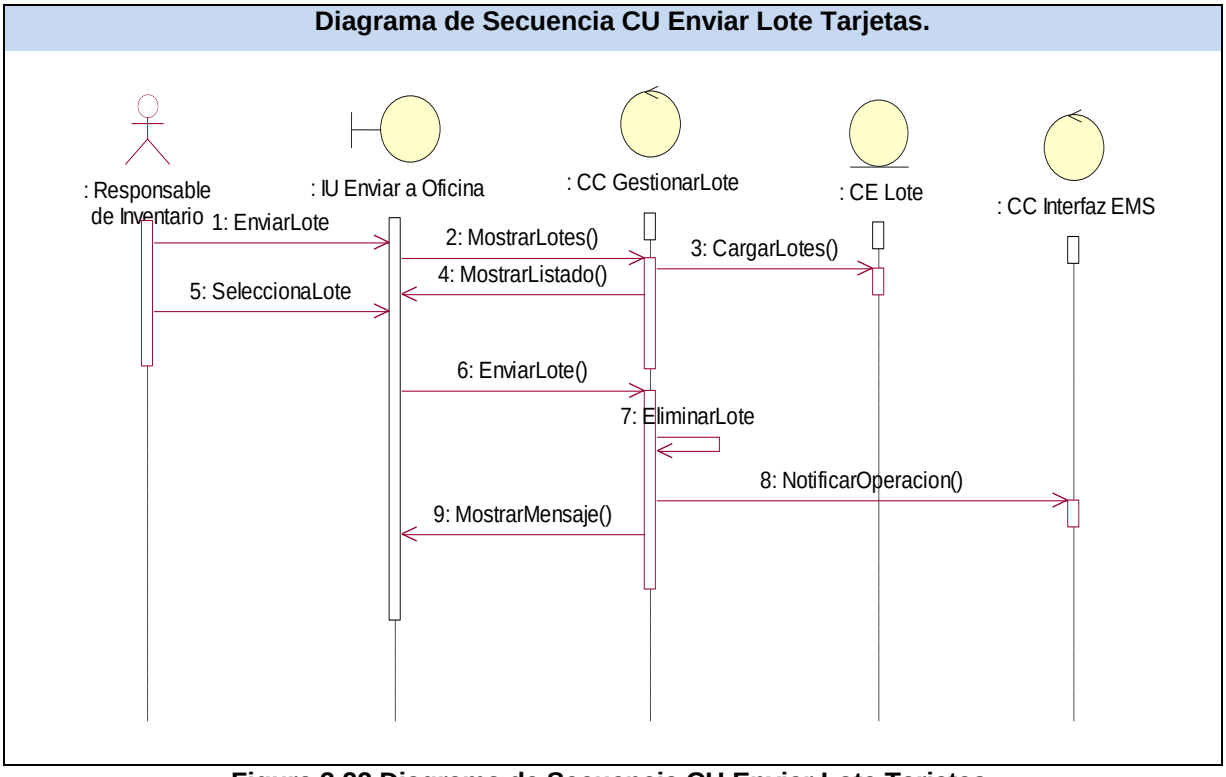

Figura 3.22 Diagrama de Secuencia CU Enviar Lote Tarjetas.

Diagrama de Secuencia CU Gestionar Aplicación.

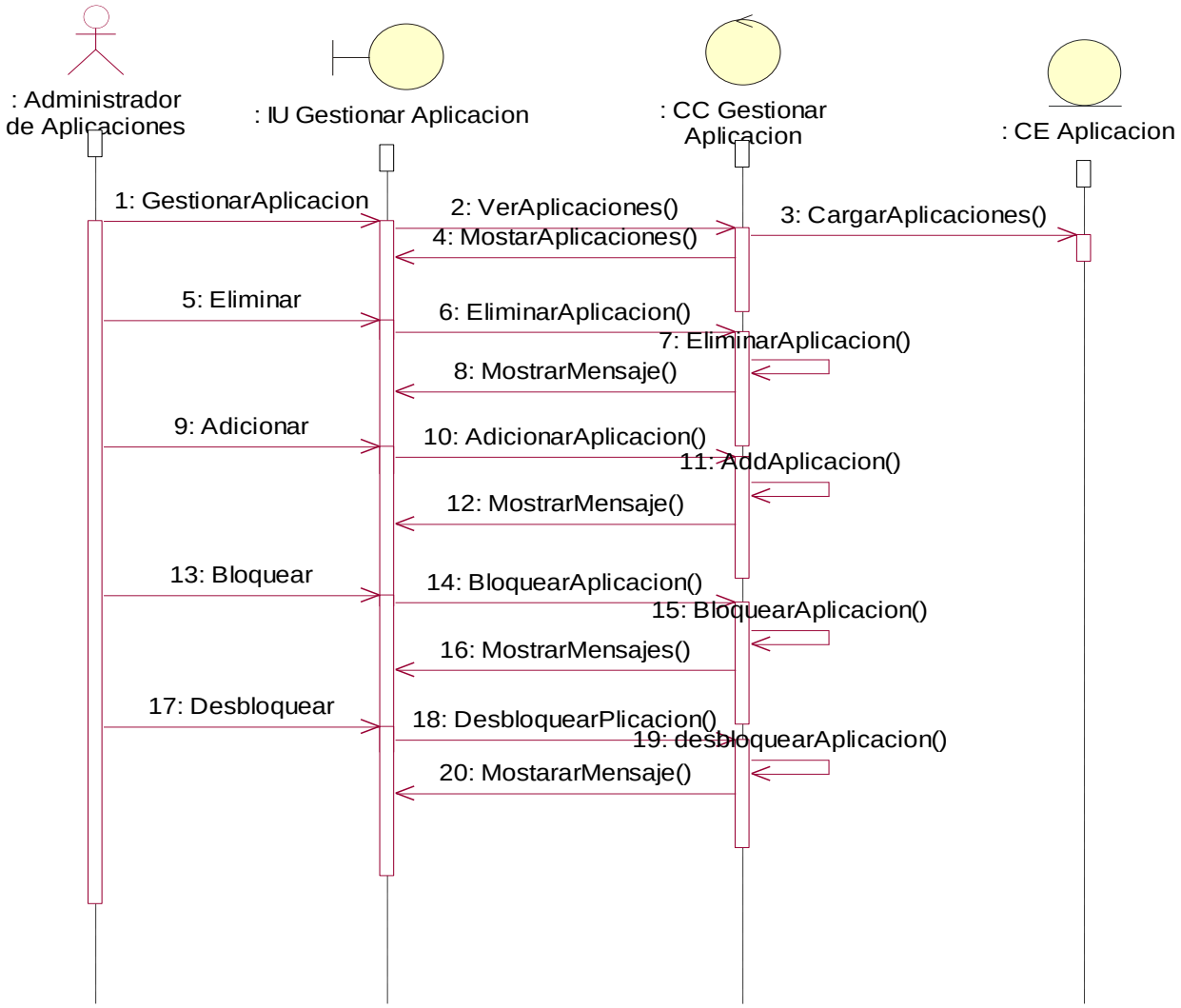

Figura 3.23 Diagrama de Secuencia CU Gestionar Aplicación.

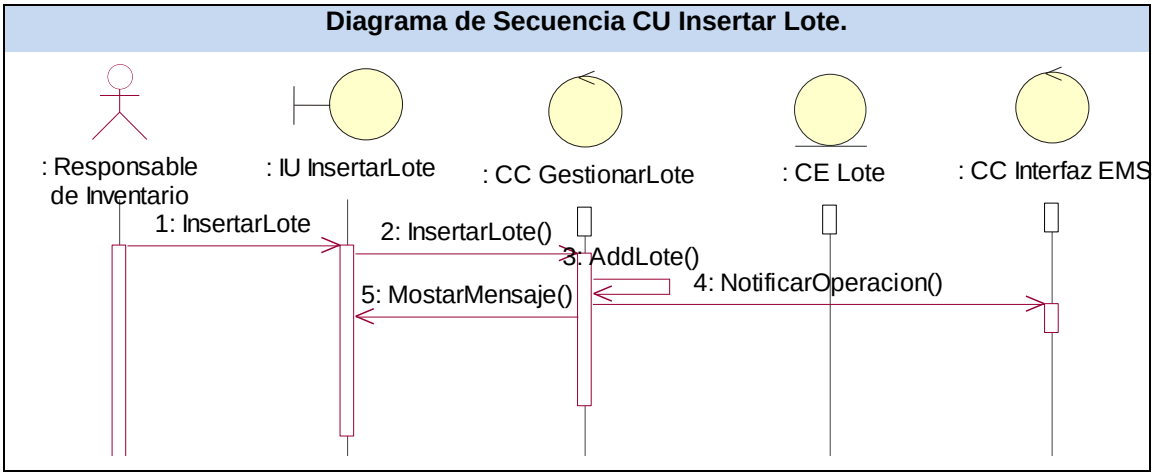

Figura 3.24 Diagrama de Secuencia CU Insertar Lote.

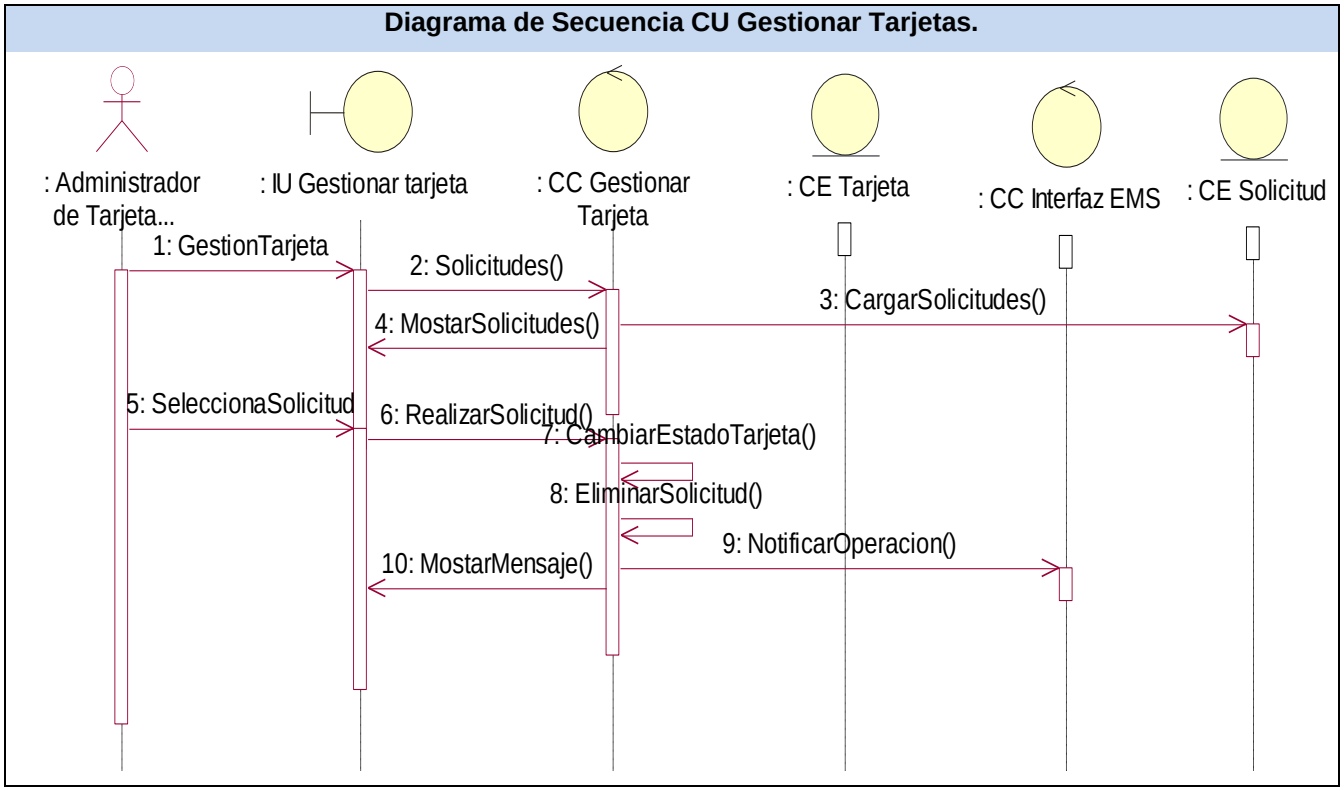

Figura 3.25 Diagrama de Secuencia CU Gestionar Tarjetas.

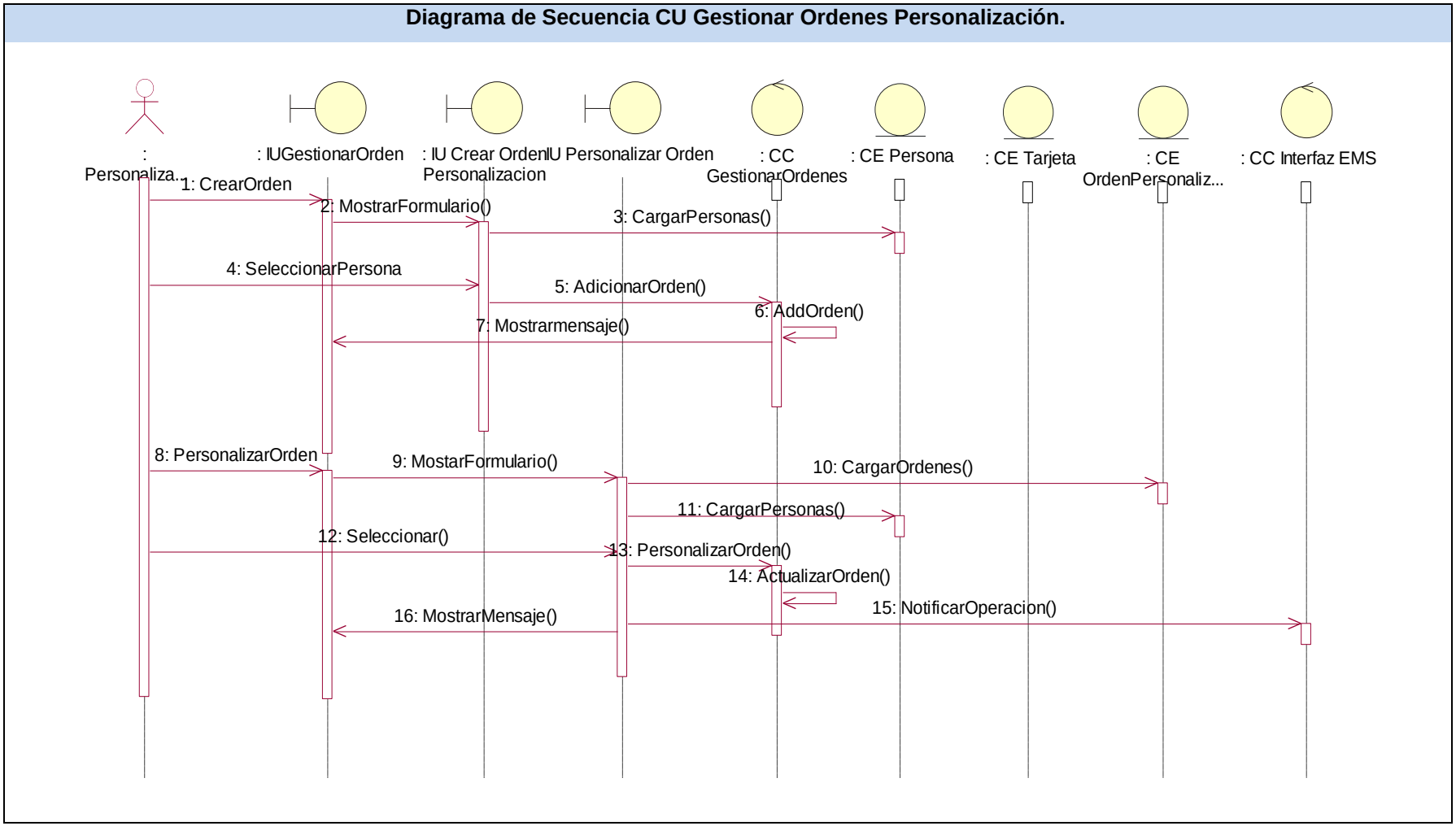

Figura 3.26 Diagrama de Secuencia CU Gestionar Ordenes Personalización.

3.3.4. Diseño de Bases de Datos

Uno de los pasos cruciales en la construcción de una aplicación que maneje una base de datos, es sin duda, el diseño de la misma; a continuación se presentan los modelos entidad - relación necesarios para la gestión del ciclo de vida y las aplicaciones de la cédula de identidad electrónica de la República Bolivariana de Venezuela.

3.3.4.1. Modelos Entidad Relación

Los diagramas o modelos entidad-relación (denominado por su siglas, E-R "Entity relationship") son una herramienta para el modelado de datos de un sistema de información. Estos modelos expresan entidades relevantes para el sistema; sus relaciones y propiedades.

Modelo Entidad - Relación del EMS.[22]

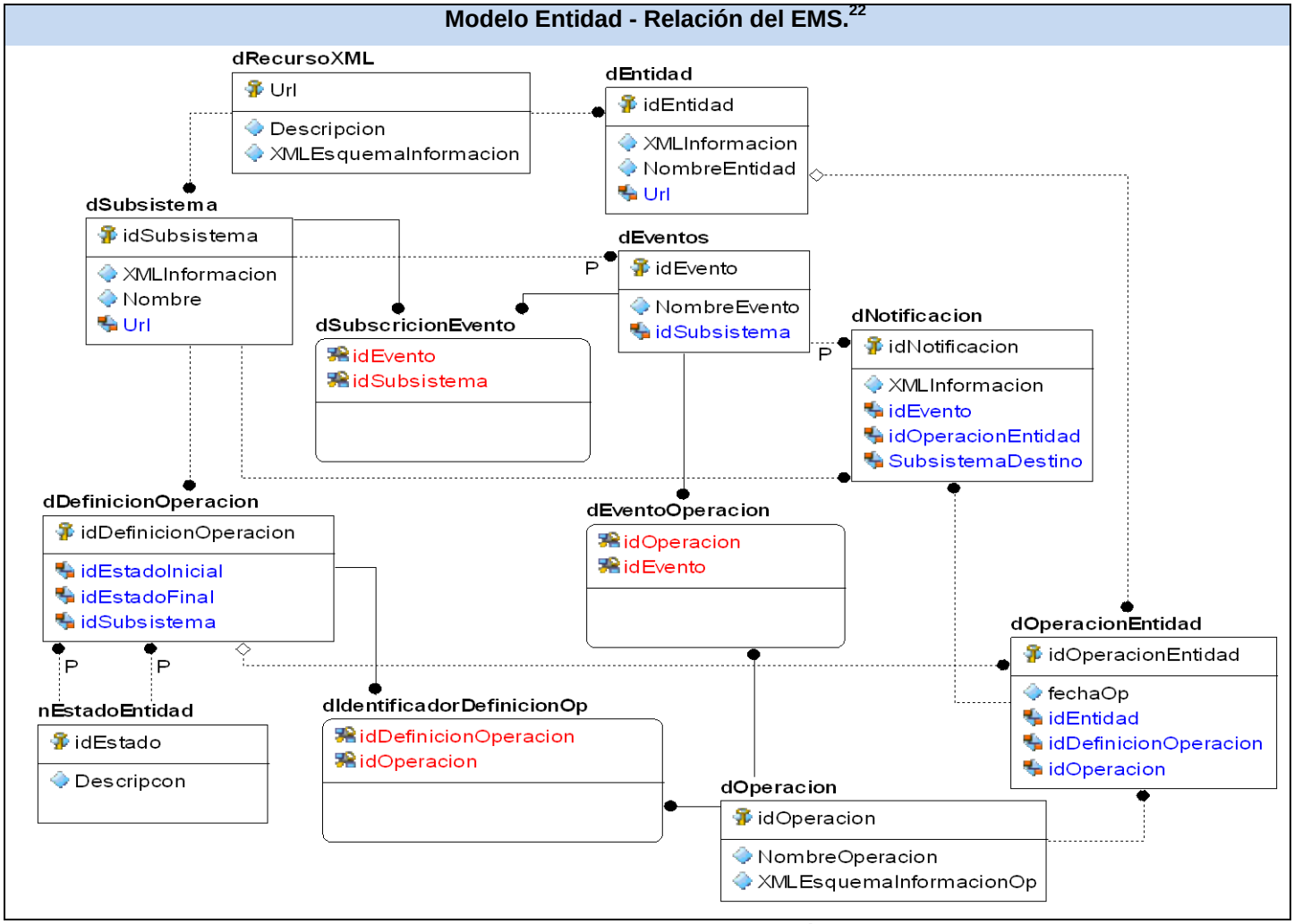

Figura 3.27 Modelo Entidad - Relación EMS.

[22] Ver Anexo 17 Descripción de las tablas del modelo entidad – relación del EMS.

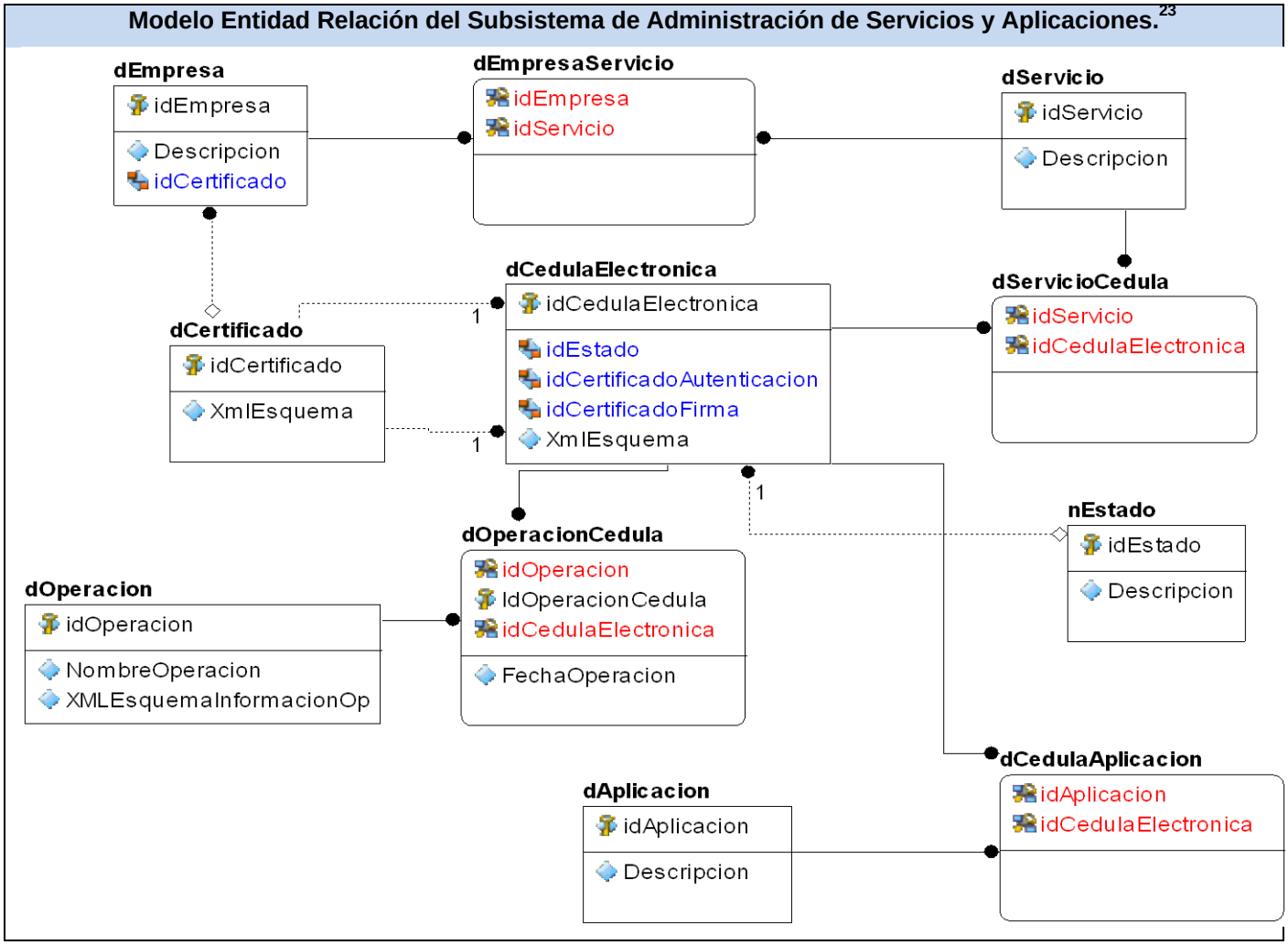

Figura 3.28 Modelo Entidad Relación del Subsistema de Administración de Servicios y Aplicaciones.

3.3.5. Definiciones de diseño

La interfaz gráfica del usuario es el medio por el cual este interactúa con el sistema, por lo que esta debe ser lo más amigable y clara posible; no solo para lograr uniformidad en la distribución de sus elementos sino también para que la persona que trabaje con ellas se sienta cómoda y logre adaptarse fácilmente a su ambiente de trabajo.

3.3.5.1. Pautas que se proponen para la confección de las interfaces de usuarios

- Diseñar para 1024 X 768.
- Debe solo verse lo que el usuario puede usar en ese momento y no otras opciones.

[23] Ver Anexo 18 Descripción de las tablas del modelo entidad – relación del Subsistema de Administración de Servicios y Aplicaciones.

- Solamente una acción a la vez.
- Para cambiar de acción el usuario debe decidir que hacer con la que tiene en curso.
- En caso de tener que usar varios formularios al mismo tiempo, se deben agrupar en hojas, permitiendo un acceso aleatorio a cada hoja.
- La aplicación cuenta con 6 áreas básicas menú, logo, trabajo, hojas, listado, modal.
- El área de menú es un rectángulo alargado en la parte izquierda y tendrá solo dos niveles, menú y submenú.
- El área de logo es un rectángulo pequeño en la parte inferior izquierda que contiene una imagen del logo de la oficina o la sede central.
- El área de trabajo es lo que resta en la parte derecha que se compone de dos partes hojas y listado.
- El área de hojas contiene toda la entrada y los eventos para el negocio del sistema.
- El área de listado contiene un listado producto de una búsqueda.
- El área modal tiene las funcionalidades y estructura de un listado solo que aparece en un formulario centrado y aparte y solo este puede ser accesible en la aplicación hasta tanto no se cierre.
- Los controles que tengan estrecha relación deben ser agrupados en un Panel.
- En el área de búsqueda debe poder verse como mínimo 10 registros antes de usar scroll. Si tiene que ser más pequeño por la cantidad de controles en el área de trabajo entonces utilizar una ventana modal.
- Cualquier área de búsqueda debe llevar paginación, restringidas a 20 registros como máximo.
- Las ventanas modales solo son para mostrar un listado, seleccionar un elemento o cerrarlas.
- En las ventanas modales deben poder verse de 10 a 30 registros antes de usar scroll.
- Cualquier área de búsqueda modal debe llevar paginación, restringidas a 50 registros como máximo.
- Cualquier imagen debe aparecer en el tamaño estándar del formato legal, como ejemplo foto de cédula.

La interfaz de usuario del sistema propuesto, es a través de formularios Windows y la pantalla principal se divide en tres áreas:

1. **Menú de acciones**. En el menú se muestran las acciones a las que tiene acceso el aperador que esté autenticado en la aplicación.
2. **Ícono de la aplicación**.
3. **Área de trabajo**. En esta área es donde salen los distintos formularios en dependencia de la acción que se ha seleccionado.

3.3.6. Tratamiento de errores

Para lograr que el sistema que se desarrolle sea de fácil soporte y mantenimiento, debe adoptarse una estrategia apropiada para el tratamiento de excepciones. Al diseñarse un sistema, se debe garantizar que este sea capaz de:

- Detectar las excepciones.
- Mostrar información sobre la excepción detectada.
- Crear trazas de las excepciones detectadas que puedan ser monitoreados y permitan una pronta detección del error y su rápida resolución.

El .Net Framework a través de sus clases nos permite realizar las operaciones de detección, encapsulamiento y propagación de excepciones. Así como métodos para la definición y obtención de los mensajes a mostrar al usuario.

Como el CAMS debe integrarse al Sistema SAIME su tratamiento de errores debe estar a cargo de la capa Sistema.Excepciones; desarrollada con este fin. Esta capa permite almacenar los errores en los trazas del sistema y en la base de datos, para que puedan ser auditados en caso de reportarse algún problema.

Sistema.Excepciones posee una interfaz única para mostrar los errores, lográndose una estandarización en la forma de presentarlos al usuario.

Pantalla de error del sistema.

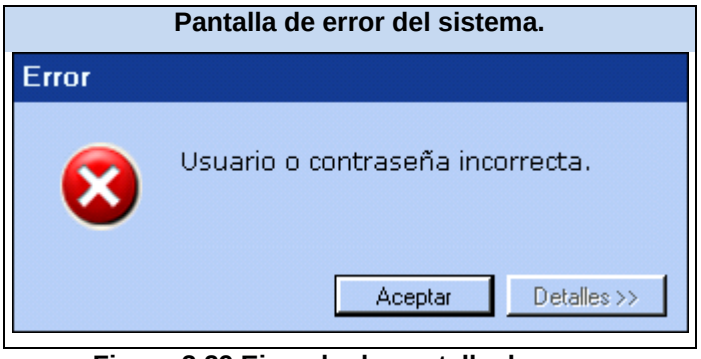

Figura 3.29 Ejemplo de pantalla de error.

3.3.7. Seguridad

La seguridad de la aplicación es uno de los requerimientos más importantes que debe ser cumplido. Está basada en las restricciones más comúnmente utilizadas en cualquier sistema que la implemente; el uso de los conceptos de autenticación y autorización, cimentados en requerir una cuenta de usuario válida y activa, además de un conjunto de roles que le permitirán al funcionario realizar sólo aquellas operaciones que se le han asignado. Este chequeo ocurre al iniciar la aplicación; realizándose las verificaciones de autenticación contra la base de datos y la comprobación de la autorización a las distintas funcionalidades del sistema.

De esta forma el usuario sólo tendrá acceso a las operaciones que se le asignen y permitirá aumentar la fiabilidad del sistema. El CAMS, como parte del Sistema SAIME, se basa en las restricciones de seguridad definidas en las primeras fases de análisis y diseño de este último. Para el acceso a la base de datos y para el registro de las operaciones de los usuarios que interactúan con ella, existen tanto usuarios físicos como lógicos que son registrados por el sistema, a estos usuarios se les asignan roles lógicos también registrados que se corresponden con los roles físicos de la base de datos; estos roles poseen permiso para la ejecución de los procedimientos almacenados que permiten el intercambio de datos. De esta forma el usuario sólo tiene los roles que le son asignados y con estos puede interactuar con los procedimientos almacenados a los cuales tenga permiso de ejecución a través de sus roles.

3.3.8. Concepción de la ayuda

Para la utilización del sistema adecuadamente y tener conocimiento de las acciones que se pueden realizar con el sistema se hace uso del manual de usuario, sistema de ayuda y manual de procedimientos concebidos desde la realización del sistema.

3.4. Conclusiones

En este capítulo se ha presentado el flujo de trabajo de más peso en la fase de Elaboración que propone la metodología RUP; se desarrolló una explicación detallada de la aplicación a través de los diferentes diagramas.

Entre los diagramas analizados se encuentran los de clases del análisis y diseño, los diagramas de interacción (de secuencia) y los correspondientes a la base de datos. Además se definieron los principios de diseño de la interfaz de usuario, seguridad y tratamiento de errores. A partir de este momento han quedado sentadas las bases para comenzar con la implementación del CAMS para la cédula de identidad electrónica de la República Bolivariana de Venezuela.

Conclusiones

En el desarrollo de este trabajo se ha especificado la documentación completa del sistema propuesto, abarcando desde un estudio de los sistemas de administración de tarjetas y aplicaciones existentes hasta el análisis y diseño del CAMS para la cédula de identidad electrónica de la República Bolivariana de Venezuela.

- Se ha demostrado la necesidad de crear un sistema que permita gestionar el ciclo de vida y las aplicaciones de la cédula de identidad electrónica.
- Se ha diseñado un sistema que se integra al Sistema SAIME, con iguales características de diseño y programación.
- Se ha profundizado en el conocimiento de los procesos actuales y se propusieron mejoras que permitirán lograr mayor seguridad y eficiencia en el proceso de identificación.

  Se diseño un subsistema para el Sistema SAIME que permite gestionar las entidades y subsistemas relacionados a estas durante su ciclo de vida.

Recomendaciones

- Garantizar que en la etapa de implementación se cumpla con el diseño propuesto para garantizar cumplir con las características generales presentes en los CAMS.

- Lograr en la etapa implementación la integración total y desde la misma base con el Sistema SAIME: a nivel de clases, bases de datos y documentación para que todos los sistemas hablen un lenguaje común.

- Revisar en conjunto con los Especialistas de Procesos la integración de los diferentes procesos propios del CAMS dentro de los procesos automatizados en el Sistema SAIME. Esto se debe a que en el piloto hay que validar la implementación de los procesos una vez que el sistema se termine

- Ampliar el conjunto de programadores que intervienen en la implementación del CAMS con el objetivo de reducir el tiempo de implementación, cumpliendo los plazos previstos en el cronograma del Proyecto y para lograr que un grupo mayor de especialistas domine el sistema.

- Preparar la documentación con el objetivo de que se pueda realizar el proceso de transferencia tecnológica al cliente e integrarla documentación del Sistema SAIME

- Realizar el estudio de factibilidad para una implementación usando herramientas de software libre para que el CAMS pueda ser utilizado e integrado a otros sistemas de Identidad y no solo al Sistema SAIME.

Bibliografía Citada.

ActivIdentity. 2008. ActivIdentity. [En línea] enero de 2008. http://www.actividentity.com/products/actividcms__home.php.

DevelpmentTeam, Privacy International. 2008. Privacy International. [En línea] enero de 2008. http://www.privacyinternational.org/. 3.

Diccionario de la Real Academia de la Lengua Española. 2007. Diccionario de la Real Academia de la Lengua Española. [En línea] diciembre de 2007. http://buscon.rae.es/draeI/. 2.

Dirección de Informática de la ONIDEX. 2007. ONIDEX. [En línea] octubre de 2007. http://www.onidex.gov.ve/Mis_ident/mision_ident.php. 1.

Effing, Wolfgang Rankl and Wolfgang. 2008. *Smart Card Hand Book.* Munich : John Wiley & Sons, Ltd, 2008. 6.

Enrrique Vasquez Gallo, Carmen Sanches Avila. 2007. *VT_eID.* Madrid : Universidad Técnica de Madrid, 2007.

Europes Information Society. 2008. Europes Information Society. [En línea] enero de 2008. http://ec.europa.eu/information_society/eeurope/i2010/index_en.htm. 4.

RSA Security. 2008. RSA . [En línea] enero de 2008. http://www.rsa.com/node.aspx?id=2972.

Safe Net. 2008. Safe Net Oficial Web Site. [En línea] enero de 2008. http://www.safenet-inc.com/products/cms/index.asp.

Siemens. 2008. SIEMENS. [En línea] enero de 2008. http://www.medical.siemens.com/webapp/wcs/stores/servlet/CategoryDisplay~q_catalogId~e_-11~a_categoryId~e_1009810~a_catTree~e_100010,1008631,1009812,1009807,1009806,1009810~a_langId~e_-11~a_storeId~e_10001.htm.

Smart Cards Alliance. 2008. Smart Cards Alliance. [En línea] enero de 2008. http://www.smartcardalliance.org/pages/smart-cards-intro-glossary. 5.

Smart Cards Forum. 2003. *JAVA CARD MANAGEMENT SPECIFICATION.* 2003.

Bibliografía Consultada.

ActivIdentity. 2008. ActivIdentity. [En línea] enero de 2008. http://www.actividentity.com/products/actividcms__home.php.

DevelpmentTeam, Privacy International. 2008. Privacy International. [En línea] enero de 2008. http://www.privacyinternational.org/. 3.

Diccionario de la Real Academia de la Lengua Española. 2007. Diccionario de la Real Academia de la Lengua Española. [En línea] diciembre de 2007. http://buscon.rae.es/draeI/. 2.

Dirección de Informática de la ONIDEX. 2007. ONIDEX. [En línea] octubre de 2007. http://www.onidex.gov.ve/Mis_ident/mision_ident.php. 1.

Effing, Wolfgang Rankl and Wolfgang. 2008. *Smart Card Hand Book.* Munich : John Wiley & Sons, Ltd, 2008. 6.

Enrrique Vasquez Gallo, Carmen Sanches Avila. 2007. *VT_eID.* Madrid : Universidad Técnica de Madrid, 2007.

Europes Information Society. 2008. Europes Information Society. [En línea] enero de 2008. http://ec.europa.eu/information_society/eeurope/i2010/index_en.htm. 4.

RSA Security. 2008. RSA . [En línea] enero de 2008. http://www.rsa.com/node.aspx?id=2972.

Safe Net. 2008. Safe Net Oficial Web Site. [En línea] enero de 2008. http://www.safenet-inc.com/products/cms/index.asp.

Siemens. 2008. SIEMENS. [En línea] enero de 2008. http://www.medical.siemens.com/webapp/wcs/stores/servlet/CategoryDisplay~q_catalogId~e_-11~a_categoryId~e_1009810~a_catTree~e_100010,1008631,1009812,1009807,1009806,1009810~a_langId~e_-11~a_storeId~e_10001.htm.

Smart Cards Alliance. 2008. Smart Cards Alliance. [En línea] enero de 2008. http://www.smartcardalliance.org/pages/smart-cards-intro-glossary. 5.

Smart Cards Forum. 2003. *JAVA CARD MANAGEMENT SPECIFICATION.* 2003.

Glosario de Términos.

SAIME: Servicio Autónomo de Identificación, Migración y Extranjería.

ONIDEX: Oficina Nacional *de* Identificación *y* Extranjería.

Chip: conjunto de circuitos electrónicos comprimidos en una pastilla de silicio.

Verificación biométrica: Medio por el cual una persona puede ser identificada por la evaluación de uno o más atributos biológicos distintivos.

Smart Card (Tarjeta Inteligente): Cualquier tarjeta del tamaño pequeño con circuitos integrados incluidos que permitan la ejecución de cierta lógica programada.

Microprocesador: es un circuito integrado que contiene todos los elementos necesarios para conformar una "unidad central de procesamiento" UCP.

CPU (Central Processing Unit): Es el componente en una computadora digital que interpreta las instrucciones y procesa los datos contenidos en los programas de computadora.

ROM (Read-Only Memory): Una memoria de semiconductor destinada a ser leída y no destructible, es decir, que no se puede escribir sobre ella y que conserva intacta la información almacenada, incluso en el caso de que se interrumpa la corriente (memoria no volátil).

EEPROM: Es un tipo de memoria ROM que puede ser programado, borrado y reprogramado eléctricamente, a diferencia de la EPROM que ha de borrarse mediante rayos ultravioletas.

RAM: Es un tipo de memoria temporal que pierde sus datos cuando se queda sin energía (por ejemplo, al apagar la computadora), por lo cual es una memoria volátil.

PKI (Infraestructura de llave pública): Es una combinación de hardware y software, políticas y procedimientos de seguridad que permiten la ejecución con garantías de operaciones criptográficas como el cifrado, la firma digital o el no repudio de transacciones electrónicas.

JavaCard: Es una tecnología que permite ejecutar de forma segura pequeñas aplicaciones Java (applets) en tarjetas inteligentes y dispositivos similares.

JCRE (JavaCard Rumtime Evironmetn): Describe el comportamiento del entorno de desarrollo de JavaCard.

API (Interfaz de Programación de Aplicaciones**)**: es el conjunto de funciones y procedimientos (o métodos si se refiere a programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

Applet: Es un componente de una aplicación que se ejecuta en el contexto de otro programa, por ejemplo un navegador web.

ATR (Answer To Reset): Describe la primera comunicación entre el chip de la tarjeta y el lector, luego de que el lector ha indicado el Reset a la tarjeta.

Biometría: Es el estudio de métodos automáticos para el reconocimiento único de humanos basados en uno o más rasgos conductuales o físicos intrínsecos. El término se deriva de las palabras griegas "bios" de vida y "metron" de medida.

Match On Card: Tecnología que permite realizar la verificación biométrica dentro del chip de la tarjeta.

Identidad Digital: Es el conjunto de rasgos que caracterizan a un individuo o colectivo en un medio de transmisión digital.

ID-1: Formato que especifica el tamaño de las tarjetas.

DNI: Documento de identificación nacional.

PIN (Personal Identification Number): Valor numérico usado para identificarse y poder tener acceso a ciertos sistemas o artefactos.

VPN (Virtual Private Network): es una tecnología de red que permite una extensión de la red local sobre una red pública o no controlada, como por ejemplo Internet.

USB (bus universal en serie): Es un puerto que sirve para conectar periféricos a una computadora.

UML: Lenguaje estandarizado de especificación visual para la modelación de objetos.

RUP (Rational Unified Process)**:** Es un proceso iterativo de desarrollo de software.

CLR (Common Language Runtime): Es el componente de máquina virtual del .NET Framework de Microsoft. Es la implementación del estándar Common Language Infrastructure (CLI) que define un ambiente de ejecución para los códigos de los programas.

MSIL (Microsoft Intermediate Language)**:** Es el lenguaje de programación legible por humanos de más bajo nivel en el Common Language Infrastructure y en él .NET Framework.

EMS: Entity Management System.

CMS: Card Management System.

CAMS: Card and Applications Management System.

Framework: Es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado.

DLL (Dynamic Linking Library): Término con el que se refiere a los archivos con código ejecutable que se cargan bajo demanda del programa por parte del sistema operativo.

Anexos.

Anexo 1: Países que utilizan documentos de identificación en el mundo.

Arabia Saudita	Sudáfrica	Suiza
Argelia	Francia	Taiwán
Argentina	Grecia	Turquía
Austria	Hungría	Singapur
Bielorrusia	Italia	Uruguay
Bosnia	Indonesia	Ucrania
Brasil	Israel	Venezuela
Bélgica	Islandia	Vietnam
Bulgaria	Lituania	República Checa
Canadá	Madagascar	Rumania
Chile	Malasia	Rusia
China	Malta	Serbia
Chipre	Marruecos	Sir Lanka
Colombia	México	Estonia
Corea del Sur	Mozambique	Eslovaquia
Costa Rica	Pakistán	Finlandia
Croacia	Perú	España
Cuba	Polonia	Egipto
El Salvador	Portugal	

Anexo 2: Medidas de seguridad física de las tarjetas inteligentes.

Feature Size: in 5 years' time the size of transistors and wires on the chip surface has shrunk from more than 1 µm to less than 200 nm. This size is too small for optical microscopes to analyze and too small for probe stations to put needles on. Sophisticated microscopes and Focused Ion Beams can still handle this size, though.

Multi-Layering: today's smartcard chips use multiple layers. Not only is the number of semiconductors that can be produced, larger, but is it also possible to hide sensitive data lines (buried layers) underneath other layers that contain less sensitive connections.

Protective Layer: in order to prevent analysis of live data processing it is possible to use a top layer that contains an active grid carrying a protection signal. Interruption of that signal will cause the chip to erase its memories and halt. However, skilled attackers might still be able to make a bypass through the grid and then penetrate the protective layer. Therefore, advanced grids would use a large number of seemingly non-correlated and frequently changing signals. This will significantly reduce the attacker's ability to access underlying lines by means of FIB modifications.

Sensors: signals that measure environment variables such as light, temperature, power supply and clock frequency can be used to disable the chip as soon as out-off-bound conditions are detected. This will reduce the attacker's possibility to do live data analysis on a prepared chip. On the other hand they may also affect the reliability of the chip and for that reason be tuned quite fault-tolerant.

Bus-Scrambling: the data bus between various building blocks (e.g. processors and memories) can be scrambled using a sophisticated non-constant scrambling technique. An attacker attempting to interpret the bus data needs to do a full reverse engineering of the scrambler logic.

Glue Logic: instead of placing functional blocks in separate sections on the chip it is also possible to mix it all up and create glue logic. This way an attacker will no longer be able to easily identify the functional building blocks by analyzing the physical structures on the chip.

Anexo 3: Sistemas nacionales de identificación que utilizan tarjetas inteligentes.

País	Fecha de inicio del sistema	Población total en 2006 (millones)	Tarjetas emitidas hasta septiembre 2006 (millones)
España	03/2006	43,76	0,1 (hasta noviembre)
Bélgica	03/2006	10,51	3,5
Estonia	01/2002	1,34	0,998
Finlandia	12/1999	5,25	0,12

Anexo 4: DNIe España.

Anexo 5: DNIe Estonia.

Anexo 6: Características de Sistemas de Administración de Tarjetas Inteligentes.

Características	Sistemas analizados											
	Safe Net	RSA Security	Mascot	Easy Card	ActivId	Siemens	Visu-Bank	Future Card	Pay Ware	Aci	Tranz Ware	Athena
Control de Inventario de tarjetas vírgenes.	x	x	x	x	x	x	x	x	x	x	x	x
Personalización de tarjetas vírgenes.	x	x	x	x	x	x	x	x	x	x	x	x
Administración del ciclo de vida de las tarjetas.	x	x	x	x	x	x	x	x	x	x	x	
Gestión a través de la Web.	x	x	x	x	x	x			x			x
Gestión a través de centros de llamadas.				x	x							
Personalizable	x	x	x		x						x	
Infraestructura de Llave Pública (PKI).	x				x	x						X
Tokens de identificación (iKey) USB.	x	x						x				x
Seguridad a través de roles.					x	x						
Gestión de auditorias.			x		x			x				

Características	Sistemas analizados											
	Safe Net	RSA Security	Mascot	Easy Card	ActivId	Siemens	Visu-Bank	Future Card	Pay Ware	Aci	Tranz Ware	Athena
Múltiples formas de lectura/escritura de la tarjeta (en línea, fuera de línea).				x	x							
Opera con varios tipos de tarjetas.			x		x			x	x	x		
Producción de sus propias tarjetas.				x								
Integración a sistemas similares.		x		x				x			x	
Cifrado VPN.	x					x	x					x

Anexo 7: Pasos Necesarios para tipos de trámites de cedulación.

Pasos Necesarios	Descripción
Captación de Datos	A partir de la entrevista y los documentos presentados por los ciudadanos se captan todos los datos necesarios para el trámite de cedulación. Incluye asistencia al proceso por parte de un fiscal de CNE (Consejo Nacional Electoral) y un Coordinador de Oficina y consulta a las bases de datos de Identidad.
Captación de Imágenes	Captura de huellas, firmas y fotografías. A partir de las huellas captadas se realizará el chequeo dactiloscópico automatizado contra las bases de datos de huellas existentes.
Supervisión	Reúne varias funcionalidades para el control y la supervisión de los trámites, entre ellas: impresión de la planilla de control, aprobación de los documentos, datos e imágenes y chequeo del estado de los trámites y sus pasos mediante un panel control.
Chequeo dactiloscópico	Proceso automatizado transparente al usuario y mediante el cual se realiza la verificación de la identidad del ciudadano a través de la comparación de las huellas captadas durante el proceso de captación de imágenes, evitando así posibles casos de fraude o usurpación de identidad. Este paso se realiza en ayuda del AFIS (acrónimo de Automated Fingerprint Identification System, `Sistema Automatizado de identificación de Huellas Digitales`).
Asignación de número de Cédula	En el caso de un trámite Original se realiza la asignación de un número de cédula dentro del rango de seriales disponibles para venezolanos y extranjeros.
Impresión de Documentos	Se realiza la impresión de las cédulas que se encuentran listas para este paso en la oficina correspondiente.
Entrega de Documentos	Permite que se controle la entrega de cédulas a los ciudadanos que terminaron satisfactoriamente el proceso.

Anexo 8: Descripción de las operaciones que deben registrar los subsistemas que forman el CAMS.

Subsistema de Inventario.	
Operaciones	**Descripción**
Inserción en el sistema de lotes de tarjetas inicializadas	Para realizar esta operación, el Responsable de Inventario del CPID accede al Subsistema de Inventario donde se le muestra la estructura que se ha definido en su almacén y donde se encuentran las posibles ubicaciones que puede utilizar para guardar uno o varios lotes de tarjetas, esta sección del sistema le permite además visualizar los lotes que ya tiene registrado en su inventario, una vez que este funcionario ha consultado los datos anteriores procede a insertar el nuevo lote.
Reclamación de lotes de tarjetas inicializadas al proveedor	Esta operación es registra cuando se descubre un lote o una tarjeta que esta en mal estado , ya sea rota o con malfuncionamientos, entonces el Responsable de Inventario del CPID accede al Subsistema de Inventario donde se le muestra la estructura que se ha definido en su almacén, selecciona el lote o lotes en mal estado y realiza la reclamación que consiste en informarle al proveedor que un lote o varias tarjetas estaban en mal estado para esto se imprime un acta que es firmada por el responsable de Inventario CPID donde se plasman las cusas por las cuales se realiza la reclamación. Provoca un cambio de estado las entidades de tarjetas que forman el lote (de Inicializadas a Anuladas).
Anulación de lotes de tarjetas o tarjetas individuales	Esta operación se registra cuando el Responsable de Inventario decide que un lote o tarjeta no son actas para ser utilizadas en el proceso de personalización. Provoca un cambio de estado las entidades de tarjetas que forman el lote (de Inicializadas a Anuladas).
Envío a Oficina de lotes de tarjetas	Esta operación se registra cuando el Responsable de Inventario de CPID decide a que oficina enviar un lote de tarjetas que ha pasado el control de la calidad y asigna a este lote a dicha oficina. Provoca un cambio de estado las entidades de tarjetas que forman el lote (de Listas para Enviar a Oficina a Enviadas a Oficina).
Baja de lotes de tarjetas anuladas	Esta operación se registra cuando el funcionario autorizado decide destruir las tarjetas que se encontraban reportadas como anuladas en el Subsistema de Inventario. Provoca un cambio de estado las entidades de tarjetas que forman el lote (de Anuladas a De Baja).
En este subsistema luego de cada operación debe quedar registrado el estado actual de cada tarjeta.	
Subsistema de Personalización.	
Operaciones	**Descripción**
Personalización de lotes de tarjetas.	Esta operación se registra cuando los lotes de tarjetas son personalizados y las tarjetas están listas para pasar al control de la calidad. Provoca un cambio de estado las entidades de tarjetas que forman el lote (de Inicializadas a Personalizadas).
Anulación de lotes de tarjetas o tarjetas individuales.	Esta operación se registra cuando los lotes de tarjetas o tarjetas individuales sufren algún daño durante el proceso de personalización que impide que estas sean utilizadas como cédula de identidad. Provoca un cambio de estado las entidades de tarjetas (de Inicializadas a Anuladas).
Subsistema de Control de la Calidad.	
Operaciones	**Descripción**

Aprobación de Personalización de lotes de tarjetas o tarjetas individuales.	Esta operación se registra cuando los lotes de tarjetas o tarjetas individuales cumplen con los requisitos planteados por las normas de calidad y pueden ser utilizadas como cédula de identidad. Provoca un cambio de estado las entidades de tarjetas (de Personalizadas a Listas para Enviar a Oficina).
Anulación de lotes de tarjetas o tarjetas individuales.	Esta operación se registra cuando los lotes de tarjetas o tarjetas individuales no cumplen con los requisitos planteados por las normas de calidad lo que impide que estas sean utilizadas como cédula de identidad. Provoca un cambio de estado las entidades de tarjetas (de Personalizadas a Anuladas).
Subsistema de Oficinas.	
Operaciones	**Descripción**
Recepción de lotes de tarjetas o tarjetas individuales.	Esta operación se registra cuando los lotes de tarjetas o tarjetas individuales llegan correctamente al almacén. Provoca un cambio de estado las entidades de tarjetas (de Enviadas a Oficina a En Oficina).
Anulación de lotes de tarjetas o tarjetas individuales.	Esta operación se registra cuando los lotes de tarjetas o tarjetas individuales que llegan a la oficina se encuentran en mal estado y no es posible realizar la entrega a los ciudadanos. Provoca un cambio de estado las entidades de tarjetas (de Enviadas a Oficina a Anuladas).
Entrega de cédula de identidad al ciudadano.	Esta operación se registra cuando las tarjetas son entregadas a los ciudadanos. Provoca un cambio de estado las entidades de tarjetas (de En Oficina a Entregadas).
Subsistema de Administración de Servicios y Aplicaciones.	
Operaciones	**Descripción**
Anulación de Tarjetas.	Esta operación se registra cuando una tarjeta es reportada por un ciudadano como perdida, robada o con mal funcionamiento y se ha cumplido el tiempo definido por las autoridades para probar que lo indicado por el ciudadano es definitivo. Provoca un cambio de estado en las entidades de tarjetas (de Entregadas a Anuladas).
Bloqueo de Tarjetas.	Esta operación se registra cuando una tarjeta es reportada por un ciudadano como perdida, robada o cuando se detecta alguna anomalía en su uso y es necesario inhabilitarla temporalmente. Provoca un cambio de estado en las entidades de tarjetas (de Entregada a Bloqueada).
Desbloqueo de Tarjetas.	Esta operación se registra cuando una tarjeta que se encontraba bloqueada puede ser utilizada nuevamente. Provoca un cambio de estado en las entidades de tarjetas (de Bloqueada a Desbloqueada).

Anexo 9: Criterios para determinar la Complejidad de los actores del sistema.

Tipo de Actor	Descripción	Factor de Peso
Simple.	Otro sistema que interactúa con el sistema a desarrollar mediante una interfaz de programación (API, Application Programming Interface).	1
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

Anexo 10: Criterios para determinar la Complejidad de los casos de uso del sistema.

Tipo de Caso de Uso	Descripción	Factor de Peso
Simple	El Caso de Uso contiene de 1 a 3 transacciones.	5
Medio	El caso de uso contiene de 4 a 7 transacciones.	10
Complejo	El Caso de Uso contiene más de 8 transacciones.	15

Anexo 11: Descripción de las clases para la Interfaz de Usuario del Subsistema EMS.

Nombre: frmGestionarSubsistemas	
Tipo de clase: Interfaz	
Atributo	**Tipo**
Responsabilidades	
Nombre	**Descripción**
InicializeComponent()	Inicializa los controles del Formulario.
Dispose() : void	Libera Los controles del Formulario.
Show() : void	Muestra el Formulario.
frmGestionarSubsistema()	Constructor del Formulario.

Nombre: frmGestionarMaquinaEstado	
Tipo de clase: Interfaz	
Atributo	**Tipo**
Responsabilidades	
Nombre	**Descripción**
InicializeComponent()	Inicializa los controles del Formulario.
Dispose() : void	Libera Los controles del Formulario.
Show() : void	Muestra el Formulario.
frmGestionarMaquinaEstado ()	Constructor del Formulario.

Nombre: frmGestionarEventos	
Tipo de clase: Controladora	
Atributo	**Tipo**
Responsabilidades	
Nombre	**Descripción**
InicializeComponent()	Inicializa los controles del Formulario.
Dispose() : void	Libera Los controles del Formulario.
Show() : void	Muestra el Formulario.
frmGestionarEventos ()	Constructor del Formulario.

Nombre: accGestionarSubsistemas	
Tipo de clase: Interfaz	
Atributo	**Tipo**
_GestorSubsistema	GestorSubsistemas
_Subsistemas	ColeccionSubsistemas
_Formulario	frmGestionarSubsistemas
_AccionGestionarMaquinaEstado	accGestionarMaquinaEstado
_AccionGestionarEventos	accGestionarEventosSubsistema
Responsabilidades	
Nombre	**Descripción**
CrearForma() : void	Crea el objeto de la forma.
MostrarFormulario() : void	Muestra el formulario.
CrearSubsistema(Subsistema) : Subsistema	Crea un nuevo Subsistema.

EliminarSubsistema(Subsistema) : bool	Elimina el subsistema dado.
MostrarSubsistemas() : void	Muestra los subsistemas de la BD
MostrarEventosSubsistema(Subsistema) : void	Muestra los Eventos de un Subsistema dado.
MostrarMaquinaEstadoSubsistema(Subsistema) : void	Muestra la maquina de estado de un subsistema.
MostrarEntidadesSubsistema(Subsistema) : void	Muestra las entidades de un Subsistema.
EliminarEntidadSubsistema(Subsistema, Entidad) : bool	Elimina una entidad de un subsistema.
MostrarEstadosSubsistema(Subsistema) : void	Muestra los estados de un subsistema.
EliminarEstadoSubsistema(Subsistema, Estado) : bool	Elimina estados de un subsistema.

Nombre: accGestionarMaquinaEstado	
Tipo de clase: Controladora	
Atributo	**Tipo**
_GestorSubsistema	GestorSubsistemas
_MaquinaEstado	MaquinaEstado
_Formulario	frmGestionarMaquinaEstado
Responsabilidades	
Nombre	**Descripción**
CrearForma() : void	Crea el objeto de la forma.
MostrarFormulario() : void	Muestra el formulario.
MostrarTranciciones() : void	Muestra las definiciones de Operación de un subsistema.
CrearTransicionEstados(Estado ini, Estado fin, IdentificadorO	Crea una definición de operación para un subsistema.
MostrarOperaciones(Subsistema) : void	Muestra las operaciones de un subsistema.
MostrarEstados(Subsistema) : void	Muestra los estados de un subsistema.
EliminarOperacion(Operacion, Subsistema) : bool	Elimina una operación de un Subsistema.
EliminarTransicion(Estado ini, Estado fin) : bool	Elimina una definición de Operación de un subsistema.

Nombre: accGestionarEventosSubsistema	
Tipo de clase: Controladora	
Atributo	**Tipo**
_GestorSubsistema	GestorSubsistemas
_Eventos	ColeccionEventoSubsistema
_Formulario	frmGestionarEventos
Responsabilidades	
Nombre	**Descripción**
CrearForma() : void	Crea el objeto de la forma.
MostrarFormulario() : void	Muestra el formulario.
MostrarEventos(Subsistema) : void	Muestra los eventos de un subsistema.
Eliminarevento(Subsistema) : bool	Elimina un evento del subsistema dado.
MostrarOperaciones(Subsistema) : void	Muestra las operaciones de un subsistema.
EliminarOperacion(Operacion, Subsistema) : bool	Elimina una operación de un Subsistema.

Anexo 12: Descripción de las clases controladoras del Subsistema EMS.

Clases Colecciones.

Nombre: Coleccion	
Tipo de clase : Controladora	
Atributo	**Tipo**
This	Colección
Responsabilidades	
Nombre	**Descripción**
AddRango(ColeccionSubsistemas)	Adiciona al final de la colección existente otra que recibe como parámetro.
Add(Objeto)	Adiciona al final de la colección existente un objeto que recibe como parámetro.
ObtenerIndice(Objeto)	Retorna el índice, dentro de la colección, del objeto que recibe como parámetro.
Insertar(int, Objeto)	Inserta el objeto que recibe como parámetro en la posición indicada.
Eliminar(Objeto)	Elimina el objeto que recibe como parámetro.
Contiene(Objeto)	Retorna verdadero si el objeto que recibe como parámetro está contenido en la colección en caso contrario falso.

Esta clase varía en dependencia del tipo de objeto que forme la colección pero sus métodos son los mismos.

Clases Gestoras y Conectoras.

Nombre: GestorSubsistemas	
Tipo de clase : Controladora	
Atributo	**Tipo**
_Subsistemas	ColeccionSubsistemas
_GestorMaquinaE	GestorMaquinaEstado
_GestorEvento	GestorEventos
_GestorSubsistemaEstadoEntidad	GestorSubsistemaEstadoEntidad
Responsabilidades	
Nombre	**Descripción**
AdiconarSubsistema(Subsistema)	Adiciona el subsistema que recibe como parámetro a la colección _Subsistemas y a la Fuente de Datos a través de los métodos del _ConectorSubsistema.
CargarSubsistemas()	Carga la colección _Subsistemas de la fuente de datos la primera vez es invocado siempre retorna la colección que obtiene.
CargarMaquinaEstado(Subsistema)	Llena la máquina de estado del subsistema que recibe como parámetro y lo retorna.
ModificarSubsistema(Subsistema, Subsistema)	Modifica los datos de un subsistema con los datos de otro y retorna el modificado. Esta operación se realiza sobre uno de los que forman la colección _Subsistemas y se actualiza además la fuente de datos a través del _ConectorSubsistema.
EliminarSubsistema(Subsistema)	Elimina un subsistema de la colección _Subsistemas y de la fuente de datos a través del _ConectorSubsistema.
CargarEstados(Subsistema)	Carga los estados del subsistema que recibe como parámetro.
CargarOperaciones(Subsistema)	Carga las operaciones del subsistema que recibe como parámetro.
ActualizarMaquinaEstado(Subsistema,	Actualiza la máquina de estado del subsistema que recibe como

Nombre: GestorSubsistemas	
Tipo de clase : Controladora	
Atributo	**Tipo**
MaquinaEstado)	parámetro con los datos de la MaquinaEstado que recibe también, utiliza _ConectorSubsistema y el _GestorMaquinaE para actualizar la fuente de datos.
EliminarMaquinaEstado(Subsistema)	Elimina la máquina de estado del subsistema que recibe como parámetro y lo retorna, utiliza _ConectorSubsistema y el _GestorMaquinaE para actualizar la fuente de datos.
CargarEventos(Subsistema)	Carga los eventos del subsistema que recibe como parámetro y lo retorna.
CargarEventosSubscrito(Subsistema)	Carga los eventos a los cuales está subscrito el subsistema que recibe como parámetro y lo retorna.
EliminarEvento(Subsistema, EventoSubsistema)	Elimina el eventos que recibe como parámetro al subsistema indicado y lo retorna utiliza el _GestorEvento para actualizar los eventos de la fuente de datos.
EliminarEventoSubscrito(Subsistema, EventoSubsistema)	Elimina la subscripción al eventos que recibe como parámetro del subsistema indicado y lo retorna utiliza el _GestorEvento para actualizar los eventos de la fuente de datos.
AdicionarEvento(Subsistema, Evento)	Adiciona el evento al subsistema que recibe como parámetro y lo retorna utiliza el GestorEvento y el _ConectorSubsistema para actualizar la fuente de datos.
AdicionarEventoSubscrito(Subsistema, Evento)	Subscribe el subsistema que recibe como parámetro al evento indicado y lo retorna. Utiliza el GestorEvento y el _ConectorSubsistema para actualizar la fuente de datos.
CargarEntidades(Subsistema, Iformacion)	Carga las entidades y sus estados del subsistema que recibe como parámetro y lo retorna.
ActualizarEstadoEntidad(Subsistema, Entidad, Estado)	Actualiza los datos de la entidad y el estado del subsistema que recibe como parámetro y lo retorna. Utiliza el _ConectorSubsistema para actualizar la fuente de datos.
ConsumirNotificaciones(Subsistema)	Carga las notificaciones que tenga el subsistema que recibe como parámetro, lo retorna y utiliza el _GestorEvento para eliminar las notificaciones de la fuente de datos.
CrearNotificacion(Notificacion)	Crea una Notificacion y la retorna.
ConsultarCiclodeVida(Entidad, Subsistema)	Retorna un xml de información que contiene las operaciones y estrados que ha tenido la entidad que recibe como parámetro en el subsistema indicado.

Nombre: IConectorSubsistemas	
Tipo de clase: Interfaz, deben implementarlas las clases controladoras que interactúan con la fuente de datos.	
Responsabilidades	
Nombre	**Descripción**
AdicionaSubsistema(Subsistema)	Adiciona el subsistema que recibe como parámetro a la Fuente de Datos y lo retorna.
CargarSubsistemas()	Carga los subsistemas de la fuente de datos y retorna la colección.
ActualizarSubsistemas(Subsistema , Subsistema)	Modifica los datos de un subsistema con los datos de otro y retorna el modificado.
EliminarSubsistema(Subsistema)	Elimina el subsistema que recibe como parámetro

Nombre: GestorEventos	
Tipo de clase: Controladora.	
Atributo	**Tipo**
_EventosSubsistema	ColeccionEventosSubsistema
_GestorNotificaciones	GestorNotificaciones
_ConectorEvento	IConectorEvento
Responsabilidades	
Nombre	**Descripción**
CargarEventosSubsistema(Subsistema) : void	Carga de la BD los eventos de un subsistema dado.
CargaEventosSubscritos(Subsistema) : void	Carga de la BD los Eventos a los que esta subscrito un Subsistema dado.
EliminarOperacion(Operacion, EventoSubsistema) : EventoSubsistema	Elimina a un Evento una de las operaciones que lo dispara.
AdicionarOperacion(Operacion, EventoSubsistema) : EventoSubsistema	Adiciona una operación a un evento.
EliminarEvento(Subsistema, EventoSubsistema) : bool	Elimina un Evento a un subsistema dado.
AdicionarEvento(Subsistema, EventoSubsistema) : bool	Adiciona un evento a un Subsistema dado.
ConsumirNotificaciones(EventosSubsistema[]) : Notificacion[]	Consume del EMS las notificaciones de los eventos a los que esta subscrito un subsistema dado.

Nombre: IConectorNotificaciones	
Tipo de clase: Controladora.	
Responsabilidades	
Nombre	**Descripción**
EliminarNotificacion(EventoSubsistema) : bool	Elimina una notificación del EMS.
SalvarNotificacion(EventoSubsistema) : bool	Inserta en la BD una Notificacion.
CargarNotoficaion(Subsistema) : EventoSubsistema	Carga una Notificacion para un subsistema dado.

Nombre: InterfazEMS	
Tipo de clase: Controladora.	
Atributo	**Tipo**
_GestorSubsistema	GestorSubsistemas
Responsabilidades	
Nombre	**Descripción**
ConsultaraCiclodeVida(XML) : XMLCiclodeVida	Devuelve las operaciones realizadas a una entidad dada.
ConsumirNotificaciones(XMLInformacionNotificacion) : XMLInformacionNotificacion	Devuelve la notificación que coincide con el esquema dado.
ValidarConeccionSubsistema(NombreSubsistema : string) : bool	Verifica que el subsistema conectado realmente pertenece al CAMS.
NotificarOperaciones(XMLInformacion : string) : void	Notifica al EMS una operación realizada sobre una Entidad.

Nombre: IConectorEventos	
Tipo de clase: Controladora.	
Nombre	**Descripción**
EliminarEvento(EventoSubsistema) : bool	Elimina un evento de la BD entidad dada.
SalvarEvento(EventoSubsistema) : bool	Inserta Un evento en la BD
CargarEvento(Subsistema) : EventoSubsistema	Carga un evento dado de un subsistema.

Nombre: GestorMaquinaEstad	
Tipo de clase: Controladora.	
Atributo	**Tipo**
_MaquinaEstado :	MaquinaEstado
_GestorNotificacion	GestorNotificacion
_ConectorMaquinaEstado	IConectorMaquinaEstado
Responsabilidades	
Nombre	**Descripción**
CrearTransicion(Estado ini, Estado fin, IdentificadorOperacion) : void	Crea una definición de operación nueva.
EliminarTransicion(definicionOperacion) : void	Elimina una definición de operación dada.
CargarMaquinaEstado(Subsisteam) : void	Carga la maquina de estado de una subsistema dado.
ExisteTrasicion(DefinicionOperacion) : Bool	Si existe la definición de Operación dada.
ValidarOperacion(Operacion) : bool	Si la operación puede ser realizada,
RaelizarOperacion(Entidad, Operacion) : Bool	Realiza la operación dada sobre la entidad dada.
CambiarEstado(SubsistemaEstadoEntidad, Estado) : bool	Cambia el estado de una entidad que se encuentra en un subsistema.
CrearNotificaciones(Subsistema) : void	Crea notificaciones de cambio de estado al EMS.

Nombre: GestorSubsistemaEstadoEntidad	
Tipo de clase: Controladora.	
Atributo	**Tipo**
_EntidadesEstado :	SubsistemaEstadoEntidad
_ConectorSubsistemEstadoEntidad :	IConectorEstadoEntidad
Responsabilidades	
Nombre	**Descripción**
ActualizarEstadoEntidad(Subsistema, Estado, Entidad) : Entidad	Cambia el estado de una entidad en un subsistema dado.
CargarEntidades(Subsistema) : void	Carga las entidades de un subsistema dado.
AdicionarEntidades(Subsistema, SubsistemaEstadoEntidad) : void	Adiciona una entidad a un subsistema.
EliminarEntidades(Subsistema, SubsistemaEstadoEntidad) : voi	Elimina una entidad de un subsistema dado.

Nombre: IConectorMaquinaEstado	
Tipo de clase: Controladora.	
Nombre	**Descripción**
CargarTrancicion(string id)	Carga un definición de operación dado su id.
EliminarTrancicion(string id)	Elimina una definición de operación dada.
CargarMaquinaEstado(string id)	Carga una máquina de estado de un subsistema dado.

Nombre: IConectorSubsistemaEstadoEntidad	
Tipo de clase: Controladora.	
Atributo	**Tipo**
Nombre	**Descripción**
ActualizarEstadoEntidad(SubsistemaEstadoEntidad)	Cambia una entidad en un subsistema.
CargarEntidades(string)	Carga las Entidades de Un subsistema dado.
AdicionarEntidad(Entidad,Subsistema)	Adiciona a un subsistema una Entidad.
EliminarEntidad(Entidad , Subsistema)	Elimina una Entidad se un Subsistema

Anexo 13: Descripción de las clases entidades del Subsistema EMS.

Clases Entidades.

Nombre: Subsistema	
Tipo de clase : Entidad	
Atributo	**Tipo**
_MaquinaEstado	MaquinaEstado
_Eventos	ColeccionEventosSubsistema
_EsquemaInformacion	EsquemaInformacion
_Estados	ColeccionEstados
_EstadosEntidadSubsistemas	ColeccionSubsistemaEstadoEntidad
_Notificaciones	ColeccionNotificacion

Nombre: MaquinaEstado	
Tipo de clase : Entidad	
Atributo	**Tipo**
_DefinicionesOperacion	ColeccionDefinicionOperacion

Nombre: EventoSubsistema	
Tipo de clase : Entidad	
Atributo	**Tipo**
_IdentificadoresOperacion	ColeccionIdentificadorOperacion

Nombre: Estado	
Tipo de clase : Entidad	
Atributo	**Tipo**
_IdentificadroEstado	string
_DescripcionEstado	string

Nombre: SubsistemaEstadoEntidad	
Tipo de clase : Entidad	
Atributo	**Tipo**
_EntidadSubsistema	Entidad
_EstadoEntidad	Estado

Nombre: IdentificadorOperacion	
Tipo de clase : Entidad	
Atributo	**Tipo**
_IdentificadorOperacio	string
_EsquemaInformacionOP	EsquemaInformacion

Nombre: DefinicionOperacion	
Tipo de clase : Entidad	
Atributo	**Tipo**
_EstadoInicial	Estado
_EstadoFinal	Estado
_IdentificadoresOperacion	string

Nombre: Operación	
Tipo de clase : Entidad	
Atributo	**Tipo**
_DefenicionOperacion	DefinicionOperacion
_Entidadoperacion	Entidad

Nombre: Notificacion	
Tipo de clase : Entidad	
Atributo	**Tipo**
_EventoNotificacion	EventoSubsistema
_Entidad	Entidad

Nombre: Entidad	
Tipo de clase : Entidad	
Atributo	**Tipo**
_IdentificadorEntidad	string
_EsquemaInformacion	EsquemaInformacion

Nombre: EsquemaInformacion	
Tipo de clase : Entidad	
Atributo	**Tipo**
_XSDEsquema	string
_XMLInformacion	string

Nombre: ProyeccionInformacion	
Tipo de clase : Entidad	
Atributo	**Tipo**
_EsquemaInformacion	EsquemaInformacion
_Columnas	ColeccionColumnas
_Indices	ColeccionIndices

Anexo 14: Descripción de las clases para la Interfaz de Usuario del Subsistema de Administración de Servicios y Aplicaciones.

Nombre: frmGestionarTarjetas	
Tipo de clase :Interfaz	
Responsabilidades.	
Nombre:	InicializeComponent()
Descripción:	Inicializa el formulario.
Nombre:	Dispose() : void
Descripción:	Libera los recursos del Formulario.
Nombre:	Show() : void
Descripción:	Muestra el formulario.
Nombre:	FrmGestionarTarjetas()
Descripción:	Constructor del Formulario.

Nombre: frmGestionarServiciosEmp	
Tipo de clase: Interfaz	
Responsabilidades.	
Nombre:	InicializeComponent()
Descripción:	Inicializa el formulario.
Nombre:	Dispose() : void
Descripción:	Libera los recursos del Formulario.
Nombre:	Show() : void
Descripción:	Muestra el formulario.
Nombre:	FrmGestionarTarjetas()
Descripción:	Constructor del Formulario.

Nombre: frmGestionarAplicaciones	
Tipo de clase: Interfaz	
Responsabilidades	
Nombre:	InicializeComponent()
Descripción:	Inicializa el formulario.
Nombre:	Dispose() : void
Descripción:	Libera los recursos del Formulario.
Nombre:	Show() : void
Descripción:	Muestra el formulario.
Nombre:	FrmGestionarTarjetas()
Descripción:	Constructor del Formulario.

Nombre: accGestionarTarjetas	
Tipo de clase: Controladora	
Atributo	**Tipo**
_ accGestionarServiciosEmp	accGestionarServiciosEmp
_Tarjetas	ColeecionTarjetas
_frmGestionarTarjetas	frmGestionarTarjetas
_accGestionarAplicaciones	accGestionarAplicaciones
Responsabilidades	
Nombre:	CrearForma() : void
Descripción:	Crea el objeto de la forma.
Nombre:	MostrarFormulario() : void

Descripción:	Muestra el formulario.
Nombre:	MostrarEstado(Tarjeta) : void
Descripción:	El estado de una tarjeta dada.
Nombre:	BloquearTarjeta(Tarjeta) : bool
Descripción:	Cambia a bloqueado el estado de una tarjeta.
Nombre:	DesbloquearTarjeta(Tarjeta) : void
Descripción:	Desbloquea una tarjeta.
Nombre:	MostrarServicios(Tarjeta) : void
Descripción:	Muestra los servicios a los que tiene acceso una tarjeta.
Nombre:	MostrarAplicaciones(Tarjeta) : void
Descripción:	Muestra las Aplicaciones instaladas en una tarjeta.

Nombre: accGestionarServiciosEmp	
Tipo de clase: Controladora	
Atributo	**Tipo**
_frmGestionarServiciosEmp	frmGestionarServiciosEmp
_ServiciosEmp	ColeccionServicios
Responsabilidades	
Nombre:	CrearForma() : void
Descripción:	Crea el objeto de la forma.
Nombre:	MostrarFormulario() : void
Descripción:	Muestra el formulario.
Nombre:	MostrarServiciosEMp(Empresa) : void
Descripción:	Muestra los servicios que brinda una empresa.
Nombre:	EliminarServiciosEMp(Empresa) : bool
Descripción:	Le elimina un servicio a una Empresa.
Nombre:	DesbloquearTarjeta(EventoSubsistema) : void
Descripción:	Desbloquea una tarjeta.
Nombre:	AdicionarServiciosEMp(Empresa) : bool
Descripción:	Adiciona Servicios a la Empresa.
Nombre:	MostrarServiciosEmpTarjeta(Tarjeta, Empresa) : void
Descripción:	Muestra los servicio de una Empresa que tiene una tarjeta.

Nombre: accGestionarAplicaciones	
Tipo de clase: Controladora	
Atributo	**Tipo**
_frmGestionarAplicaciones	frmGestionarAplicaciones
Responsabilidades	
Nombre:	CrearForma() : void
Descripción:	Crea el objeto de la forma.
Nombre:	MostrarFormulario() : void
Descripción:	Muestra el formulario.
Nombre:	MostrarAplicaciones(Tarjeta) : void
Descripción:	Muestra las aplicaciones contenidas en una tarjeta.
Nombre:	EliminarAplicaciones(Tarjeta) : bool
Descripción:	Le elimina una aplicación a una Tarjeta.
Nombre:	DesbloquearAplicaciones(Tarjeta) : void
Descripción:	Desbloquea una aplicación en una tarjeta.
Nombre:	AdicionarAplicaciones(Tarjeta) : bool
Descripción:	Adiciona una(s) Aplicación en un tarjeta
Nombre:	BloquearAplicacion(Tarjeta, Aplicacion) : void

Anexo 15: Descripción de las clases controladoras del Subsistema de Administración de Servicios y Aplicaciones.

Clases Gestoras y Conectoras.

Nombre: IConectorCertificado	
Tipo de clase: Controladora	
Responsabilidades	
Nombre:	GuardarCertificado(CertificadoDigital, Tarjeta) : CertificadoDigital
Descripción:	Guarda en la BD un certificado Digital
Nombre:	CargarCertificado(Servicio) : Servicio
Descripción:	Carga de la BD el certificado Asociado a un servicio dado
Nombre:	EliminarCertificado(CertificadoDigital) : CertificadoDigital
Descripción:	Elimina un Certificado digital dado

Nombre: IConectorEmpresa	
Tipo de clase: Controladora	
Responsabilidades	
Nombre:	CargarEmpresas(ColeccionEmpresa) : ColeccionEmpresa
Descripción:	Carga las empresas que brindan servicios
Nombre:	EliminarEmpresa(Empresa) : bool
Descripción:	Elimina una empresa dada
Nombre:	AdicionarEmpresa(Empresa) : Empresa
Descripción:	Adiciona a la BD una empresa

Nombre: GestorEmpresa	
Tipo de clase: Controladora.	
Atributo	**Tipo**
_Empresas	coleccion
_gestordeCervicios	GestorServicios
_Conectorempresa	IConectorEmpresa
Responsabilidades	
Nombre:	CargarEmpresas(ColeccionEmpresa) : ColeccionEmpresa
Descripción:	Carga las empresas que brindan servicios
Nombre:	EliminarEmpresa(Empresa) : bool
Descripción:	Elimina una empresa dada
Nombre:	CrearEmpresa(Empresa) : Empresa
Descripción:	Adiciona a la BD una empresa
Nombre:	ModificarEmpresa(Empresa) : Empresa
Descripción:	Actualiza una empresa en la BD

Nombre: GestorCertificado	
Tipo de clase: Controladora.	
Atributo	**Tipo**
_Certificado	CertificadoDigital
_EmsInterfaz	EMSInterfaz
_GestorTarjetas	GestorTarjetas
Responsabilidades	
Nombre:	SolicitarCertificado(Empresa, Servicio) : CertificadoDIGITAL
Descripción:	Solicita a la AC un certificado digital para una empresa que brindara un Servicio dado.
Nombre:	EliminarServicioTarjeta(Servicio, Tarjeta : bool)
Descripción:	Elimina El servicio dado a la tarjeta dada.

Nombre:	CrearNotificacionOpCertificado(DefinicioOperacion) : string XMLInfor
Descripción:	Crea una notificación de operación sobre una entidad.
Nombre:	NotificacarOperacionXml(XMLInformacion) : void
Descripción:	Notifica al ems la operación creada en CrearNotificacionOpCertificado(DefinicioOperacion).
Nombre:	ConsumirNotificacionCertificado(string) : XMLInformacionNotificacion
Descripción:	Consume las notificaciones para el subsistema del EMS
Nombre:	ConvertirXmlNotificacionOpSistema(XmlInformacionNot : string) :Operacion
Descripción:	Coonvierte un Xml con el esquema de la Operacion a un objeto Operación.

Nombre: IConectorServicio	
Tipo de clase (Controladora)	
Responsabilidades	
Nombre:	CargarServiciosEmpresa(Empresa) : Empresa
Descripción:	Carga de la BD todos los Servicios de una Empresa dada
Nombre:	EliminarServicioEmpresa(Empresa) : bool
Descripción:	Le elimina un servicio a una empresa dada
Nombre:	AdicionarServicioEmpresa(Empresa) : Empresa
Descripción:	Le adiciona un Servicio a una Empresa dada

Nombre: GestorServicioEmpresa	
Tipo de clase (Controladora)	
Atributo	**Tipo**
_Servicios	Coleccion
_ConectorServicios	IConectorServicio
Responsabilidades	
Nombre:	SolicitarCertificado(Empresa, Servicio) : CertificadoDIGITAL
Descripción:	Solicita a la AC un certificado digital para una empresa que brindara un Servicio dado.
Nombre:	CargarServiciosEmpresa(Empresa) : Empresa
Descripción:	Carga de la Bd todos los Servicios de una Empresa
Nombre:	EliminarServicioEmpresa(Empresa) : Empresa
Descripción:	Elimina un servicio a una Empresa
Nombre:	AdicionarServicioEmpresa(Empresa) : Empresa
Descripción:	Adiciona un Servicio a una Empresa
Nombre:	CargaCertificadoServicio(Servicio) : CertificadoDigital
Descripción:	Carga de la BD el Certificado de un Servicio.
Nombre:	EliminarCertificadoServicio(Servicio, Tarjeta) : bool
Descripción:	Elimina un Servicio de una Tarjeta.

Nombre: GestorCedula	
Tipo de clase (Controladora)	
Atributo	**Tipo**
_Tarjeta	CedulaElectronica
_EMSInterfaz	InterfazEMS
Responsabilidades	
Nombre:	CargarTarjeta(string) : void
Descripción:	Carga una tarjeta de la BD dado su id
Nombre:	AdicionarServicioTarjeta(Tarjeta, ServicioEmpresa) : CedulaElectronica
Descripción:	Se le adiciona un servicio a una tarjeta.
Nombre:	EliminarServicioTarjeta(Tarjeta, ServicioEmpresa) : CedulaElectronica
Descripción:	Elimina un servicio a una Trajeta
Nombre:	CargarServiciosTarjeta(Tarjeta) : CedulaElectrnica

Descripción:	Todos los servicios de una tarjeta.
Nombre:	CrearNotificacionOpTarjeta(DefinicioOperacion) : string XMLInformacion
Descripción:	Notifica al EMS de una operación realizada en una tarjeta
Nombre:	ConsumirNotificacionTarjeta(DefinicioOperacion) : DefinicioOperacion
Descripción:	Carga las notificaciones existentes para este subsistema

Nombre: GestorAplicacion	
Tipo de clase (Controladora)	
Atributo	**Tipo**
_Aplicacion	Aplicacion
_GestorTarjetas	GestorTarjeta
Responsabilidades	
Nombre:	InstalarApplet(Aplicacion) : void
Descripción:	Instala un nuevo Applet en la tarjeta.
Nombre:	EliminarApplet(Aplicacion) : bool
Descripción:	Elimina un Applet de la tarjeta.
Nombre:	CambiarEstadoApplet(Aplicacionr) : Aplicacion
Descripción:	Cambia el estado de un applet dado en la tarjeta.

Nombre: IConectorTarjeta	
Tipo de clase (Controladora)	
Responsabilidades	
Nombre:	CargarTarjeta(string) : CedulaElectrnica
Descripción:	Carga una tarjeta dado su identificador.
Nombre:	AdicionarServicioTarjeta(Tarjeta, ServicioEmpresa) : CedulaElectrnica
Descripción:	Se le adiciona un servicio dado a una tarjeta.
Nombre:	EliminarServicioTarjeta(Tarjeta, ServicioEmpresa) : CedulaElectrnica
Descripción:	Se Elimina un servico dado de una tarjeta.
Nombre:	CargarServiciosTarjeta(Tarjeta) : CedulaElectrnica
Descripción:	Carga los servicios de una tarjeta.

Nombre: IConectorTarjeta	
Tipo de clase (Controladora)	
Responsabilidades	
Nombre:	CargarTarjeta(string) : CedulaElectrnica
Descripción:	Carga una tarjeta dado su identificador.
Nombre:	AdicionarServicioTarjeta(Tarjeta, ServicioEmpresa) : CedulaElectrnica
Descripción:	Se le adiciona un servicio dado a una tarjeta.
Nombre:	EliminarServicioTarjeta(Tarjeta, ServicioEmpresa) : CedulaElectrnica
Descripción:	Se Elimina un servicio dado de una tarjeta.
Nombre:	CargarServiciosTarjeta(Tarjeta) : CedulaElectrnica
Descripción:	Carga los servicios de una tarjeta.

Anexo 16: Descripción de las clases entidades del Subsistema de Administración de Servicios y Aplicaciones.

Nombre: IdentificadorOperacion	
Tipo de clase: Entidad	
Atributo	**Tipo**
_IdentificadorOperacion	string
_EsquemaInformacion	EsquemaInformacion

Nombre: Entidad	
Tipo de clase: Entidad	
Atributo	**Tipo**
_IdentificadorEntidad	string
_EsquemaInformacion	EsquemaInformacion

Nombre: Operacion	
Tipo de clase: Entidad	
Atributo	**Tipo**
_DefinicionOperacion	DefinicionOperacion
_EntidadOperacion	Entidad

Nombre: DefinicionOperacion	
Tipo de clase: Entidad	
Atributo	**Tipo**
_EstadoInicial	Estado
_EstadoFinal	Estado
_IdentificadorOperacion	IdentificadorOperacoion

Nombre: DefiniconEntidad	
Tipo de clase: Entidad	
Atributo	**Tipo**
EsquemaInformacionEntidad	EsquemaInformacion

Nombre: CertificadoDigital	
Tipo de clase: Entidad	
Atributo	**Tipo**
_XMLCertificadoInformacion	XMLType
_Estado	Estado

Nombre: Estado	
Tipo de clase: Entidad	
Atributo	**Tipo**
_IdentificadorEstado	string
_DescripcionEstado	string

Nombre: DatosAC	
Tipo de clase: Entidad	
Atributo	**Tipo**
_CertificadoFirma	CertificadoDigital
_CertificadoInformacion	CertificadoDigital

Nombre: Empresa	
Tipo de clase: Entidad	
Atributo	**Tipo**
_CertificadoFirma	CertificadoDigital
_CertificadoInformacion	CertificadoDigital

Nombre: CedulaElectronica	
Tipo de clase: Entidad	
Atributo	**Tipo**
_Servicios	Colleccion
_DatosAC	DatosAC
_Aplicaciones	colleccion
_NumeroSerie	string
_CodigoBarra	Image
_FechaExpedicion	datetime
_DatosPersonales	DatosPersona
_FechaExpiracion	datetime
_CapacidadTotal	int
_capacidadRealdelChip	int
_IdentificadorTarjeta	Byte[]
_Estado	Estado
Responsabilidades:	
Nombre:	RegionesAlmacenamineto(XmlInformacion) : int[]
Descripción:	Se le pasa el esquema de información de la tarjeta y te devuelve las secciones de almacenamiento libres en la tarjeta
Nombre:	EspacioLibrePorRegione() : int[]
Descripción:	Devuelve el espacio libre en las regiones

Nombre: ServicioEmpresa	
Tipo de clase (entidad)	
Atributo	**Tipo**
CertificadoAcceso	CErtificadoDigital
_nombreServicio	string

Nombre: Aplicacion	
Tipo de clase (entidad)	
Atributo	**Tipo**
_NombreAplicacion	string
_descripcion	string
_Ubicacion	string

Nombre: DatosPersona	
Tipo de clase (entidad)	
Atributo	**Tipo**
_PrimerNombre	string
_SegundoNombre	string
_PrimerApellido	string
_SegundoApellido	string
_FechaNacimiento	Image
_NumeroId	Byte[]
_Sexo	Char
_TipoCiudadadano	char
_EstadoCivil	char
_Foto	Image
_NumeroIDFiscal	Byte[]
_Firma	Image

Anexo 17: Descripción de las tablas del modelo entidad – relación del EMS.

Nombre: dRecursoXML		
Descripción: Contiene los datos de definición de la Información.		
Atributo	Tipo	Descripción
Descripción	NVARCHAR(30)	Descripción del Esquema Información.
XMLEsquemaInformacion	XMLType	Esquema de Información.

Nombre: dEntidad		
Descripción: Contiene las entidades que son manejadas en el EMS		
Atributo	Tipo	Descripción
XMLInformacion	XMLType	Información asociada a la entidad.
Nombre de la entidad	NVARCHAR(30)	El nombre de la entidad,

Nombre: dSubsistema		
Descripción: Contiene la información de los subsistemas del CAMS.		
Atributo	Tipo	Descripción
XMLInformacion	XMLType	Información asociada al subsistema
Nombre	NVARCHAR(30)	El nombre del subsistema,

Nombre: dEventos		
Descripción: Contiene los eventos de los subsistemas en el CAMS.		
Atributo	Tipo	Descripción
NombreEvento	NVARCHAR(30)	El nombre del evento.

Nombre: dNotificacion		
Descripción: Contiene la información de las notificaciones del CAMS		
Atributo	Tipo	Descripción
XMLInformacion	XMLType	La información asociada a la notificación.

Nombre: dSuscribcionEvento
Descripción: Contiene los eventos a los que están subscritos los Subsistemas.

Nombre: dEventoOperacion
Descripción: Contiene los eventos que desencadenan las operaciones.

Nombre: dDefinicionOperacion
Descripción: el estado final y el inicial de un subsistema al realizar operaciones.

Nombre: nEstadoEntidad		
Descripción: Los Estados por los que transitan las entidades.		
Atributo	Tipo	Descripción
Descripcion	NVARCHAR(30)	Descripción de los estados.

Nombre: dOperacion		
Descripción: Contiene las operaciones que se pueden realizar sobre las entidades.		
Atributo	Tipo	Descripción
NombreOperacion	NVARCHAR(30	El nombre de la Operación.
XMLEsquemaInformacionOp	XMLType	Contiene el formato de la operación.

Nombre: dIdentificadorDefinicionOp
Descripción: Contiene las operaciones con sus definiciones.

Nombre: dOperacionEntidad		
Descripción: Contiene las operaciones que se realizaron sobre las entidades.		
Atributo	Tipo	Descripción
FechaOp	DateTime	La fecha en que se realizo la Operación.

Anexo 18: Descripción de las tablas del modelo entidad – relación del Subsistema de Administración de Servicios y Aplicaciones.

Nombre: nEstado		
Descripción: Los Estados por los que transitan las entidades.		
Atributo	Tipo	Descripción
Descripción	NVARCHAR(30)	Descripción de los estados.

Nombre: dEmpresa		
Descripción: Contiene información de las Empresas.		
Atributo	Tipo	Descripción
Descripción	NVARCHAR(30)	Breve descripción de las Empresas.

Nombre: dServicio		
Descripción: Contiene información de los Servicios.		
Atributo	Tipo	Descripción
Descripción	NVARCHAR(30)	Breve descripción de los Servicios.

Nombre: dEmpresaServicio
Descripción: Contiene información de los Servicios que brindan las Empresas.

Nombre: dCertificado		
Descripción: Contiene información de los Certificados.		
Atributo	Tipo	Descripción
XMLEsquema	XMLType	XML con el esquema de los certificados.

Nombre: dCedulaElectronica		
Descripción: Contiene información de los Certificados.		
Atributo	Tipo	Descripción
XMLEsquema	XMLType	XML con el esquema de las Cédulas.

Nombre: dServicioCedula
Descripción: Contiene información de los Servicios que a los que tiene acceso una tarjeta.

Nombre: dOperacion		
Descripción: Contiene información de las Operaciones.		
Atributo	Tipo	Descripción
NombreOperacion	NVARCHAR(30)	Nombre de la operación.
XMLEsquemaInformacionOp	XMLType	XML con el esquema de la operación.

Nombre: dOperacionCedula		
Descripción: Contiene información de las Operaciones que se realizan sobre la cédula.		
Atributo	Tipo	Descripción
FechaOperacion	DateTime	Fecha en que se realizo la operación.

Nombre: dAplicacion		
Descripción: Contiene información de las Aplicaciones.		
Atributo	Tipo	Descripción
Descripción	NVARCHAR(30)	Breve descripción de las Aplicaciones.

Nombre: dCedulaAplicacion
Descripción: Contiene información de las Aplicaciones en las Cedulas.

Anexo 19: Descripción de extendida de los Casos de Uso del Sistema.

Descripción del CU Gestionar Subsistema.

Caso de uso:	Gestionar Subsistemas	
Actores:	Administrador del CAMS.	
Propósito:	Insertar un nuevo subsistema, modificarlo o eliminarlo.	
Resumen:	El caso de uso inicia cuando el administrador selecciona la opción para la gestión de subsistemas.	
Referencias:	RFEMS 1, RFEMS 1.1, RFEMS 1.4,	
Precondiciones:	El Administrador del CAMS debe estar autenticado en el sistema.	
Flujo Normal de los Eventos		
Acción del Actor		**Respuesta del Sistema**
1.- El administrador solicita gestionar un subsistema.		2.- El sistema muestra la interfaz para la gestión con el listado de los subsistemas que forman el CAMS.
		3.- El sistema ejecuta alguna de las siguientes acciones: a) Si decide insertar un nuevo subsistema, ir a la Sección 1. b) Si decide modificar un subsistema ir a la Sección 2. c) Si decide eliminar un subsistema ir a la Sección 3.
		4.- El sistema se actualiza.
Sección 1: Nuevo Subsistema.		
Acción del Actor		**Respuesta del Sistema**
3.1.- El administrador introduce el nombre del subsistema.		3.2.- El sistema activa el control para inserción de estados y operaciones.
3.3.- El administrador introduce las operaciones y los estados por los que atravesará la entidad.		3.4.- El sistema muestra mensaje de aceptación del nuevo subsistema.
Sección 2: Modificar Subsistema.		
Acción del Actor		**Respuesta del Sistema**
3.5.- El administrador selecciona el subsistema y modifica alguno (s) de sus datos (nombre, estados, operaciones).		3.6.- El sistema muestra mensaje de aceptación de la operación realizada.
Sección 3: Eliminar Subsistema.		
Acción del Actor		**Respuesta del Sistema**
3.7.- El administrador selecciona el subsistema y lo elimina.		3.8.- El sistema muestra mensaje de comprobación.
		3.9.- El sistema muestra mensaje de aceptación de la operación realizada.
Flujos Alternos		
Acción del Actor		**Respuesta del Sistema**
Poscondiciones:	Sección1: Se inserta un nuevo subsistema. Sección2: Se modifica un subsistema. Sección3: Se elimina un subsistema.	
Puntos de extensión:	Punto de extensión 1: Caso de uso: Gestionar Máquinas de Estado. Punto de extensión 2: Caso de Uso: Gestionar Eventos.	

Descripción del CU Gestionar Máquina de Estado.

Caso de uso:	Gestionar Máquinas de Estado.
Actores:	Administrador del CAMS.
Propósito:	Crear la máquina de estado, modificarla o eliminarla para un subsistema determinado.
Resumen:	El caso de uso inicia cuando el administrador selecciona el subsistema al cual gestionar su máquina de estado y la opción de gestión a realizar.
Referencias:	RFEMS 1.2, RFEMS 1.3
Precondiciones:	El Administrador del CAMS debe estar autenticado en el sistema.

Flujo Normal de los Eventos	
Acción del Actor	**Respuesta del Sistema**
1.- El administrador solicita gestionar la máquina de estado de un subsistema seleccionado.	2.- El sistema muestra la interfaz para la gestión de máquinas de estado, donde se muestran los estados y operaciones del subsistema seleccionado.
	3.- El sistema ejecuta alguna de las siguientes acciones: a) Si decide crear la máquina de estado del subsistema, ir a la Sección 1. b) Si decide modificar la máquina de estado del subsistema ir a la Sección 2. c) Si decide eliminar la máquina de estado del subsistema ir a la Sección 3.
	4.- El sistema se actualiza.
Sección 1: Crear Máquina de Estado del Subsistema.	
Acción del Actor	**Respuesta del Sistema**
3.1.- El administrador selecciona los estados que formarán la máquina de estado y las operaciones que provocan transición de un estado a otro.	3.2.- El sistema activa el control para la representación de la máquina de estado y muestra las transiciones especificadas por el administrador.
	3.3.- El sistema muestra mensaje de aceptación de la máquina de estado.
Sección 2: Modificar Máquina de Estado del Subsistema.	
Acción del Actor	**Respuesta del Sistema**
3.4.- El administrador modifica alguno (s) de los datos de la máquina de estado del subsistema seleccionado (adiciona/ elimina operaciones entre estados, elimina/adiciona estados).	3.5.- El sistema muestra mensaje de aceptación de la operación realizada.
Sección 3: Eliminar Máquina de Estado del Subsistema.	
Acción del Actor	**Respuesta del Sistema**
3.6.- El administrador elimina la máquina de estado del subsistema.	3.7.- El sistema muestra mensaje de comprobación.
	3.8.- El sistema muestra mensaje de aceptación de la operación realizada.
Flujos Alternos	
Acción del Actor	**Respuesta del Sistema**

Poscondiciones:	Sección1: Se crea la máquina de estado el subsistema. Sección2: Se modifica la máquina de estado el subsistema. Sección3: Se elimina la máquina de estado el subsistema.

Descripción del CU Gestionar Eventos.

Caso de uso:	Gestionar Eventos.
Actores:	Administrador del CAMS.
Propósito:	Subscribir o eliminar eventos para un subsistema determinado.
Resumen:	El caso de uso inicia cuando el administrador selecciona el subsistema al cual gestionar sus eventos y la opción de gestión a realizar.
Referencias:	RFEMS 1.5, RFEMS 1.6, RFEMS 1.7
Precondiciones:	El Administrador del CAMS debe estar autenticado en el sistema.

Flujo Normal de los Eventos	
Acción del Actor	**Respuesta del Sistema**
1.- El administrador solicita gestionar los eventos de un subsistema seleccionado.	2.- El sistema muestra la interfaz para la gestión de eventos, donde se muestran los eventos a los cuales puede subscribirse el subsistema seleccionado y aquellos a los cuales ya está subscrito.
	3.- El sistema ejecuta alguna de las siguientes acciones: a) Si decide subscribir el subsistema a nuevos eventos (ya existentes), ir a la Sección 1. b) Si decide eliminar subscripciones de eventos al subsistema ir a la Sección 2. c) Si decide subscribir el subsistema a nuevos eventos (no existentes), ir a la Sección 3.
	4.- El sistema se actualiza.
Sección 1: Subscribir el Subsistema a Eventos Existentes.	
Acción del Actor	**Respuesta del Sistema**
3.1.- El administrador selecciona el evento al que desea subscribir el subsistema seleccionado.	3.2.- El sistema comprueba si el subsistema ya está subscrito al evento seleccionado.
	3.3.- El sistema muestra mensaje de aceptación del evento indicado.
Sección 2: Eliminar Subscripciones a eventos de un Subsistema.	
Acción del Actor	**Respuesta del Sistema**
3.4.- El administrador selecciona los eventos a los cuales no desea estar subscrito y los elimina.	3.5.- El sistema muestra mensaje de comprobación.
	3.6.- El sistema muestra mensaje de aceptación de la operación realizada.
Sección 3: Subscribir el Subsistema a Eventos no Existentes	
3.7.- El administrador selecciona la opción para crear un nuevo evento.	3.8.- El sistema activa el control para crear un nuevo evento.
3.9.- El administrador escribe el nombre del evento y selecciona el subsistema donde ocurrirán las operaciones/cambios de estado de las entidades que indiquen la ocurrencia del evento.	3.10.- Muestra las operaciones y los estados del subsistema seleccionado.
3.11.- Selecciona las operaciones los estados que desea controlar en el evento y queda creado el nuevo evento.	3.12.- El sistema muestra mensaje de aceptación de la operación realizada.
3.13.- Subscribe el Subsistema inicialmente seleccionado al evento que acaba de crear.	3.14.- El sistema muestra mensaje de aceptación de la operación realizada.
Flujos Alternos	
Acción del Actor	**Respuesta del Sistema**
Poscondiciones:	Sección1: Se subscribe un subsistema a eventos ya existente.

	Sección2: Se eliminan subscripciones a eventos de un subsistema. Sección3: Se subscribe un subsistema a eventos nuevos.

Descripción de CU Insertar Lote de Tarjetas.

Caso de uso:	Insertar Lote de Tarjetas.	
Actores:	Responsable Inventario.	
Propósito:	Insertar lote de tarjetas al sistema, estas se registran como “Inicializadas”.	
Resumen:	El caso de uso inicia cuando el administrador selecciona el subsistema al cual gestionar sus eventos y la opción de gestión a realizar.	
Referencias:	RFINV 1	
Precondiciones:	Responsable Inventario debe estar autenticado en el sistema.	
Flujo Normal de los Eventos		
Acción del Actor		**Respuesta del Sistema**
1.- El Responsable Inventario solicita la opción de insertar un lote nuevo de tarjetas.		2.- El sistema muestra la interfaz para la inserción de lotes, donde se muestran los que ya han sido registrados en el inventario y las ubicaciones disponibles en el almacén.
3.- El Responsable Inventario, escribe los datos del nuevo lote (Serie inicial y final, el número de caja, y el proveedor), selecciona la ubicación para el nuevo lote y lo inserta		4.- El sistema inserta el nuevo lote.
		5.- El sistema notifica la operación realizada al Subsistema de EMS.
		6.- El sistema muestra mensaje de aceptación.
Flujos Alternos		
Acción del Actor		**Respuesta del Sistema**
Poscondiciones:	Se inserta un lote de tarjetas al inventario y se notifica esta operación al Subsistema de EMS.	

Descripción de CU Reclamar Lote de Tarjetas.

Caso de uso:	Reclamar Lote de Tarjetas.	
Actores:	Responsable Inventario.	
Propósito:	Notificar al proveedor que el lote de tarjetas está en mal estado para que se reponga.	
Resumen:	El caso de uso inicia cuando el Responsable Inventario detecta un lote de tarjetas en mal estado y selecciona la opción de reclamar al proveedor dicho lote.	
Referencias:	RFINV 1	
Precondiciones:	Responsable Inventario debe estar autenticado en el sistema.	
Flujo Normal de los Eventos		
Acción del Actor		**Respuesta del Sistema**
1.- El Responsable Inventario solicita la opción de reclamar un lote de tarjetas.		2.- El sistema muestra la interfaz para la reclamación de lotes, donde se muestran los que ya han sido registrados en el inventario.
3.- El Responsable Inventario, selecciona el lote a reclamar y efectúa la reclamación.		4.- El sistema cambia el estado de las tarjetas que forman el lote a Anuladas.
		5.- El sistema notifica la operación realizada al Subsistema de EMS.
		6.- 5.- El sistema muestra mensaje de aceptación.
Flujos Alternos		

Acción del Actor	Respuesta del Sistema
Poscondiciones:	Se reclama un lote de tarjetas al proveedor, se cambia el estado de las tarjetas que forman el lote y se notifica esta operación al Subsistema de EMS.

Descripción de CU Enviar Lote de Tarjetas a Oficina.

Caso de uso:	Enviar Lote de Tarjetas a Oficina.
Actores:	Responsable Inventario.
Propósito:	Enviar los lotes de tarjetas que estén listas para enviar a oficina hacia las oficinas.
Resumen:	El caso de uso inicia cuando el Responsable Inventario decide enviar los lotes de las tarjetas que estén listas para enviar a oficina hacia las oficinas.
Referencias:	RFINV 1
Precondiciones:	Responsable Inventario debe estar autenticado en el sistema.
Flujo Normal de los Eventos	
Acción del Actor	**Respuesta del Sistema**
1.- El Responsable Inventario solicita la opción de enviar lotes de tarjetas a las oficinas.	2.- El sistema muestra la interfaz para el envío a oficinas, donde se muestran los lotes de tarjetas que están listos para enviar a oficina, las oficinas hacia donde los puede enviar y los posibles transportadores.
3.- El Responsable Inventario, selecciona una oficina de destino, un transportador y los lotes que decida enviar a esa oficina.	4.- El sistema cambia el estado de las tarjetas que forman el lote a Enviadas a Oficina.
	5.- El sistema notifica la operación realizada al Subsistema de EMS.
	6.- 5.- El sistema muestra mensaje de aceptación.
Flujos Alternos	
Acción del Actor	**Respuesta del Sistema**
Poscondiciones:	Se envían lotes de tarjetas a las oficinas, se cambia el estado de las tarjetas que forman el lote y se notifica esta operación al Subsistema de EMS.

Descripción del CU Cambiar Estado de Certificados Digitales.

Caso de uso:	Cambiar Estado de Certificados Digitales.
Actores:	Subsistema
Propósito:	Informar al Subsistema de AC un cambio de estado en los Certificados Digitales y registrar este cambio en el EMS.
Resumen:	El caso de uso inicia cuando ocurre algún evento en alguno de los subsistemas generalizados en el actor Subsistema que indica un cambio de estado de certificados digitales, generalmente estos eventos están asociados a cambios de estado de las tarjetas.
Referencias:	RFAC 2
Precondiciones:	Está establecida la conexión entre el Subsistema y el Subsistema de AC.
Flujo Normal de los Eventos	
Acción del Actor	**Respuesta del Sistema**
1.- El Subsistema informa que se ha realizados una operación sobre una tarjeta que implica un cambio de estado.	2.- El sistema procesa la operación y registra el cambio de estado de los Certificados Digitales de la tarjeta en el EMS.
	3.- El sistema notifica a la AC el cambio de estado de certificados digitales de una tarjeta.
Flujos Alternos	
Acción del Actor	**Respuesta del Sistema**
Poscondiciones:	Cambia el estado de los Certificados Digitales de una tarjeta.

Descripción del CU Crear Certificados Digitales y Llaves.

Caso de uso:	Crear Certificados Digitales y Llaves
Actores:	Subsistema de Personalización
Propósito:	Crear par de llaves y los Certificados Digitales de Autenticación y Firma derivados de estas.
Resumen:	El caso de uso inicia cuando el Subsistema de Personalización solicita al Subsistema de AC los Certificados Digitales y las llaves con las cuales generan estos para utilizarlos en el proceso de personalización.
Referencias:	RFAC 1
Precondiciones:	Está establecida la conexión entre el Subsistema de Personalización y el Subsistema de AC.
Flujo Normal de los Eventos	
Acción del Actor	**Respuesta del Sistema**
1.- El Subsistema de Personalización realiza una petición al Subsistema de AC de Certificados Digitales y Llaves para el proceso de personalización.	2.- El sistema solicita a la AC generar un par de llaves y los certificados digitales de firma y autenticación a partir de estas llaves.
	3.- El sistema crea paquetes en dependencia del número de peticiones que recibió, con el par de llaves y los certificados obtenidos, los encripta y los envía al Subsistema de Personalización.
Flujos Alternos	
Acción del Actor	**Respuesta del Sistema**
Poscondiciones:	Se obtienen los certificados digitales y las llaves con las cuales se generaron para utilizarlos en el proceso de personalización.

Descripción del CU Gestionar Órdenes de Personalización.

Caso de uso:	Gestionar Órdenes de Personalización.
Actores:	Personalizador de Documentos.
Propósito:	Crear o Personalizar Órdenes de Personalización.
Resumen:	El caso de uso inicia cuando el Personalizador de Documentos selecciona la opción para la gestión de órdenes de personalización.
Referencias:	RFPER 1, RFPER 1.2, RFPER 2, RFPER 2
Precondiciones:	El Personalizador de Documentos debe estar autenticado en el sistema.
Flujo Normal de los Eventos	
Acción del Actor	**Respuesta del Sistema**
1.- El Personalizador de Documentos solicita gestionar las órdenes de personalización.	2.- El sistema muestra la interfaz para la gestión de órdenes de personalización, donde se muestran el listado de personas disponibles, las tarjetas inicializadas y las órdenes existentes.
	3.- El sistema ejecuta alguna de las siguientes acciones: a) Si decide crear una orden de personalización, ir a la Sección 1. b) Si personalizar una orden de personalización, ir a la Sección 2.
Sección 1: Crear una orden de personalización.	
Acción del Actor	**Respuesta del Sistema**
3.1.- El Personalizador de Documentos selecciona en el listado de personas aquellas que van a conformar la orden, las tarjetas en la cuales se van a grabar sus	3.2.- El sistema crea la orden de personalización y la muestra en el listado de órdenes listas para personalizar.

datos y cada una de estas le asocia un número de identidad.	
	3.3.- El sistema muestra mensaje de aceptación.
Sección 2: Personalizar Orden de Personalización.	
Acción del Actor	**Respuesta del Sistema**
3.4.- El Personalizador de Documentos selecciona las órdenes y las manda a personalizar.	3.5.- El sistema solicita a la AC los Certificados Digitales de Firma y Autenticación y las llaves que se utilizaron para generarlos para cada una de las tarjetas que forman la orden.
	3.6.- El sistema graba los datos de las personas y los certificados en las tarjetas.
	3.7.- El sistema notifica el cambio de estado de las tarjetas al Subsistema de EMS.
	3.8.- El sistema muestra un mensaje de aceptación de la operación realizada.
Flujos Alternos	
Acción del Actor	**Respuesta del Sistema**
Poscondiciones:	Sección1: Se crean las órdenes de personalización. Sección2: Se graban en las tarjetas los datos de los ciudadanos y se cambia el estado de las tarjetas.

Descripción del CU Gestionar Lotes.

Caso de uso:	Gestionar Lotes
Actores:	Responsable de Inventario.
Propósito:	Recibir o rechazar lotes de tarjetas.
Resumen:	El caso de uso inicia cuando el Responsable de Inventario selecciona la opción para la gestión de lotes de tarjetas en la Oficina.
Referencias:	RFOF 1
Precondiciones:	Responsable de Inventario debe estar autenticado en el sistema.
Flujo Normal de los Eventos	
Acción del Actor	**Respuesta del Sistema**
1.- El Responsable de Inventario solicita gestionar los totes de tarjetas en la Oficina.	2.- El sistema muestra la interfaz para la gestión de los totes, donde se muestran las ubicaciones de su almacén y los lotes que se encuentran en estas.
	3.- El sistema ejecuta alguna de las siguientes acciones: a) Si decide recibir un lote nuevo en la Oficina, ir a la Sección 1. b) Si decide rechazar un lote, ir a la Sección 2.
Sección 1: Recibir Lote de Tarjetas.	
Acción del Actor	**Respuesta del Sistema**
3.1.- El Responsable de Inventario introduce los datos del lote que va a recibir en su almacén y lo inserta en una de sus ubicaciones.	3.2.- El sistema notifica al Subsistema de EMS un cambio de estado en las tarjetas que forman el lote recibido.
	3.3.- El sistema muestra mensaje de aceptación.
Sección 2: Rechazar Lote de Tarjetas.	
Acción del Actor	**Respuesta del Sistema**
3.4.- El Responsable de Inventario introduce los datos del lote que va a rechazar.	3.5.- El sistema notifica al Subsistema de EMS un cambio de estado en las tarjetas que forman el lote rechazado.
	3.6.- Imprime un acta de constancia con las causas del

	rechazo.
	3.7.- El sistema muestra mensaje de aceptación.
Flujos Alternos	
Acción del Actor	**Respuesta del Sistema**
Poscondiciones:	Sección1: Se recibe un nuevo lote en la Oficina y se cambia el estado de las tarjetas que lo forman. Sección2: Se rechaza un lote y se cambia el estado de las tarjetas que lo forman.

Descripción del CU Entregar Cédula.

Caso de uso:	Entregar Cédulas.
Actores:	Responsable de Entrega de Cedulación
Propósito:	Entregar las cédulas a los ciudadanos.
Resumen:	El caso de uso inicia cuando el ciudadano se presenta en la Oficina y solicita que se le entregue su cédula y se procese a realizar esta operación por parte del Responsable de Entrega de Cedulación.
Referencias:	RFOF 1
Precondiciones:	Responsable de Entrega de Cedulación debe estar autenticado en el sistema.
Flujo Normal de los Eventos	
Acción del Actor	**Respuesta del Sistema**
1.- Responsable de Entrega de Cedulación selecciona la opción de entregar cédulas.	2.- El sistema muestra el listado de los ciudadanos que pueden recibir sus cédulas.
3.- El Responsable de Entrega de Cedulación selecciona en el listado mostrado por el sistema la persona a la que se va a entregar la cédula.	4.- El sistema verifica la identidad de la persona. a) Si la comprobación arroja como resultado que la persona no es quien dice ser, ver Flujo Alterno 1.
5.- El Responsable de Entrega de Cedulación entrega la cédula al ciudadano.	6.- El sistema notifica un cambio de estado en la tarjeta al Subsistema de EMS.
	7.- El sistema muestra mensaje de aceptación.
Flujo Alterno1: Identidad no Válida.	
	5.- El sistema muestra un mensaje informando que la persona no es quien dice ser.
6.- El Responsable de Entrega de Cedulación informa a las autoridades y al ciudadano de esta irregularidad.	
Acción del Actor	**Respuesta del Sistema**
Poscondiciones:	Los ciudadanos obtienen sus cédulas de identificación.

Descripción del CU Gestionar Servicios.

Caso de uso:	Gestionar Servicios de tarjetas
Actores:	Administrador de Tarjetas y Servicios.
Propósito:	Asignar o eliminar servicios a una tarjeta.
Resumen:	El caso de uso inicia cuando el Administrador de Tarjetas y Servicios selecciona la opción para la gestión de servicios.
Referencias:	RFASA 1
Precondiciones:	Administrador de Tarjetas y Servicios debe estar autenticado en el sistema.
Flujo Normal de los Eventos	
Acción del Actor	**Respuesta del Sistema**
1.- El Administrador de Tarjetas y Servicios solicita gestionar los servicios.	2.- El sistema muestra la interfaz para la gestión de servicios, donde se muestran las empresas que brindan servicios y un listado de estos.
	3.- El sistema ejecuta alguna de las siguientes

	acciones: a) Si decide adiciona un servicio nuevo a la tarjeta, ir a la Sección 1. b) Si decide eliminar un servicio a la tarjeta, ir a la Sección 2.
Sección 1: Adicionar Servicio a la Tarjeta.	
Acción del Actor	**Respuesta del Sistema**
3.1.- El Administrador de Tarjetas y Servicios introduce el identificador de la tarjeta a la cual le va a asignar un servicio.	3.2.- El sistema muestra los servicios que tiene asignada la tarjeta.
3.3.- El Administrador de Tarjetas y Servicios selecciona la Empresa que brinda el servicio que se va a asignar a la tarjeta.	3.4.- El sistema muestra los servicios que brinda la Empresa seleccionada.
3.5.- El Administrador de Tarjetas y Servicios selecciona el servicio y lo adiciona a la tarjeta.	3.6.- El sistema solicita al Subsistema de AC un certificado digital con los permisos de lectura/escritura sobre una sección de almacenamiento libre dentro de la tarjeta.
	3.7.- El sistema notifica a la Empresa que ya posee los permisos necesarios sobre una región de almacenamiento de la tarjeta y envía a la empresa el Certificado que le permite brindar el servicio.
	3.8.- El sistema muestra mensaje de aceptación.
Sección 2: Eliminar Servicios a una Tarjeta.	
Acción del Actor	**Respuesta del Sistema**
3.9.- El Administrador de Tarjetas y Servicios introduce el identificador de la tarjeta a la cual le va a eliminar un servicio.	3.10.- El sistema muestra los servicios que tiene asignada la tarjeta.
3.11.- El Administrador de Tarjetas y Servicios selecciona el servicio que desea eliminar y lo elimina a la tarjeta.	3.12.- El sistema solicita al Subsistema de AC que revoque el certificado que tiene asignado el servicio que se quiere eliminar.
	3.13.-El sistema notifica a la Empresa y al ciudadano que se eliminó el servicio en la tarjeta.
	3.7.- El sistema muestra mensaje de aceptación.
Flujos Alternos	
Acción del Actor	**Respuesta del Sistema**
Poscondiciones:	Sección1: Se le adiciona un nuevo Servicio a la tarjeta. Sección2: Se elimina un servicio de la tarjeta.

Descripción del CU Gestionar Tarjetas.

Caso de uso:	Gestionar Tarjetas
Actores:	Administrador de Tarjetas y Servicios.
Propósito:	Bloquear o desbloquear Tarjetas.
Resumen:	El caso de uso inicia cuando el Administrador de Tarjetas y Servicios selecciona la opción para la gestión de tarjetas.
Referencias:	RFASA 1, RFASA 3, RFASA 4
Precondiciones:	Administrador de Tarjetas y Servicios debe estar autenticado en el sistema.
Flujo Normal de los Eventos	
Acción del Actor	**Respuesta del Sistema**
1.- El Administrador de Tarjetas y Servicios solicita la gestión de tarjetas.	2.- El sistema muestra la interfaz para la gestión de Tarjetas, donde se muestran las solicitudes de bloqueo y desbloqueo de tarjetas así como las causas

4.- El Administrador de Tarjetas y Servicios selecciona la solicitud a la cual le dará solución y le cambia el estado a la tarjeta según lo indicado por la solicitud.	3.- El sistema elimina la solicitud de la lista, cambia el estado de la tarjeta y notifica el cambio de estado de la tarjeta al Subsistema de EMS.
	4.- El sistema muestra mensaje de aceptación.
Flujos Alternos	
Acción del Actor	**Respuesta del Sistema**
Poscondiciones:	Se modifica el estado de la Tarjeta.

Descripción del CU Gestionar Notificación de Servicios.

Caso de uso:	Gestionar Notificación de Servicios
Actores:	Notificador de Servicios.
Propósito:	Notificar o eliminar de servicios que puede brindar una empresa.
Resumen:	El caso de uso inicia cuando el Notificador de Servicios selecciona la opción para la gestión de notificaciones.
Referencias:	RFASA 4, RFASA 5, RFASA 6
Precondiciones:	Notificador de Servicios debe estar autenticado en el sistema.
Flujo Normal de los Eventos	
Acción del Actor	**Respuesta del Sistema**
1.- El Notificador de Servicios solicita gestionar las notificaciones.	2.- El sistema muestra la interfaz para la gestión de notificaciones de servicios, donde se muestran las empresas que brindan servicios y un listado de estos.
	3.- El sistema ejecuta alguna de las siguientes acciones: a) Si decide crear una notificación de servicios nueva, ir a la Sección 1. b) Si decide eliminar notificación de servicios, ir a la Sección 2.
Sección 1: Crear Notificación de Servicios.	
Acción del Actor	**Respuesta del Sistema**
3.1.- El Notificador de Servicios introduce los datos de la Empresa que realiza la notificación de los servicios que puede brindar.	3.2.- El sistema registra la notificación..
	3.3.- El sistema muestra mensaje de aceptación.
Sección 2: Eliminar Notificación de Servicios.	
Acción del Actor	**Respuesta del Sistema**
3.5.- El Notificador de Servicios selecciona la empresa a la cual eliminar alguna notificación de servicios.	3.6.- El sistema muestra las notificaciones de servicios que tiene la empresa seleccionada.
3.7.- El Notificador de Servicios selecciona la notificación que desea eliminar y la elimina.	3.8.- El sistema muestra mensaje de aceptación.
Flujos Alternos	
Acción del Actor	**Respuesta del Sistema**
Poscondiciones:	Sección1: Se crea una Notificación de servicios que puede brindar una empresa. Sección2: Se elimina una Notificación de servicios que puede brindar una empresa.

Descripción del CU Gestionar Aplicaciones.

Caso de uso:	Gestionar Aplicaciones
Actores:	Administrador de Aplicaciones.
Propósito:	Adicionar, eliminar, bloquear y desbloquear applets de la tarjeta.
Resumen:	El caso de uso inicia cuando el Administrador de Aplicaciones selecciona la opción para la gestión de Aplicaciones.
Referencias:	RFASA 2, RFASA 4
Precondiciones:	El Administrador de Aplicaciones debe estar autenticado en el sistema. La tarjeta a la cual se le van a gestionar sus aplicaciones debe estar captada por un lector.

Flujo Normal de los Eventos	
Acción del Actor	**Respuesta del Sistema**
1.- El Administrador de Aplicaciones selecciona la opción para la gestión de las aplicaciones una tarjeta.	2.- El sistema muestra la interfaz para la gestión de aplicaciones y muestra los applets de la tarjeta que se encuentra captada por el lector.
	3.- El sistema ejecuta alguna de las siguientes acciones: a) Si decide adicionar una nueva aplicación, ir a la Sección 1. b) Si decide eliminar una aplicación, ir a la Sección 2. c) Si decide bloquear una aplicación, ir a la Sección 3. d) Si decide desbloquear una aplicación, ir a la Sección 4.
Sección 1: Adicionar Aplicación.	
Acción del Actor	**Respuesta del Sistema**
3.1.- El Administrador de Aplicaciones carga los applets que tiene almacenados.	3.2.- El sistema muestra los applets disponibles.
3.3.- El Administrador de Aplicaciones selecciona el Applet que desea adicionar a la tarjeta.	3.4.- El sistema verifica si la tarjeta tiene espacio para adicionar un nuevo Applet y si la tarjeta no contiene ya el Applet indicado. a) Si la comprobación arroja que ya existe o que la tarjeta no tiene espacio suficiente, ver Flujo Alterno1.
	3.5.- El sistema adiciona el nuevo Applet a la tarjeta.
	3.6.- El sistema muestra mensaje de aceptación.
Sección 2: Eliminar Aplicación.	
Acción del Actor	**Respuesta del Sistema**
	3.7.- El sistema muestra los applets que contiene la tarjeta.
3.8.- El Administrador de Aplicaciones selecciona el Applet deseado y lo elimina.	3.8.- El sistema muestra mensaje comprobación con la información de las consecuencias de eliminar este Applet.
	3.9.- El sistema elimina el Applet y si esta operación implica un cambio de estado en la tarjeta la notifica al Subsistema de EMS.
	3.10.- El sistema muestra mensaje de aceptación.
Sección 3: Bloquear Aplicación.	
Acción del Actor	**Respuesta del Sistema**
	3.9.- El sistema muestra los applets que contiene la tarjeta.

3.10.- El Administrador de Aplicaciones selecciona el Applet deseado y lo bloquea.	3.11.- El sistema muestra mensaje comprobación con la información de las consecuencias de bloquear este Applet.
	3.12.- El sistema bloquea el Applet y si esta operación implica un cambio de estado en la tarjeta la notifica al Subsistema de EMS.
	3.11.- El sistema muestra mensaje de aceptación.
Sección 4: Desbloquear Aplicación.	
Acción del Actor	**Respuesta del Sistema**
	3.12.- El sistema muestra los applets bloqueados que contiene la tarjeta.
3.13.- El Administrador de Aplicaciones selecciona el Applet deseado y lo desbloquea.	3.14.- El sistema muestra mensaje comprobación con la información de las consecuencias de desbloquear este Applet.
	3.12.- El sistema desbloquea el Applet y si esta operación implica un cambio de estado en la tarjeta la notifica al Subsistema de EMS.
	3.11.- El sistema muestra mensaje de aceptación.
Flujos Alternos	
Acción del Actor	**Respuesta del Sistema**
Poscondiciones:	Sección1: Se adiciona un Applet nuevo a la tarjeta. Sección2: Se elimina un Applet a la tarjeta. Sección3: Se bloquea un Applet de la tarjeta. Sección4: Se desbloquea un Applet de la tarjeta.

Anexo 20: Propuesta de Interfaz de Usuario.