

**Universidad de las Ciencias Informáticas**

**Facultad 7**



**Título: Registro de Ciudadanos 2.0  
del Sistema de Información para la Salud.**

**Trabajo de Diploma para optar por el título de**

**Ingeniero en Ciencias Informáticas**

**Autores:** Orisbel García Arcacha  
Reynaldo Morales Báez

**Tutores:** Lic. María del Carmen Padermi López  
Ing. Yosvani Turruelles Tejeda

**Asesor:** Lic. Juana I. Pérez Rodríguez

La Habana, Junio del 2008

“Año 50 de la Revolución”

## DECLARACIÓN DE AUTORÍA

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

Para que así conste firmamos la presente a los 18 días del mes de Junio del año 2008.

---

Orisbel García Arcacha  
Firma del Autor

---

Lic. María del Carmen Padermi López  
Firma del Tutor

---

Reynaldo Morales Báez  
Firma del Autor

---

Ing. Yosvani Turruelles Tejeda  
Firma del Tutor

## DATOS DE CONTACTO

### Tutores:

**Lic. María del Carmen Padermi López:** Especialista A en Sistemas Organizativos e Informativos de la empresa SOFTEL. Graduada en Cibernética Matemática en el año 1979 en la Universidad de La Habana. Posee 27 años de experiencia en el desarrollo de software desempeñando diferentes roles. Ha trabajado en las empresas EDISIPRODA perteneciente al MITRANS (1979 – 1984) y SOFTEL (1984 hasta la actualidad). Ha recibido varios cursos de superación y postgrado. Ha sido tutora y oponente de tesis en varios cursos.

Correo electrónico: [carmenchu@softel.cu](mailto:carmenchu@softel.cu)

**Ing. Yosvani Turruelles Tejeda:** Graduado como Ingeniero en Ciencias Informáticas en el año 2007 en la Universidad de Ciencias Informáticas (UCI). Actualmente se desempeña como profesor de Gestión de Software en la Facultad 7. Posee la categoría docente de Instructor Recién Graduado. Forma parte del proyecto Atención Primaria de Salud de la Facultad 7.

Correo electrónico: [yturruelles@uci.cu](mailto:yturruelles@uci.cu)

### Asesor:

**Lic. Juana Isabel Pérez Rodríguez:** Graduado de Licenciado en Educación en la Especialidad de Lengua Inglesa en el año 2000 en el ISPEJV en la Facultad de Lenguas Extranjeras. Posee categoría docente de Profesor Instructor Principal y cursa en estos momentos una maestría en la Universidad de La Habana. Ha impartido las asignaturas Idioma Extranjero I y II en la Facultad 7 desde el curso 2004-2005. Ha presentado ponencias en eventos científicos nacionales e internacionales. Se desempeña como Jefa de la Asignatura en la propia facultad.

Correo electrónico: [jane@uci.cu](mailto:jane@uci.cu)

*“El futuro de Cuba tiene que ser necesariamente  
un futuro de hombres de ciencia,  
de hombres de pensamiento.”*

***Fidel Castro Ruz***

## AGRADECIMIENTOS

*A la Revolución, por darnos la oportunidad de estudiar en esta Universidad...*

*A nuestras familias, por siempre confiar en nosotros y apoyarnos en todo...*

*A todos los especialistas de la empresa Softel: Mirna, Carmenchu, Cary,*

*Denis, Liannys...*

*A los profes: Yosvani, Johander, Pura...*

*A todos los integrantes del Proyecto APS, en especial a nuestros compañeros de 5to año....*

*A todos los que han sido participes de la realización de este trabajo de diploma  
muchas gracias...*

## DEDICATORIA

*Dedico este trabajo especialmente a mi niño Ian Manuel, por tantos días de ausencia, alejado de su mamá...pero, ya se acabó!!!*

*A mis padres, Lourdes y Orestes por siempre confiar en mí y por ambos entregarse completamente a cuidar y educar a mi más valioso tesoro: Ian. Gracias a ellos hoy logré ser lo que soy...*

*A mi hermano Orelbis por ser un tío abnegado...*

*A mi abuela Norma, por ser siempre tan complaciente...*

*A mis abuelos Nilda y Oreste por su amor y cariño...*

*A mi Piquetón: Yenisbel, Lisandra, Yaquelín, Ariuska y Yaira por ser incondicionales.*

*Al proyecto APS...*

*Y a todos, les dedico este trabajo...*

***Orisbel García Arcacha***

*Dedico este trabajo a mi mamá por estar siempre ahí cuando la he necesitado...*

*A mi papá por apoyarme y velar por mis estudios...*

*A mi hermana Leydi y a mi novia Lisbeth, por soportarme todas las malcriadeces...*

*A mis tíos, mi abuelo, mi cuñado y primo...*

*A todas aquellas personas que han estado cerca de mí todo el tiempo y se han preocupado por todas mis cosas, que en fin de cuenta son mi familia...*

*Al proyecto APS en especial al Newpa, Yonha, Yuma, Fide, Lisander y demás...*

*Y a todos, les dedico este trabajo...*

***Reynaldo Morales Baez***

## RESUMEN

En el desarrollo del Sistema de Información para la Salud, surge la necesidad de contar con información no perteneciente este, por lo que se requiere de un componente que manipule los datos del carné de identidad de los ciudadanos del país y se decide desarrollar el Registro de Ciudadanos.

El presente trabajo tiene como objetivo mejorar e incorporar nuevas funcionalidades al Registro de Ciudadanos existente en el Sistema de Información para la Salud, permitiendo obtener una mayor centralización de la información para ser utilizada por el resto de los componentes de SISalud.

Para cumplir este objetivo se utilizan tecnologías actuales como PHP, XSL, XML, MySQL y Apache. Además de algunas herramientas como Zend Development Environment, MySQL-Front, Rational Rose, Dreamweaver y Style Studio. En el modelado de los flujos de trabajo se empleó la metodología de desarrollo que brinda el Proceso de Desarrollo Unificado (RUP) y el lenguaje de modelado visual UML, siguiendo la arquitectura y tecnologías propuesta por el MINSAP.

El sistema propuesto permitirá al país contar con el sistema automatizado Registro de Ciudadanos 2.0, que brindará información homogénea de los ciudadanos cubanos y extranjeros al Sistema de Información para la Salud. Además, se espera, con el mantenimiento correctivo y evolutivo del Registro de Ciudadanos actual, un eficiente manejo de los datos que este gestiona, así como poder contar con información actualizada y precisa de los ciudadanos cubanos y extranjeros.

## TABLA DE CONTENIDOS

<b>AGRADECIMIENTOS.....</b>	<b>V</b>
<b>DEDICATORIA.....</b>	<b>VI</b>
<b>RESUMEN.....</b>	<b>VII</b>
<b>INTRODUCCIÓN.....</b>	<b>11</b>
<b>CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....</b>	<b>16</b>
1.1 Sistema Nacional de Salud.....	16
1.2 Informatización del Sistema Nacional de Salud.....	17
1.2.2 Registro Informatizado de Salud (RIS).....	19
1.2.3 Sistema de Información para la Salud.....	20
1.3 Análisis de las soluciones existentes.....	21
1.4 Políticas para el desarrollo de la informatización del Sistema Nacional de Salud.....	22
1.5 Tendencias y tecnologías actuales a considerar.....	22
1.5.1 Software libre.....	22
1.5.2 Servicios Web.....	23
1.5.3 Arquitectura de Software.....	24
1.5.4 Plataforma de servicio (PlaSer).....	27
1.5.5 Servicio Web Apache.....	28
1.5.6 Lenguajes de Programación Web.....	28
1.5.7 Sistema de Gestión de Base de Datos.....	32
1.5.8 Metodologías de Desarrollo de Software.....	33
1.5.9 Herramientas a utilizar.....	35
<b>CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA.....</b>	<b>38</b>



2.1	Situación Problémica y Problema a Resolver.....	38
2.2	Modelo de Dominio.....	39
2.2.1	Conceptos Fundamentales.....	39
2.2.2	Diagrama del Modelo de Dominio.....	39
2.3	Propuesta del Sistema.....	40
2.3.1	Especificación de Requerimientos de Software.....	40
2.3.2	Modelo de Casos de Uso del Sistema.....	44
<b>CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA .....</b>		<b>56</b>
3.1	Modelo de Análisis.....	56
3.1.2	Diagramas de Clases de Análisis.....	56
3.1.3	Diagramas de Interacción.....	61
3.2	Modelo de Diseño.....	67
3.2.1	Justificación del uso de Patrones.....	68
3.2.2	Estructura del Diseño.....	69
3.2.3	Definición de Elementos de Diseño.....	71
3.2.4	Diagrama de Clases del Diseño.....	72
3.2.5	Descripción de las clases y atributos.....	77
<b>CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA .....</b>		<b>89</b>
4.1	Justificación de integración con otros sistemas.....	89
4.2	Modelo de Implementación.....	90
4.2.1	Diagramas de Componentes.....	91
4.2.2	Diagrama de Despliegue.....	93
4.2.3	Descripción de los Métodos.....	95
4.3	Estándares de diseño, codificación y tratamiento de errores.....	96
4.4	Prueba.....	102

4.4.1 Métodos de Prueba.....	102
<b>CONCLUSIONES .....</b>	<b>106</b>
<b>RECOMENDACIONES.....</b>	<b>107</b>
<b>REFERENCIAS BIBLIOGRÁFICAS.....</b>	<b>108</b>
<b>BIBLIOGRAFÍA.....</b>	<b>110</b>
<b>ANEXOS .....</b>	<b>112</b>
<b>GLOSARIO DE TÉRMINOS .....</b>	<b>114</b>

## INTRODUCCIÓN

El futuro de la humanidad dependerá en gran medida del potencial humano, de la gestión de la producción y de los conocimientos que se alcancen. La informática en sus diferentes manifestaciones tiene asegurado un papel protagónico en este futuro. Cuba, en momentos en que la globalización neoliberal arrasa despiadadamente con los más diversos escenarios, se propone su utilización justa y racional sobre principios éticos sostenibles.<sup>1</sup>

Cuba está consciente de que una sociedad para ser más eficaz, eficiente y competitiva debe aplicar la informatización en todas sus esferas y procesos y convencida de que para los países subdesarrollados resulta imprescindible el logro de este propósito, ya que su objetivo fundamental es lograr la supervivencia de sus pueblos.

En este sentido, se ha identificado desde muy temprano la conveniencia y necesidad de dominar e introducir en la práctica social las tecnologías de la información y las comunicaciones y lograr una cultura digital como una de las características imprescindibles del hombre nuevo, lo que facilitaría a la sociedad acercarse más hacia el objetivo de un desarrollo sostenible.<sup>2</sup>

Desde el triunfo de la Revolución Cubana, se están diseñando y organizando estrategias que permiten convertir los conocimientos y las tecnologías de la información y las comunicaciones en instrumentos a disposición del avance de las profundas transformaciones de la sociedad cubana y la Salud Pública. Es uno de los sectores que se ha favorecido por los impactos de las nuevas tecnologías, donde el eje fundamental y centro del proceso de informatización del sector lo constituye el paciente, quién será el

---

<sup>1</sup> DELGADO RAMOS ARIEL, VIDAL LEDO MARÍA Informática en la salud pública cubana *Revista Cubana Salud Pública*, 2006.

<sup>2</sup> GONZÁLEZ VALDÉS DAMARIS, ALEMÁN SÁNCHEZ PEDRO CARLOS ,&, LAURA DÍAZ ORTEGA. *Impacto de la informatización en la sociedad y estomatología cubanas* La Habana, Facultad de Ciencias Médicas de La Habana 2007.

principal beneficiado al garantizar las aplicaciones, la calidad, oportunidad y consistencia de la información, lo que incrementará la efectividad y eficiencia de los procesos relacionados con la salud.

La informatización del Sistema Nacional de Salud Pública (SNS) está dada por el conjunto de métodos, técnicas, procedimientos y actividades gerenciales dirigidas al manejo de la información en salud, la cual comprende la información sobre el estado de salud de la población, la información sobre el conocimiento de las ciencias de la salud y la información en general para la toma de decisiones, clínico-epidemiológicas, operativas y estratégicas.

En el 2003 el Ministerio de Salud Pública como organismo rector del Sistema Nacional de Salud define como una prioridad su informatización, con el propósito de elevar la eficiencia, seguridad, calidad y estética en todas las direcciones. Para ello convoca a un grupo de instituciones propias de sector, del Ministerio de Informática y Comunicaciones y de otros organismos de la administración central del estado, para definir de conjunto la estrategia a desarrollar. En algunos casos se ha tomado como punto de partida sistemas ya desarrollados en el país en el marco de aquella primera estrategia de desarrollo.<sup>3</sup>

Para cumplir este objetivo se le asigna a SOFTEL, empresa especializada en brindar soluciones informáticas para la salud, la misión de desarrollar productos y servicios informáticos.

Esta empresa del Ministerio de la Informática y las Comunicaciones (MIC), utiliza una estrategia nunca antes concebida en un proceso de desarrollo de software en el país, organiza el proceso productivo a través de una eficiente gestión de requerimientos. En ella participan, desde un inicio, médicos y trabajadores de la salud, vinculados directamente a la Atención Primaria de la Salud en calidad de expertos funcionales en estrecho vínculo con los especialistas de Informática, a través de una metodología con la que se logró describir en tiempo récord y con ninguna experiencia anterior los procesos que se propusieron para automatizar, lo que representa en el mundo actual del desarrollo de software el 75% del éxito de un proyecto. La experiencia de este trabajo en SOFTEL debe constituir el

---

<sup>3</sup> Ídem referencia 1

inicio de buenas prácticas en la producción de software con alta calidad y un ejemplo de normativa para los proyectos que deben irse abriendo de ahora en adelante en la informatización del sector de la salud.<sup>4</sup>

Actualmente se han trazado los primeros pasos para la reorientación del SNS en la Atención Primaria de Salud, proyecto con el que se persigue implementar un Sistema Integral de Informatización del SNS que abarque todos los niveles de atención. Es en este momento que comienza el desarrollo del Registro Informatizado de Salud (RIS), implementándose cinco módulos iniciales.

Conjuntamente con esta informatización, surge el proyecto Atención Primaria de la Salud (APS), el cual se lleva a cabo por especialistas de Softel, estudiantes y profesores pertenecientes a la Universidad de las Ciencias Informáticas (UCI) y expertos funcionales del Ministerio de Salud Pública.

Para el mejor funcionamiento del Sistema de Información para la Salud (SISalud) surge la necesidad de contar con información no perteneciente al dominio de Salud, al no encontrarse disponible la misma por parte de las instituciones encargadas de brindarla. Al comenzar a desarrollarse el RIS, se hizo necesario disponer de un componente que suministrara los datos fundamentales del carné de identidad de los ciudadanos del país, surgiendo la necesidad de simular un sistema que brindara esta información de forma centralizada. Esta situación condujo al surgimiento del Registro de Ciudadanos (RC), concebido con capacidad de gestionar los datos de los ciudadanos cubanos, sean pacientes o personal de la salud, brindando información estándar a todos los componentes de SISalud y evitando la duplicidad de la misma.

En los inicios, al pensar en lo que sería este Registro, se omitieron aspectos importantes como la diferencia que existe entre los datos del carné de identidad de los ciudadanos cubanos militares y los ciudadanos cubanos civiles, además existen pacientes extranjeros en Cuba recibiendo los beneficios de la salud cubana y personal de la salud extranjeros residentes en el país brindando sus servicios. La información referente de estas personas, cubanos militares y extranjeros, no es recogida y ni gestionada, hasta estos momentos en el Registro de Ciudadanos.

---

<sup>4</sup> MARÍN DÍAZ, M. Consideraciones sobre el proyecto de informatización de la Atención Primaria de Salud *Revista Habanera de Ciencias Médicas*, 2004, 3.

Teniendo en cuenta lo expuesto anteriormente se plantea como **Problema a Resolver**: ¿Cómo gestionar de forma eficiente la información referente a los Ciudadanos Cubanos, militares y civiles y a los Ciudadanos Extranjeros en el Sistema Nacional de Salud?

Se define como **Objeto de Estudio** el Proceso de gestión de la información en el Sistema Nacional de Salud.

El **Campo de Acción** se enfoca en el proceso de gestión de la información de los Ciudadanos Cubanos y Extranjeros en el Sistema Nacional de Salud.

El presente trabajo tiene como **Objetivo General** efectuar un mantenimiento correctivo y evolutivo del Registro de Ciudadanos, que garantice la información estándar para todos los componentes de Sistema de Información para la Salud, referente a los Ciudadanos Cubanos, militares y civiles y a los Ciudadanos Extranjeros.

Para dar cumplimiento al objetivo, se definen las siguientes **Tareas de la Investigación**:

- Identificar la información necesaria para el Sistema de Información para la Salud, relacionada con los Ciudadanos Cubanos y Extranjeros.
- Asimilar la arquitectura definida por el MINSAP para el desarrollo de sus aplicaciones, orientada a servicios y basada en componentes (OBA\_CBA), PLASER y el Registro Informatizado de la Salud.
- Describir las herramientas y tecnologías informáticas para el mantenimiento del Registro de Ciudadanos.
- Identificar elementos de integración con otros componentes del Sistema Nacional de Salud y el impacto que pueden producir en estos, los cambios en el registro actual.
- Modelar, siguiendo el Proceso Unificado de Desarrollo, los Flujos de Trabajo: “Modelamiento del Negocio”, “Gestión de Requerimientos”, “Análisis”, “Diseño”, “Implementación” y “Pruebas”, en su totalidad, por no existir la documentación previa del sistema actual.

- Implementar el Registro de Ciudadanos 2.0, utilizando los estándares de codificación y de tratamiento de errores o excepciones definidos.

La investigación realizada se estructurará en cuatro capítulos como sigue:

**Capítulo 1. Fundamentación Teórica:** Se describen los conceptos fundamentales. Antecedentes y la situación problemática relacionada con el dominio del problema. Además, se hace un estudio de las técnicas, tendencias y tecnologías a considerar, en las que se apoya la solución del problema.

**Capítulo 2. Características del Sistema:** Se exponen los conceptos fundamentales de las entidades involucradas, mediante un modelo de dominio. Se presentan los requerimientos funcionales y no funcionales. Se describen los casos de uso del sistema y se justifican los actores.

**Capítulo 3. Análisis y Diseño del Sistema:** Se realiza el diseño de la solución propuesta. Se realiza la justificación del uso de patrones y subsistemas de servicios. Se presentan los diagramas de clases de diseño y se describen las clases y atributos. Además se presentan los diagramas de clases persistentes, los modelos de datos y las descripciones de las tablas y atributos del Registro de Ciudadanos 2.0.

**Capítulo 4. Implementación y Pruebas:** Se describe el proceso de implementación a partir del diseño de la solución anteriormente propuesta. Se justifica la integración con otros componentes del Registro Informatizado de Salud. Se muestran los diagramas de componentes del Registro de Ciudadanos 2.0, así como el diagrama de despliegue. Se describen los métodos del negocio. Se detallan las pautas que se definieron para estandarizar las interfaces, así como los estándares de codificación y tratamiento de errores.

## CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

En este capítulo, se abordarán distintos aspectos que serán utilizados como soporte teórico del sistema que se pretende analizar. Se fundamentan el objeto de estudio, un breve estudio acerca de las aplicaciones vinculadas al campo de acción y el estado del arte de las tecnologías y herramientas utilizadas para el análisis del sistema propuesto.

### 1.1 Sistema Nacional de Salud.

El triunfo de la Revolución Cubana, en enero de 1959, pone fin a unos duros años en la salud del pueblo y como profundo proceso político social y por una clara voluntad, se traza ya el desarrollo del Sistema Nacional de Salud.<sup>5</sup>

El Ministerio de Salud Pública (MINSAP) es el Organismo rector del Sistema Nacional de Salud. Encargado de dirigir, ejecutar y controlar la aplicación de la política del Estado y del Gobierno en cuanto a la Salud Pública, el desarrollo de las Ciencias Médicas y la Industria Médico Farmacéutica.

El Sistema Nacional de Salud tiene carácter estatal y social de la medicina, accesibilidad y gratuidad de los servicios, orientación profiláctica, aplicación adecuada de los adelantos de la ciencia y la técnica, participación de la comunidad e intersectorialidad, colaboración internacional, centralización normativa y descentralización ejecutiva.<sup>6</sup>

El Sistema Nacional de Salud se estructura en tres niveles que se corresponden con la estructura político-administrativa del país (**Ver Anexo 1**). El nivel nacional está representado por el Ministerio de Salud Pública y es el órgano rector con funciones metodológicas, normativas y de coordinación y control al cual

---

<sup>5</sup> DOTRES, C. *Revista Cubana de Medicina Tropical*, 1997.

<sup>6</sup> INFOMED. Disponible en: [http://www.sld.cu/sistema\\_de\\_salud/aspectos.html](http://www.sld.cu/sistema_de_salud/aspectos.html)



se le subordinan directamente los centros universitarios, institutos de investigaciones, centros hospitalarios de asistencia médica altamente especializados, centros de distribución y comercializadoras de suministros y tecnologías médicas así como otros centros y entidades nacionales destinados a actividades técnicas y de apoyo.

Los otros dos niveles están representados por las direcciones provinciales y municipales de salud que agrupan las instituciones de salud a su respectivo nivel y que, al igual que en el nivel central, se subordinan desde el punto de vista administrativo a las estructuras de Gobierno en los distintos niveles organizativos, representando sus intereses ante ellos y dando respuesta a las demandas y necesidades de la población.<sup>7</sup>

El Sistema Nacional de Salud presenta otra clasificación según la atención a la población: Atención Primaria, Atención Secundaria y Atención Terciaria.

La Atención Primaria: Representa el primer nivel de contacto con el Sistema Nacional de Salud con los individuos, la familia y la comunidad.

La Atención Secundaria: Se brinda a nivel de las instituciones hospitalarias, por lo general son de carácter provincial.

La Atención Terciaria: Es aquella que por su condición muy especializada, solo se brinda en determinados centros como el Instituto de Nefrología, Instituto de Cirugía Cardiovascular, entre otros.

### **1.2 Informatización del Sistema Nacional de Salud.**

En el año 2003 el MINSAP y el Ministerio de la Informática y las Comunicaciones, acuerdan un grupo de acciones para garantizar definitivamente la Informatización del SNS como un paso más en el desarrollo estratégico en cuanto a las alianzas externas y máxima expresión de los cambios que se introducen en el sector. Se logra una integración política, estratégica y de acción que asegura una adecuada introducción

---

<sup>7</sup> RAMÍREZ MÁRQUEZ ABELARDO, CASTELL-FLORIT SERRATE,&, PASTOR MESA GUILLERMO El Sistema Nacional de Salud de Cuba, 2003.

de la tecnología de la información y las comunicaciones en todo el sistema y ofrece la posibilidad de obtener nuevas fuentes de exportación al país.<sup>8</sup>

La informatización del Sistema Nacional de Salud permitirá perfeccionar la calidad asistencial ofrecida a la sociedad, así como facilitar las funciones del personal de la salud, partiendo de Sistema de Atención Primaria y tomando como eje al policlínico. De manera que facilite la gestión de la información en los niveles de atención primaria, perfeccione el flujo de información hacia los niveles superiores y garantice y haga más eficiente la toma de decisiones, la investigación y la docencia.<sup>9</sup>

La informatización de la Salud se espera que aporte notables beneficios. A continuación se mencionan algunos:

Para la población:

- Una equidad distribuida de acceso a servicios, tecnologías e información de salud independiente de áreas geográficas ni niveles de atención.
- Disfrutaría la sensación de ser atendido por un personal médico mejor preparado y actualizado.
- Reducción de tiempos de esperas para el acceso a servicios especializados con la posibilidad de recibirlos en su propio escenario social.
- Reducción del número de desplazamientos innecesarios entre instituciones de salud.

Para el Sistema Nacional de Salud:

- Gestión oportuna de una información confiable y actualizada que proporcionará una optimización considerable de recursos con su lógico resultado en la reducción significativa de costos de operación de las entidades que conforman el Sistema Nacional de Salud.

---

<sup>8</sup> Ídem referencia 1.

<sup>9</sup> DELGADO RAMOS, A. Informatización Sistema Nacional de salud, 2003.

- Elevación de la capacidad y calidad de las tomas de decisiones asistenciales y gerenciales por la disposición oportuna de información actualizada para todos los niveles del Sistema Nacional de Salud.
- Disponer de un soporte y herramientas poderosas para la formación y actualización constante de sus miembros desde sus propios escenarios de desempeño, potenciando la investigación científica multi-céntrica nacional e internacional.<sup>10</sup>

### 1.2.2 Registro Informatizado de Salud (RIS).

El Registro Informatizado de Salud “RIS” es la solución propuesta por el MINSAP para la informatización de la Salud Pública, más que un software, es una plataforma de aplicaciones, abierta, con una interfaz de programación que permite incorporar nuevos módulos compatibles entre sí. Es portable a diferentes sistemas operativos, tanto en los servidores como en los clientes, replicable en otros entornos. Es una plataforma en constante desarrollo, que crece en la medida en que se implementan nuevos módulos.

Para dar cumplimiento a los objetivos propuestos se utilizó la metodología de la dirección integrada de proyectos y se organizó el proceso en 4 grupos de trabajo en la Empresa Softel del Ministerio de Informática y Comunicaciones, integrados por especialistas de la propia Empresa, de la empresa ESATEL de Santiago de Cuba, del Centro para el Desarrollo Informático de la Salud (CEDISAP) e Infomed, estos dos últimos pertenecientes al Ministerio de Salud Pública.<sup>11</sup>

El Registro Informatizado de Salud constituye la materialización de la estrategia metodológica de Informatización del Sistema Nacional de Salud como parte de la Informatización de la Sociedad. Fue desarrollado e implementado siguiendo las políticas para el desarrollo informático aprobadas por el sector a finales del 2003. Es una aplicación orientada a servicios bajo una arquitectura moderna y versátil, es un

---

<sup>10</sup> TURRUELLES TEJEDA YOSVANI, GONZALEZ DIAZ MAIKEL *Registro de Ubicación Geográfica y Registro de Localidades para el Sistema de Información para la Salud*. Ciudad de La Habana, Universidad de las Ciencias Informáticas, 2007. p.

<sup>11</sup> DELGADO RAMOS ARIEL, CABRERA HERNÁNDEZ MIRNA ,&, JUNCAL VIRGINIA *Registro Informatizado de Salud (RIS) Temas Estadísticos de Salud*, 2005.

producto portable que no depende del motor de base de datos escogido para su desarrollo y funciona tanto con el Sistema Operativo “Windows” como en “Linux”.<sup>12</sup>

El proceso de administración del RIS tiene un carácter jerárquico, el administrador nacional crea los editores y visualizadores para su propio nivel así como los administradores a nivel inferior (**Ver Anexo 2**).

En el año 2004 surge el proyecto Atención Primaria de la Salud (APS) el cual tiene como misión perfeccionar componentes existentes en el RIS y desarrollar otros que complementen la nueva propuesta de Informatización del Sistema Nacional de Salud.

### 1.2.3 Sistema de Información para la Salud.

Para lograr la informatización en el sector de la Salud se pretende en primer lugar que todos los módulos estén incluidos en un conjunto de aplicaciones que formarán parte del Sistema de Información para la Salud (SISalud), el cual está formado por el Registro Informatizado de Salud (RIS), el Sistema Informatizado de Atención Primaria (SIAP), el Sistema Informatizado de Gestión Hospitalaria (SIGH) y el Sistema Informatizado de Atención Especializada (SIAE).<sup>13</sup>

El Registro Informatizado de Salud (RIS) tiene como objetivos:

- Proveer al Sector de la Salud de una información única y confiable en tiempo real para la toma de decisiones en los diferentes niveles de dirección, la docencia, investigación y la gestión en salud.
- Implementar una solución informática integral.

El Sistema Informatizado de Atención Primaria (SIAP), en el que se incluirán en la etapa actual los módulos específicos de este nivel de atención, constituirá una nueva herramienta para la transformación de los servicios que se brindan en este nivel, ya que integrarán diversos subsistemas como las actividades diarias del EBS, la dispensarización y la planificación de las acciones de salud, tanto individual como

---

<sup>12</sup> Ídem referencia 11.

<sup>13</sup> CABRERA HERNÁNDEZ, M. *Propuesta de Esquema del Sistema de Información para la Salud (SISalud)*. La Habana, SOFTEL, 2006.

familiar. Entre ellos el Registro de Población (RPOB), el Registro de Actividades Diarias (RAD) y otros que se irían incorporando según las prioridades y necesidades del usuario.

El Sistema Informatizado de Gestión Hospitalaria (SIGH), en el que se agruparán los módulos que pertenecen al nivel de atención secundario u hospitalario. Entre ellos el Registro de Autopsias.

El Sistema Informatizado de Atención Especializada (SIAE), en el que se agruparán los módulos que pertenecen al nivel de atención terciaria.<sup>14</sup>

### **1.3 Análisis de las soluciones existentes.**

#### **Registro de Ciudadanos del RIS.**

El Registro de Ciudadanos (RC) permite contar con un registro único en el país, con los datos del Carné de Identidad de cualquier ciudadano residente en Cuba, que trabaje en la salud y/o reciba sus servicios. La información fundamental que gestiona este registro está relacionada con los Datos Generales (Carné de Identidad, Nombre, Apellidos, Sexo, Color de la piel y los ojos, Peso y Talla); Datos de Nacimiento (Provincia, Municipio y Fecha); Dirección, Nombre de los padres, Tomo y Folio.

La información en el Registro del Ciudadano se introduce desde otras aplicaciones de SISalud, que acceden a sus servicios Web para introducirlos. Permite introducir los datos del personal que trabaja en la salud en el Registro de Personal de la Salud, de los estudiantes que pueden ser ubicados en las áreas de salud en el Registro de Estudiantes y de los pacientes en el Registro de Población. También se puede acceder al Registro del Ciudadano en el nivel Nacional para gestionar su información.

Este sistema resuelve parcialmente el problema existente, ya que brinda información de los Ciudadanos Cubanos y gestiona información de estos. Sin embargo, es de vital importancia para el buen funcionamiento de otros componentes, además de manejar datos de los ciudadanos cubanos, contar con información referente a los ciudadanos cubanos militares y ciudadanos extranjeros residentes o no en el país, los cuales pueden también, recibir servicios de salud y/o trabajar en este sector. Por las razones

---

<sup>14</sup> Ídem referencia 13

anteriormente expuestas este Registro de Ciudadanos no responde en su totalidad a las exigencias actuales del Sistema de Información para la Salud.

### **1.4 Políticas para el desarrollo de la informatización del Sistema Nacional de Salud.**

A partir de 2001 se realizaron profundos cambios en métodos y estilos de trabajo en el sector de la salud que propiciaron un salto cualitativo en el desarrollo estratégico de la informatización. Actualmente se trabaja en proyectos basados en nuevas tecnologías de Internet (XML, PHP, Web Services), documentación de todo el proceso productivo, requisitos de seguridad del software, independencia de la base de datos, desarrollo en multiplataforma, empleo de estándares internacionales para los productos relacionados con la salud y software libre (PHP, MySQL, Linux) y otros que garantizan una explotación integrada y compatible y que serán introducidos utilizando como infraestructura la Red Telemática de Salud.<sup>15</sup>

Por las razones antes expuestas es que se decide hacer una investigación de estas tendencias y tecnología en la que se desarrolla la informatización del Sistema Nacional de Salud.

### **1.5 Tendencias y tecnologías actuales a considerar.**

En este epígrafe se tratarán los conceptos fundamentales relacionados con las herramientas, metodologías, lenguajes de programación y arquitectura que se consideran para el proceso de desarrollo del sistema a desarrollar, teniendo en cuenta las políticas definidas por el MINSAP expuestas en el epígrafe anterior.

#### **1.5.1 Software libre.**

Software Libre se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software.

---

<sup>15</sup> Ídem referencia 1

El Software Libre es un asunto de libertad, no de precio. Para entender el concepto, se debe pensar en libre como en libertad de expresión, no como en cervezas gratis.

De modo más preciso, se refiere a cuatro libertades de los usuarios del software:

- La libertad de usar el programa, con cualquier propósito (libertad 0).
- La libertad de estudiar el funcionamiento del programa, y adaptarlo a las necesidades (libertad 1).
- La libertad de distribuir copias, con lo que puede ayudar a otros (libertad 2).
- La libertad de mejorar el programa y hacer públicas las mejoras, de modo que toda la comunidad se beneficie (libertad 3).

Software Libre no significa no comercial. Un programa libre debe estar disponible para uso comercial, desarrollo comercial y distribución comercial. El desarrollo comercial del software libre ha dejado de ser inusual; el software comercial libre es muy importante.<sup>16</sup>

## 1.5.2 Servicios Web.

Existen múltiples definiciones sobre lo que son los Servicios Web, lo que muestra su complejidad a la hora de dar una adecuada definición que englobe todo lo que son e implican. Una posible sería hablar de ellos como un conjunto de aplicaciones o de tecnologías con capacidad para ínter operar en la Web. Estas aplicaciones o tecnologías intercambian datos entre sí con el objetivo de ofrecer unos servicios. Los proveedores ofrecen sus servicios como procedimientos remotos y los usuarios solicitan un servicio llamando a estos procedimientos a través de la Web.

Estos servicios proporcionan mecanismos de comunicación estándares entre diferentes aplicaciones, que interactúan entre sí para presentar información dinámica al usuario. Para proporcionar interoperabilidad y extensibilidad entre estas aplicaciones, y que al mismo tiempo sea posible su combinación para realizar operaciones complejas, es necesaria una arquitectura de referencia estándar.

---

<sup>16</sup> GNU. *sitio Web de GNU*. Disponible en: <http://www.gnu.org/philosophy/free-sw.es.html>

Un Web Service es un componente de software que se comunica con otras aplicaciones codificando los mensajes en XML y enviando estos mensajes a través de protocolos estándares de Internet tales como el Hypertext Transfer Protocol (HTTP). Intuitivamente un Web Service es similar a un sitio Web que no cuenta con un interfaz de usuario y que da servicio a las aplicaciones en vez de a las personas.

Un Web Service, en vez de obtener solicitudes desde el navegador y retornar páginas Web como respuesta, lo que hace es recibir solicitudes a través de un mensaje formateado en XML desde una aplicación, realiza una tarea y devuelve un mensaje de respuesta también formateado en XML.<sup>17</sup>

### **1.5.3 Arquitectura de Software.**

La Arquitectura de Software es, a grandes rasgos, una vista del sistema que incluye los componentes principales del mismo, la conducta de esos componentes según se la percibe desde el resto del sistema y las formas en que los componentes interactúan y se coordinan para alcanzar la misión del sistema.

La Arquitectura de Software es la organización fundamental de un sistema encarnada en sus componentes, las relaciones entre ellos y el ambiente y los principios que orientan su diseño y evolución.<sup>18</sup>

#### **Arquitectura Cliente/Servidor.**

La arquitectura cliente/servidor es un modelo para el desarrollo de sistemas de información en el que las transacciones se dividen en procesos independientes que cooperan entre sí para intercambiar información, servicios o recursos. Se denomina cliente al proceso que inicia el diálogo o solicita los recursos y servidor al proceso que responde a las solicitudes.

En este modelo las aplicaciones se dividen de forma que el servidor contiene la parte que debe ser compartida por varios usuarios, y en el cliente permanece sólo lo particular de cada usuario.

---

<sup>17</sup> GONZÁLEZ C., B. *XML Web Services. DesarrolloWeb.com*

<sup>18</sup> REYNOSO CARLOS, B. *Introducción a la Arquitectura de Software* Universidad de Buenos Aires, 2004.



Entre las principales características de la arquitectura cliente/servidor se pueden destacar las siguientes:

- El servidor presenta a todos sus clientes una interfaz única y bien definida.
- El cliente no necesita conocer la lógica del servidor, sólo su interfaz externa.
- El cliente no depende de la ubicación física del servidor, ni del tipo de equipo físico en el que se encuentra, ni de su sistema operativo.
- Los cambios en el servidor implican pocos o ningún cambio en el cliente.

### **Arquitectura en tres capas.**

El modelo de 3 capas propone un ambiente para la construcción y ejecución de aplicaciones de avanzada que muy probablemente reemplazará a los sistemas actuales. Una de sus mayores ventajas es que los sistemas se independizan en cierta forma de la capacidad tecnológica y el tamaño del negocio, por lo que pueden acompañar de manera eficiente el crecimiento de las empresas que los utilizan. Dadas las características del modelo, se puede implementar y dejar operativa una solución de negocios en tiempos extremadamente cortos, permitiendo conseguir una ventaja competitiva particular respecto a otros negocios. También permite la modificación del sistema en períodos de tiempo reducidos, incluso cuando es necesario agregar características especiales a las aplicaciones. (**Ver Anexo 3**).<sup>19</sup>

- Capa de Presentación: Es la que ve el usuario, presenta el sistema al usuario, le comunica la información y captura la información del usuario dando un mínimo de proceso (realiza un filtrado previo para comprobar que no hay errores de formato). Esta capa se comunica únicamente con la capa de negocio.
- Capa de Negocio: Es donde residen los programas que se ejecutan, recibiendo las peticiones del usuario y enviando las respuestas tras el proceso. Se denomina capa de negocio (e incluso de lógica del negocio) pues es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los

---

<sup>19</sup> ESTR@TEGIAMAGAZINE. Disponible en: [www.e-estrategia.com](http://www.e-estrategia.com)

resultados, y con la capa de datos, para solicitar al gestor de base de datos para almacenar o recuperar datos de él.

- Capa de Datos: Es donde residen los datos. Está formada por uno o más gestores de bases de datos que realiza todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.<sup>20</sup>

### **Desarrollo de Software Basado en Componentes.**

Desarrollo de Software Basado en Componentes (DSBC), trata de sentar las bases para el diseño y desarrollo de aplicaciones distribuidas basadas en componentes software reutilizables.<sup>21</sup>

El DSBC introduce los términos de Alta Cohesión y Bajo Acoplamiento:

- Alta Cohesión: Dice que la información que almacena un componente debe ser coherente y está en la mayor medida de lo posible relacionada con otro componente.
- Bajo Acoplamiento: Es tener los componentes lo menos ligados entre sí que se pueda. De tal forma, en caso de producirse una modificación en alguno de ellos, se tendrá la mínima repercusión posible en el resto de los componentes, potenciando la reutilización, y disminuyendo la dependencia entre los componentes.

### **Arquitectura Orientada a Servicio (SOA).**

Una arquitectura orientada al servicio define cómo interactúan dos entidades de cómputo de tal manera que permita a una entidad efectuar una unidad de trabajo a nombre de la otra. La unidad de trabajo se conoce como servicio, y la interacción entre servicios se define utilizando un lenguaje descriptivo. Web Services basados en SOAP (Simple Object Access Protocol, o protocolo de acceso simple a objetos) se

---

<sup>20</sup> *Arquitectura de Software a emplear en los componentes del Sistema de Información para la Salud.* La Habana, SOFTEL, 2006.

<sup>21</sup> FUENTES LIDIA, TROYA JOSÉ, &, VALLECILLO ANTONIO *Desarrollo de Software Basado en Componentes* Universidad de Málaga.

convierten en la implementación más común de SOA. Sin embargo, hay implementaciones de SOA independientes de los Web Services que proveen los mismos beneficios. La independencia de protocolo del SOA indica que diferentes consumidores pueden usar los servicios comunicándose con él en diferentes maneras. Idealmente, debe existir una capa de administración entre los proveedores y los consumidores para garantizar flexibilidad completa en cuanto a protocolos de implementación.<sup>22</sup>

### 1.5.4 Plataforma de servicio (PlaSer).

“PlaSer”, acrónimo de Plataforma de Servicio, constituye una plataforma sobre la que se pueden desplegar aplicaciones “XML\_WebServices”. Este sistema está concebido completamente sobre Arquitectura Basada en Componentes y Orientada a Servicios, usando el paradigma de “XML\_WebServices”, específicamente “SOAP”. En su concepción se han utilizado estándares actuales y normas abiertas además de un enriquecido lenguaje de patrones entre los que podemos encontrar (“MVC, Fachada, Alta Cohesión, Bajo Acoplamiento, SAAA, Proxy, etc.”).

“PlaSer” es un “Framework” que está constituido o integrado por una colección de clases desarrolladas en PHP, una librería o paquete, que puede o no ser usada para que un componente se integre al Sistema de Información para la Salud.

Desde el punto de vista estructural, esta plataforma permite trabajar con cualquier base de datos que cumpla con la norma SQL ANSI 92; pero desde el punto de vista de implementación solo trabaja con las bases de datos soportadas por el componente DBX, ya que encapsula a dicho componente y lo utiliza para el acceso a bases de datos.<sup>23</sup>

---

<sup>22</sup> DACCACH T, J. SOA, 2007.

<sup>23</sup> LEÓN GARCÉS , J. and Y. FROMETA MORENO. *Plataforma de Servicios PlaSer*. La Habana, Universidad de las Ciencias Informáticas, 2008.

## 1.5.5 Servicio Web Apache.

Apache es el servidor Web hecho por excelencia, su configurabilidad, robustez y estabilidad hacen que cada vez millones de servidores reiteren su confianza en este programa.

Corre en una multitud de Sistemas Operativos, lo que lo hace prácticamente universal. Apache es una tecnología gratuita de código fuente abierto. Es un servidor altamente configurable de diseño modular. Es muy sencillo ampliar las capacidades del servidor Web Apache.<sup>24</sup>

## 1.5.6 Lenguajes de Programación Web.

En el desarrollo de Aplicaciones Web, son utilizados dos tipos fundamentales de lenguajes, los lenguajes del lado del servidor, y los lenguajes del lado del cliente.

Entre los lenguajes del lado del servidor podemos citar algunos, como son, PERL, ASP, ColdFunction, PHP, JSP, etc. Y entre los lenguajes que trabajan del lado del cliente se encuentran JavaScript, XSLT y Visual Basic Script.

### **ASP.**

Active Server Pages (ASP) es una tecnología del lado servidor de Microsoft para páginas Web generadas dinámicamente. La tecnología ASP está estrechamente relacionada con el modelo tecnológico de su fabricante. Intenta ser solución para un modelo de programación rápida ya que programar en ASP es como programar en Visual Basic, por supuesto con muchas limitaciones ya que es una plataforma que no se ha desarrollado como lo esperaba Microsoft.

### **ASP.NET.**

Una nueva versión del tradicional Active Server Pages, ASP.net forma parte del .NET Framework de Microsoft.

---

<sup>24</sup> WWW.CIBERAULA.COM. 2006. [Disponible en: [http://linux.ciberaula.com/articulo/linux\\_apache\\_intro/](http://linux.ciberaula.com/articulo/linux_apache_intro/)]

Es mucho mejor que el ASP tradicional, ASP.net trae diversas mejoras entre las cuales se destacan:

- Rendimiento: la aplicación de compila en una sola vez al lenguaje nativo, y luego, en cada petición tiene una compilación Just In Time, es decir se compila desde el código nativo, lo que permite mucho mejor rendimiento.
- Rapidez en programación: mediante diversos controles, podemos con unas pocas líneas y en menos de 5 minutos mostrar toda una base de datos y hacer rutinas complejas.<sup>25</sup>

### **PERL.**

Perl significa Practical Extraction and Report Language, algo así como lenguaje práctico de extracción y de informes. Es un lenguaje interpretado, aunque en realidad, el intérprete de Perl, como todos los intérpretes modernos, compila los programas antes de ejecutarlos.

Perl permite resolver los problemas fáciles fácilmente, y resolver también problemas difíciles. Es rápido hacer una pequeña aplicación Web.

Las ventajas principales son el que se trata de un lenguaje muy maduro, que lleva mucho tiempo funcionando, y que tiene cientos de bibliotecas operativas y listas para ser usadas.<sup>26</sup>

#### **1.5.6.1 Lenguajes de Programación Web a utilizar.**

### **PHP 4.**

PHP es un lenguaje de scripting que permite la generación dinámica de contenidos en un servidor Web. El significado de sus siglas es HyperText Preprocessor. Entre sus principales características cabe destacar su potencia, su alto rendimiento, su facilidad de aprendizaje y su escasez de consumo de recursos.

---

<sup>25</sup> *ASP en Castellano*. Disponible en: [http://www.programacion.net/asp/articulo/aspnet\\_quees](http://www.programacion.net/asp/articulo/aspnet_quees)

<sup>26</sup> *Tutorial de Perl en español*. Disponible en: <http://geneura.ugr.es/~jmerelo/tutoperl/>

Posee como principales ventajas:

- Es un lenguaje multiplataforma.
- Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL.
- Leer y manipular datos desde diversas fuentes.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite las técnicas de Programación Orientada a Objetos.
- Fácil de utilizar.
- Código portable, de fácil interpretación y rápida ejecución.

### **XML 1.0.**

XML significa lenguaje de marcas generalizado (Extensible Markup Language). Es un lenguaje usado para estructurar información en un documento o en general en cualquier fichero que contenga texto, como por ejemplo ficheros de configuración de un programa o una tabla de datos. Comparado con otros sistemas usados para crear documentos, el XML tiene la ventaja de poder ser más exigente en cuanto a la organización del documento, lo cual resulta en documentos mejor estructurados.<sup>27</sup>

No debe pensarse que XML es para crear páginas Web. XML es un lenguaje que cambia el paradigma de programación: de basada en las funciones u objetos a la programación basada en el documento. XML se puede usar para cambiar totalmente el paradigma de publicación; de un programa que recibe unas entradas y produce unas salidas, se pasa a un documento que genera otro documento, o bien programas que toman documentos y producen otros documentos.<sup>28</sup>

---

<sup>27</sup> E. VILLATE, J. *Introducción al XML*, 2001.

<sup>28</sup> GUERVOS, J. M. *Introducción al lenguaje XML*.

## **XSL.**

El XSL es el acrónimo de *Extensible Style Language (Lenguaje de Estilo Extensible)*. Es un lenguaje que nos permite definir una presentación o formato para un documento XML. Un mismo documento XML puede tener varias hojas de estilo XSL que lo muestren en diferentes formatos (HTML, PDF, RTF, VRML, PostScript, sonido, etc.)

La aplicación de una hoja de estilos XSL a un documento XML puede ocurrir tanto en el origen (por ejemplo, un servlet que convierta de XML a HTML para que sea mostrado a un navegador conectado a un servidor Web), o en el mismo navegador (como en el caso del MS IE5, y en breve, Netscape 5).

Básicamente, XSL es un lenguaje que define una transformación entre un documento XML de entrada, y otro documento XML de salida.

Una hoja de estilo XSL es una serie de reglas que determina cómo va a ocurrir la transformación. Cada regla se compone de un patrón (pattern) y una acción o plantilla (template).

De este modo, cada regla afecta a uno o varios elementos del documento XML. El efecto de las reglas es recursivo, para que un elemento situado dentro de otro elemento pueda ser también transformado. Las hojas de estilo tienen una regla raíz que, además de ser procesada, llama a las reglas adecuadas para los elementos hijos.<sup>29</sup>

## **JavaScript.**

JavaScript es un lenguaje de programación que se utiliza principalmente para crear páginas web dinámicas. Una página Web dinámica es aquella que incorpora efectos como aparición y desaparición de texto, animaciones, acciones que se activan al pulsar botones u otros elementos y ventanas con mensajes de aviso al usuario.

Técnicamente, JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas escritos con JavaScript se

---

<sup>29</sup> *Extended Style Language*. Disponible en: <http://www.ulpgc.es/otros/tutoriales/xml/XSL.html>

pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios. A pesar de su nombre, no guarda ninguna relación directa con el lenguaje de programación Java.<sup>30</sup>

## 1.5.7 Sistema de Gestión de Base de Datos.

Un Sistema de Gestión de Base de Datos pueden definirse como un paquete generalizado de software, que se ejecuta en un sistema computacional anfitrión, centralizando los accesos a los datos y actuando de interfaz entre los datos físicos y el usuario. Las principales funciones que debe cumplir un SGBD se relacionan con la creación y mantenimiento de la base de datos, el control de accesos, la manipulación de datos de acuerdo con las necesidades del usuario, el cumplimiento de las normas de tratamiento de datos, evitar redundancias e inconsistencias y mantener la integridad.

### SQL Server 2000.

Es un potente motor de bases de datos de alto rendimiento capaz de soportar millones de registros por tabla con una interface intuitivo y con herramientas de desarrollo integradas como Visual Studio 6.0 o .NET, además incorpora un modelo de objetos totalmente programable (SQL-DMO) con el que se puede desarrollar cualquier aplicación que manipule componentes de SQL Server, es decir, hacer aplicación para crear bases de datos, tablas, DTS, backups, etc., todo lo que se pueda hacer desde el administrador del SQL Server y puede hacerlo no solo en Visual C++ sino también en Visual Basic, ASP y por supuesto en .NET.

### PostgreSQL.

Es un servidor de base de datos relacional orientada a objetos de software libre, liberado bajo la licencia BSD. Tiene como algunas ventajas que es:

- Extensible: El código fuente está disponible para todos sin costo.
- Multiplataforma: PostgreSQL está disponible en casi cualquier Unix (34 plataformas en la última versión estable), y ahora en versión nativa para Windows.

---

<sup>30</sup> EGUÍLUZ PÉREZ , J. Introducción a JavaScript.



Principal desventaja es que consume bastantes recursos y carga más el sistema.

### **1.5.7.1 Sistema de Gestión de Base de Datos a utilizar.**

#### **MySQL 4.0.3.**

MySQL, el sistema de gestión de bases de datos SQL Open Source más popular, lo desarrolla, distribuye y soporta MySQL AB. Es un sistema de gestión de bases de datos. Una base de datos es una colección estructurada de datos. Puede ser cualquier cosa, desde una simple lista de compra a una galería de pintura o las más vastas cantidades de información en una red corporativa.

MySQL software es Open Source. Open Source significa que es posible para cualquiera usar y modificar el software. Cualquiera puede bajar el software MySQL desde Internet y usarlo sin pagar nada. Si lo desea, puede estudiar el código fuente y cambiarlo para adaptarlo a sus necesidades.

El servidor MySQL fue desarrollado originalmente para manejar grandes bases de datos mucho más rápido que las soluciones existentes y ha estado siendo usado exitosamente en ambientes de producción sumamente exigentes por varios años. Aunque se encuentra en desarrollo constante, el servidor MySQL ofrece hoy un conjunto rico y útil de funciones. Su conectividad, velocidad, y seguridad hacen de MySQL un servidor bastante apropiado para acceder a bases de datos en Internet.

### **1.5.8 Metodologías de Desarrollo de Software.**

Las metodologías de desarrollo de software son un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos software. Es como un libro de recetas de cocina, en el que se van indicando paso a paso todas las actividades a realizar para lograr el producto informático deseado, indicando además qué personas deben participar en el desarrollo de las actividades y qué papel deben de tener. Además detallan la información que se debe producir como resultado de una actividad y la información necesaria para comenzarla.

## **Lenguaje Unificado de Modelado (UML).**

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

UML es un lenguaje de propósito general para el modelado orientado a objetos y también puede considerarse como un lenguaje de modelo visual que permite una abstracción del sistema y sus componentes, es un lenguaje gráfico y se puede aplicar en una gran variedad de formas para dar soporte a una metodología de desarrollo de software (tal como el Proceso Unificado de Desarrollo) pero no especifica en sí mismo qué metodología o proceso usar.

## **Proceso Unificado de Desarrollo (RUP).**

El Proceso Unificado Racional (*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, define “quién” (trabajadores) está haciendo “qué” (artefactos), “cuándo” (flujo de trabajos) y “cómo” (actividades) para alcanzar un determinado objetivo.

El proyecto realizado siguiendo RUP se basa en casos de uso para describir lo que se espera del software, es centrada en la arquitectura y es iterativo e incremental. Se divide en cuatro fases: Intercepción o inicio (puesta en marcha), Elaboración (definición, análisis, diseño), Construcción (implementación) y Transición (fin del proyecto y puesta en producción).

Es uno de los procesos más generales de los existentes actualmente, ya que en realidad está pensado para adaptarse a cualquier proyecto.

## 1.5.9 Herramientas a utilizar.

Después de haber de definido algunos aspectos de interés como: servicios Web, arquitectura, metodologías de desarrollo de software y otras, damos paso a realizar una selección de las herramientas que son las encargadas de dar rendimiento a las tecnologías explicadas anteriormente.

### **Zend Studio 5.0.**

Se trata de un programa de la casa Zend, impulsores de la tecnología de servidor PHP, orientada a desarrollar aplicaciones Web, como no, en lenguaje PHP. El programa, además de servir de editor de texto para páginas PHP, proporciona una serie de ayudas que pasan desde la creación y gestión de proyectos hasta la depuración de código.

Zend Studio consta de dos partes en las que se dividen las funcionalidades de parte del cliente y las del servidor. Las dos partes se instalan por separado, la del cliente contiene el interfaz de edición y la ayuda. Permite además hacer depuraciones simples de scripts, aunque para disfrutar de toda la potencia de la herramienta de depuración habrá que disponer de la parte del servidor, que instala Apache y el módulo PHP o, en caso de que estén instalados, los configura para trabajar juntos en depuración.

Si deseamos aumentar la productividad en nuestros desarrollos PHP no cabe duda que este programa puede ayudarnos seriamente. Todas las opciones que dispone están pensadas con acierto por personas que conocen como nadie la tecnología.

### **MySQL-Front 3.0.**

MySQL-Front es una sencilla pero útil aplicación diseñada especialmente para desarrolladores que trabajan con MySQL. Con MySQL-Front se pueden realizar acciones básicas como añadir, borrar o modificar tablas, campos, registros, entre muchas otras opciones.

## **Stylus Studio 5.0.**

Stylus Studio es un completo entorno de desarrollo integrado que incluye un potente editor de XML, un "debugger" XSLT y otras muchas herramientas pensadas especialmente para facilitar y mejorar tu productividad en el desarrollo de sitios Web.

Permite la edición de XML en modo visual y sincronizado, y un completo set de herramientas para desarrollo en XSLT y una utilidad de diseño de hojas de estilo de HTML a XSLT. Soporta además edición visual de XQuery, e incluye un editor DTD, utilidades para XPath, y mucho más.

## **Macromedia Dreamweaver 8.**

El programa Dreamweaver de Macromedia es uno de los programas más utilizados en todo el mundo para la creación de páginas WEB. Es empleado tanto por profesionales como por personas que se inician en la creación de su primera pagina Web.

Dreamweaver es un programa muy sencillo e intuitivo, que puede ser utilizado a muchos niveles. Desde el más básico suficiente para crear una pagina personal de un usuario novel, hasta gestionar todo un "Site" empresarial con un contenido de cientos de páginas.

## **Rational Rose 2003.**

Es una herramienta software para el Modelado Visual mediante UML de sistemas software. Permite Especificar, Analizar, Diseñar el sistema antes de Codificarlo.

Tiene como Características:

- Mantiene la consistencia de los modelos del sistema software.
- Chequeo de la sintaxis UML.
- Generación Documentación automáticamente.

- Ingeniería Inversa (crear modelo a partir código).

En este capítulo se profundizó en los conceptos y definiciones necesarias para comprender la situación que llevó a la realización del presente trabajo. También se realizó una investigación de las tendencias, tecnologías, arquitectura, lenguajes, metodologías de desarrollo y herramientas a utilizar en todo el proceso de desarrollo, siempre siguiendo las políticas definidas por el Ministerio de Salud Pública (MINSAP).

## CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

En el capítulo se presentan las características del sistema. Se expone la situación problemática, el modelo de dominio el cual sirve de apoyo para la especificación de las condiciones, capacidades y cualidades que el sistema debe tener, los conceptos fundamentales de las entidades involucradas en el dominio y el diagrama de modelo de dominio del Registro de Ciudadanos 2.0. Se presentan los requisitos de software tanto los funcionales y no funcionales, así como el modelo de casos de uso del sistema, la definición de los actores, diagrama de caso de uso del sistema y la descripción textual de los casos de uso de ambos registros.

### 2.1 Situación Problemática y Problema a Resolver.

En la actualidad el RIS cuenta con un Registro de Ciudadanos (RC), concebido con capacidad de gestionar los datos de los ciudadanos cubanos, sean pacientes o personal de la salud, brindando información estándar a todos los componentes de SISalud y evitando la duplicidad de la misma.

En los inicios, al pensar en lo que sería este Registro, se omitieron aspectos importantes como la diferencia que existe entre los datos del carné de identidad de los ciudadanos cubanos militares y los ciudadanos cubanos civiles, además existen pacientes extranjeros en Cuba recibiendo los beneficios de la salud cubana y personal de la salud extranjeros residentes en el país brindando sus servicios. La información referente de estas personas, cubanos militares y extranjeros, no es recogida y ni gestionada, hasta estos momentos en el Registro de Ciudadanos, información que es de vital importancia para el buen funcionamiento de otros componentes. Por las razones anteriormente expuestas este Registro de Ciudadanos no responde en su totalidad a las exigencias actuales del Sistema de Información para la Salud (SISalud).

## 2.2 Modelo de Dominio.

Al no identificarse negocio en el Registro de Ciudadanos, se emplea un modelo de dominio. En el modelo de dominio se describen los conceptos fundamentales, permitiendo mostrar al cliente el entorno de la información que se maneja y de esta manera contribuir a la comprensión del contexto del sistema.

### 2.2.1 Conceptos Fundamentales.

**Ciudadanos:** Son los miembros de una comunidad, naturales o vecinos de una ciudad o lugar geográfico.

**Ciudadano Extranjero:** Toda persona que pertenece a una nación que no es la propia es decir que es o viene de otro país.

**Ciudadano Cubano:** Ciudadano natural de Cuba.

**Estado:** Es una característica que poseen los ciudadanos. Ejemplos de valores que puede tomar: muerto, residente en el extranjero, misión en el extranjero.

**Estado Civil:** Es la situación de las personas determinada por sus relaciones de familia, provenientes del matrimonio o del parentesco. Estados civiles más habituales: soltero, casado, unión consensual, divorciado y viudo.

### 2.2.2 Diagrama del Modelo de Dominio.

El Diagrama del Modelo de Dominio permite de forma gráfica ver la relación entre los objetos o clases, así como la cardinalidad de sus relaciones. A continuación se presenta el Diagrama del Modelo de Dominio del Registro de Ciudadanos 2.0.

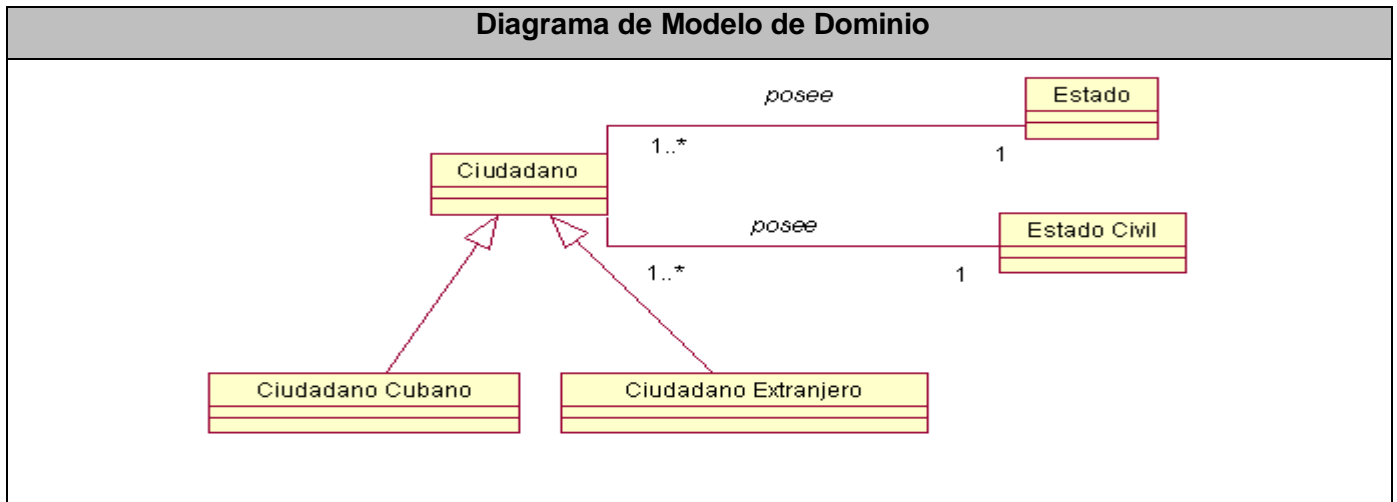


Figura 2.1 Diagrama de Modelo de Dominio.

## 2.3 Propuesta del Sistema.

Para dar solución al problema antes planteado, se decide implementar el Registro de Ciudadanos 2.0.

El Registro de Ciudadanos 2.0 gestiona información de los ciudadanos cubanos militares o civiles y de los ciudadanos Extranjeros.

El sistema permitirá la gestión y la realización de búsquedas eficientes de la información, así como la impresión de la misma. Brindarán servicios a las demás aplicaciones del Sistema de Información para la Salud. Además, contarán con un diseño de interfaces agradables y de fácil navegación, con ayuda y manual de usuario en línea.

### 2.3.1 Especificación de Requerimientos de Software.

Un Requerimiento de Software es la condición o capacidad que tiene que ser alcanzada o poseída por un sistema o componente de un sistema. En esta línea los requerimientos son el contrato que se debe cumplir, de modo que los usuarios finales tienen que comprender y aceptar los requerimientos que se especifiquen. Los Requerimientos de Software pueden ser divididos en dos grupos: requerimientos funcionales y requerimientos no funcionales.



### 2.3.1.1 Requerimientos Funcionales.

A continuación se muestra la totalidad de los Requerimientos Funcionales es decir, las capacidades o condiciones que el sistema debe cumplir, aunque es necesario señalar que se detallan todos los requerimientos porque no existe una documentación anterior como punto de partida para seleccionar los requerimientos que cubrirían el alcance de este trabajo, por lo que fue necesario realizar el proceso de ingeniería inversa para hacer la formalización de la totalidad de los requerimientos del Registro de Ciudadanos, realizando como alcance del trabajo actual, los requerimientos señalados.

**RF1:** Listar Ciudadanos que se correspondan con los criterios de búsquedas introducidos previamente (Carné de Identidad, Nombre, Sexo, Primer Apellido, Segundo Apellido y Estado).

**RF2:** Gestionar Información de los Ciudadanos Cubanos:

**RF2.2:** Insertar nuevo Ciudadano Cubano (Datos Generales: Carné de Identidad, Nombre, Apellidos, Sexo, Color de la piel y los ojos, Peso, Talla, Alias, Estado civil, Militar; Datos de Nacimiento: Provincia, Municipio y Fecha; Dirección, Nombre de los padres, Tomo y Folio).

**RF2.3:** Actualizar Ciudadano Cubano (Datos Generales: Carné de Identidad, Nombre, Apellidos, Sexo, Color de la piel y los ojos, Peso, Talla, Alias, Estado civil, Militar; Datos de Nacimiento: Provincia, Municipio y Fecha; Dirección, Nombre de los padres, Tomo y Folio).

**RF3:** Listar Extranjeros que se correspondan con los criterios de búsquedas introducidos previamente (Nombre, Sexo, Primer Apellido, Segundo Apellido, Estado, Número Pasaporte y País de procedencia).

**RF4:** Gestionar Información de los Ciudadanos Extranjeros:

**RF4.2** Insertar nuevo Ciudadano Extranjero (Datos Generales: Carné de Identidad, Nombre, Apellidos, Sexo, Color de la piel y los ojos, Peso, Talla, Alias, Estado civil, Número Pasaporte y Vence pasaporte; Datos de Nacimiento: Lugar, Nacionalidad y Fecha; Dirección en país de origen; Dirección en Cuba, Nombre de los padres).

**RF4.3** Actualizar Ciudadano Extranjero (Datos Generales: Carné de Identidad, Nombre, Apellidos, Sexo, Color de la piel y los ojos, Peso, Talla, Alias, Estado civil, Número Pasaporte y Vence pasaporte; Datos de Nacimiento: Lugar, Nacionalidad y Fecha; Dirección en país de origen; Dirección en Cuba, Nombre de los padre).

**RF5:** Imprimir información de los Ciudadanos Cubanos y Extranjeros.

### 2.3.1.2 Requerimientos No Funcionales.

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

#### **Usabilidad.**

**RNF1:** El sistema debe garantizar un acceso fácil y rápido, podrá ser usado por cualquier usuario que posea pocos conocimientos informáticos y de un ambiente Web en sentido general.

#### **Rendimiento.**

**RNF2:** El sistema debe tener una similitud en sus páginas y estar poco cargado, posibilitando que el sistema devuelva las respuestas de una manera eficiente, siendo más sencillo de entender y usar por el usuario.

#### **Soporte.**

**RNF3:** El personal que trabaja con el módulo debe contar con el nivel técnico requerido mediante adiestramiento de servicio.

#### **Portabilidad.**

**RNF4:** Permitir que el sistema se ejecute sobre el Sistema Operativo Linux, Windows 98 o superior.

#### **Seguridad.**

**RNF5:** Disponer de un mecanismo de seguridad basado en el modelo de Autenticación, Autorización y Auditoria (AAA).

**Confiabilidad:** La información manejada por el sistema está protegida de acceso no autorizado. El sistema debe prevenir posibles fallos y/o errores y presentar facilidades para una rápida recuperación en dichos casos.

**Integridad:** Que la información sea modificada (incluyendo su creación y borrado) sólo por personal autorizado.

**Disponibilidad:** Los usuarios autorizados tendrán acceso a la información en todo momento, se debe lograr balancear la carga de acceso entre múltiples servidores, disminuyendo los tiempos de respuesta.

### **Apariencia o Interfaz Externa.**

**RNF6:** La interfaz debe ser sencilla y amigable ya que el usuario no es experto en el uso de las aplicaciones Web.

### **Ayuda y Documentación en Línea.**

**RNF7:** Disponer de instrucciones en una opción de ayuda.

### **Software.**

**RNF8:** Los clientes tendrán acceso al registro a través de cualquier navegador Web. Recomendados Mozilla 1.5, Internet Explorer 5.0 o superior. El servidor debe tener PHP Versión 4.3.4 (aunque debe trabajar también con el 4.3.2), Biblioteca PEAR-SOAP 0.8RC3, Módulo XSLT (Sablotron) en PHP, Módulo DBX en PHP, Servidor de Base de Datos MySQL Versión 4 , Servidor HTTP (preferiblemente Apache) , Web Browser que soporte DHTML y CSS2.

### **Hardware.**

**RNF9:** Requerimientos mínimos:

- Ordenador Pentium o superior.
- 64 MB de Memoria RAM.
- Monitor VGA o superior.
- Teclado y Mouse.
- Procesador 486DX / 66 MHZ o superior.
- Disco duro de 20 GB.
- Impresora de puntos.
- Insumos. (Disquetes, CD RW, Papel continuo y cintas de impresora).
- La PC de trabajo debe estar conectada a una Red de Área Local (LAN).
- Conectividad con el nodo local de INFOMED.

### **2.3.2 Modelo de Casos de Uso del Sistema.**

Este artefacto se basa en la descripción de elementos o usuarios externos al sistema (actores) y de la funcionalidad del sistema (casos de uso). Un Modelo de Casos de Uso describe los requerimientos funcionales de un actor (usuario, sistema, dispositivo, etc.) en términos de las interacciones que éste ejecuta con el sistema. El modelado de casos de uso es una técnica efectiva y a la vez simple para modelar los requerimientos del sistema desde la perspectiva del usuario.

El modelo de casos de uso sirve como acuerdo entre clientes y desarrolladores para limitar las funciones con que dispondrá el sistema luego de ser implementado, además proporciona la entrada fundamental para el análisis, el diseño, la implementación y las pruebas.

Este modelo está formado por los diagramas de casos de uso y las narrativas de los casos de uso.

### 2.3.2.1 Definición de actores.

Un actor es como el rol o función que asume una persona, sistema o entidad que interactúa con el sistema que se está construyendo. A continuación se presentan los actores del sistema con su justificación, del Registro de Ciudadanos 2.0.

Actores del Sistema	Justificación
Funcionario Nacional (Editor).	Es el encargado de la gestión de toda la información del sistema, puede insertar, modificar todos los datos, además de hacer las búsquedas y listados.

Tabla 2.1 Definición de los actores del sistema.

### 2.3.2.2 Diagrama de Casos de Uso.

Un caso de uso es una secuencia de interacciones que se desarrollarán entre un sistema y sus actores en respuesta a un evento que inicia un actor principal sobre el propio sistema. Un Diagrama de Casos de Uso muestra la relación entre los actores y los casos de uso en un sistema. Una relación es una conexión entre los elementos del modelo.

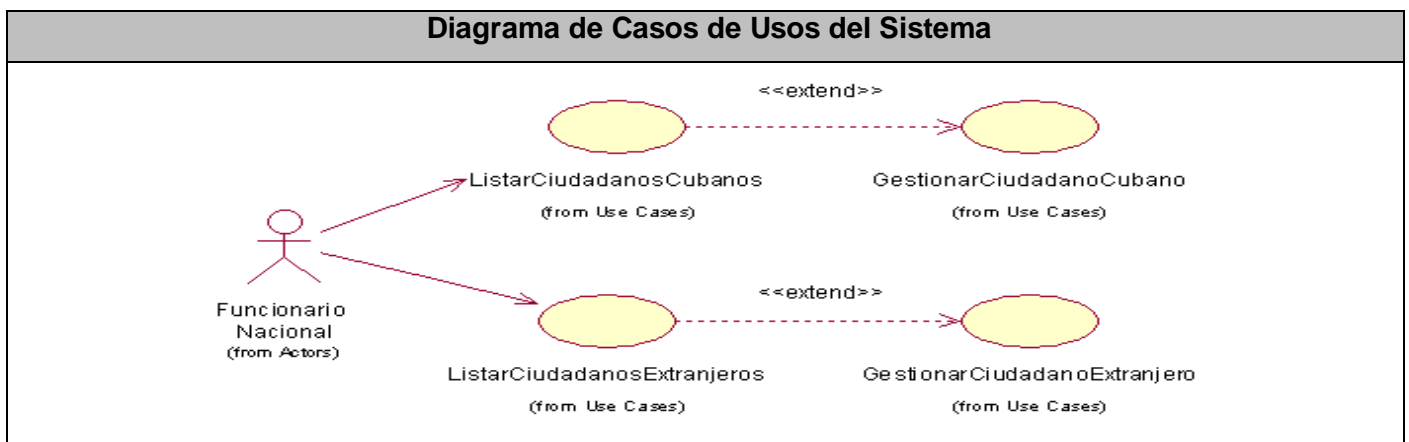


Figura 2.2 Diagrama de Casos de Usos del Sistema.

### 2.3.2.3 Descripción textual de los Casos de Uso del Sistema.

<b>Caso de Uso:</b>	ListarCiudadanosCubanos
<b>Actores:</b>	Funcionario Nacional
<b>Resumen:</b>	El Caso de Uso comienza cuando el usuario elige de las Opciones de Menú en etiqueta Ciudadanos “Listado” el submenú: cubano, el sistema hace una búsqueda (según los criterios introducidos) y muestra en la página el listado de los ciudadanos, devolviendo los resultados de 10 en 10. El usuario tiene la posibilidad de imprimir dicha información, culminando así el caso de uso.
<b>Precondiciones:</b>	El usuario debe haber sido autenticado en el Sistema.
<b>Referencias</b>	RF1, RF5
<b>Prioridad</b>	Crítica
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
1. Este caso de uso comienza una vez que el actor después de haber sido autenticado escoge de las Opciones de Menú, en la etiqueta “Ciudadanos”, la opción “Listado” el submenú: Cubanos.	1.1 El sistema muestra una interfaz con las opciones de búsqueda por CI, Nombre, Sexo, Primer Apellido, Segundo Apellido, Estado, puede hacer una búsqueda general con el botón “Mostrar Todos”.
2. El actor realiza la búsqueda por los criterios que desee (llenando los datos que aparecen en la interfaz y luego “Buscar” o hace una búsqueda general con “Mostrar Todos”).	2.1 El sistema muestra una lista del resultado de la búsqueda en la Base de Datos, según el criterio seleccionado. La información resultante de la búsqueda se muestra en forma de un listado con los datos que seleccione. Si son más de 10 los resultados encontrados, los muestra de 10 en 10 y brinda la posibilidad al actor de ver los 10 resultados próximos mediante el botón “Próximo”, los anteriores mediante el botón “Anterior”, los últimos mediante el botón “Último”,

## CARACTERÍSTICAS DEL SISTEMA

	<p>los primeros mediante el botón “Primero” y también escogerlos directamente por el número de la página donde estén por el botón “Página #”.</p> <p>2.2 El sistema brinda la posibilidad de imprimir dicha información mediante un botón llamado “Versión para Imprimir”, ver la sección Imprimir.</p>
<b>Sección Imprimir</b>	
3. Si el actor desea imprimir la información mostrada da clic en el botón “Versión para Imprimir”.	<p>3.1 El sistema muestra dos opciones de impresión:</p> <ul style="list-style-type: none"> <li>• Generada por Adobe Acrobat (pdf)</li> <li>• Generada Microsoft Excel (xls).</li> </ul>
4. El actor escoge una de las dos opciones.	<p>4.1 Si el actor escogió imprimir en forma de pdf, el sistema hace una consulta a la Base de Datos y genera (crea, guarda una copia en el servidor y muestra) un pdf y si escogió imprimir en forma de xls el sistema genera de la misma forma un excel.</p>
<b>Flujos Alternos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Negocio</b>
<b>Poscondiciones</b>	El sistema retorna los datos de los Ciudadanos Cubanos.

Tabla 2.2 Descripción del caso de uso: ListarCiudadanosCubanos.

<b>Caso de Uso:</b>	GestionarCiudadanoCubano
<b>Actores:</b>	Funcionario Nacional
<b>Resumen:</b>	El Caso de Uso comienza cuando el actor elige de las Opciones de Menú, en la etiqueta Ciudadanos alguna de las opciones que muestra la interfaz, siempre seleccionando en el submenú la opción de cubanos. Al seleccionarla el sistema

## CARACTERÍSTICAS DEL SISTEMA

	<p>muestra la interfaz correspondiente con opciones para buscar ciudadanos cubanos, el usuario realiza las búsquedas por el criterio que introdujo y se muestra un listado de los ciudadanos que resultaron, cada cual con un vínculo a sus datos particulares los cuales pueden ser modificados, además simultáneo con el listado también se muestran la opción de Agregar, si escoge Agregar el sistema muestra la interfaz con la posibilidad de insertar los nuevos datos, el actor oprime el botón para esta acción y el sistema actualiza la Base de Datos. Si el usuario lo que desea es modificar los datos de un ciudadano consulta los datos del mismo mediante un hipervínculo que posee, el sistema le muestra un formulario con los datos a modificar, este hace las modificaciones que desea y se actualiza la Base de Datos, de esta forma se culmina el Caso de Uso.</p>
<b>Precondiciones:</b>	El usuario debe haber sido autenticado en el Sistema.
<b>Referencias</b>	RF1, RF2
<b>Prioridad</b>	Critica
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Negocio</b>
1. Este caso de uso comienza una vez que el actor después de haber sido autenticado selecciona en Opciones de Menú en la etiqueta “Ciudadanos” de las opciones que ya muestra la interfaz del sistema, puede ser Listado o Nuevo.	1.1 Si escoge la opción “Listado” el submenú: Cubanos ir a la Sección Listar Cubanos y si escoge “Nuevo” el submenú: Cubanos ir a la Sección Nuevo Cubano.
<b>Sección Nuevo Cubano</b>	
	1.1 El sistema muestra la interfaz para Agregar Ciudadanos Cubanos, la cual tiene un formulario con los campos a llenar.
2. El actor llena los campos y presiona el botón “Aceptar”.	2.1 EL sistema busca en la base datos si existen ciudadanos con igual carné de identidad, en caso de existir ir a la Sección Duplicar, en caso de no existir coincidencia el sistema actualiza la Base de Datos.



## CARACTERÍSTICAS DEL SISTEMA

<b>Sección Listar Cubanos</b>	
	1.1 El sistema muestra una interfaz con las opciones de búsqueda por CI, Nombre, Sexo, Primer Apellido, Segundo Apellido, Estado, puede hacer una búsqueda general con el botón "Mostrar Todos".
3. El actor realiza la búsqueda por los criterios que desee (llenando los datos que aparecen en la interfaz y luego "Buscar" o hace una búsqueda general con "Mostrar Todos").	3.1 El sistema muestra una lista del resultado de la búsqueda en la Base de Datos, según el criterio seleccionado. La información resultante de la búsqueda se muestra en forma de un listado con los datos que seleccione. Si son más de 10 los resultados encontrados, los muestra de 10 en 10 y brinda la posibilidad al actor de ver los 10 resultados próximos mediante el botón "Próximo", los anteriores mediante el botón "Anterior", los últimos mediante el botón "Último", los primeros mediante el botón "Primero" y también escogerlos directamente por el número de la página donde estén por el botón "Página #". 3.2 El Sistema brinda opciones de Editar siempre seleccionando un Ciudadano cubano determinado, ver la sección Modificar, de Agregar un ciudadano nuevo, ver la sección Nuevo Cubano.
<b>Sección Modificar</b>	
	3.3 El sistema muestra todos los campos que el usuario puede actualizar.
4. El usuario presiona sobre el carné de identidad del ciudadano que quiere modificar.	4.1 EL sistema busca en la base datos si existen ciudadanos con igual carné de identidad, en caso de existir ir a la Sección Duplicar, en caso de no

## CARACTERÍSTICAS DEL SISTEMA

	existir coincidencia el sistema actualiza la Base de Datos.
<b>Sección Duplicar</b>	
	2.1 El sistema muestra la interfaz con las opciones: Editar u Aceptar (Duplicar).
5. El usuario escoge una de las opciones.	5.1 Si escoge la opción Editar siempre seleccionando un Ciudadano determinado, ver la sección Modificar, Si desea duplicarlo (opción: Aceptar) el sistema actualiza la Base de Datos.
<b>Flujos Alternos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Negocio</b>
<b>Poscondiciones</b>	El sistema retorna los datos de los Ciudadanos Cubanos.

Tabla 2.3 Descripción del caso de uso: GestionarCiudadanoCubano.

<b>Caso de Uso:</b>	ListarCiudadanosExtranjeros
<b>Actores:</b>	Funcionario Nacional
<b>Resumen:</b>	El Caso de Uso comienza cuando el usuario elige de las Opciones de Menú en etiqueta Ciudadanos “Listado” el submenú: Extranjeros, el sistema hace una búsqueda (según los criterios introducidos) y muestra en la página el listado de los ciudadanos, devolviendo los resultados de 10 en 10. El usuario tiene la posibilidad de imprimir dicha información, culminando así el caso de uso.
<b>Precondiciones:</b>	El usuario debe haber sido autenticado en el Sistema.
<b>Referencias</b>	RF3, RF5
<b>Prioridad</b>	Crítica
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Negocio</b>

## CARACTERÍSTICAS DEL SISTEMA

<p>1. Este caso de uso comienza una vez que el actor después de haber sido autenticado escoge de las Opciones de Menú, en la etiqueta “Ciudadanos”, la opción “Listado” el submenú: Extranjeros.</p>	<p>1.1 El sistema muestra una interfaz con las opciones de búsqueda por CI, Nombre, Sexo, Primer Apellido, Segundo Apellido, Estado, Numero Pasaporte y País de procedencia, puede hacer una búsqueda general con el botón “Mostrar Todos”.</p>
<p>2. El actor realiza la búsqueda por los criterios que desee (llenando los datos que aparecen en la interfaz y luego “Buscar” o hace una búsqueda general con “Mostrar Todos”).</p>	<p>2.1 El sistema muestra una lista del resultado de la búsqueda en la Base de Datos, según el criterio seleccionado. La información resultante de la búsqueda se muestra en forma de un listado con los datos que seleccione. Si son más de 10 los resultados encontrados, los muestra de 10 en 10 y brinda la posibilidad al actor de ver los 10 resultados próximos mediante el botón “Próximo”, los anteriores mediante el botón “Anterior”, los últimos mediante el botón “Último”, los primeros mediante el botón “Primero” y también escogerlos directamente por el número de la página donde estén por el botón “Página #”.</p> <p>2.2 El sistema brinda la posibilidad de imprimir dicha información mediante un botón llamado “Versión para Imprimir”, ver la sección Imprimir.</p>
<p><b>Sección Imprimir</b></p>	
<p>3. Si el actor desea imprimir la información mostrada da clic en el botón “Versión para Imprimir”.</p>	<p>3.1 El sistema muestra dos opciones de impresión:</p> <ul style="list-style-type: none"> <li>• Generada por Adobe Acrobat (pdf)</li> <li>• Generada Microsoft Excel (xls).</li> </ul>
<p>4. El actor escoge una de las dos opciones.</p>	<p>4.1 Si el actor escogió imprimir en forma de pdf, el sistema hace una consulta a la Base de Datos y genera (crea, guarda una copia en el servidor y</p>

## CARACTERÍSTICAS DEL SISTEMA

	muestra) un pdf y si escogió imprimir en forma de xls el sistema genera de la misma forma un excel.
<b>Flujos Alternos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Negocio</b>
<b>Poscondiciones</b>	El sistema retorna los datos de los Ciudadanos Extranjeros.

Tabla 2.4 Descripción del caso de uso: ListarCiudadanosExtranjeros.

<b>.Caso de Uso:</b>	GestionarCiudadanoExtranjero
<b>Actores:</b>	Funcionario Nacional
<b>Resumen:</b>	El Caso de Uso comienza cuando el actor elige de las Opciones de Menú, en la etiqueta Ciudadanos alguna de las opciones que muestra la interfaz, siempre seleccionando en el submenú la opción de Extranjeros. Al seleccionarla el sistema muestra la interfaz correspondiente con opciones para buscar ciudadanos extranjeros, el usuario realiza las búsquedas por el criterio que introdujo y se muestra un listado de los ciudadanos que resultaron, cada cual con un vínculo a sus datos particulares los cuales pueden ser modificados, además simultáneo con el listado también se muestran la opción de Agregar, si escoge Agregar el sistema muestra la interfaz con la posibilidad de insertar los nuevos datos, el actor oprime el botón para esta acción y el sistema actualiza la Base de Datos. Si el usuario lo que desea es modificar los datos de un ciudadano consulta los datos del mismo mediante un hipervínculo que posee, el sistema le muestra un formulario con los datos a modificar, este hace las modificaciones que desea y se actualiza la Base de Datos, de esta forma se culmina el Caso de Uso.
<b>Precondiciones:</b>	El usuario debe haber sido autenticado en el Sistema.
<b>Referencias</b>	RF3, RF4
<b>Prioridad</b>	Crítica

<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Negocio</b>
1. Este caso de uso comienza una vez que el actor después de haber sido autenticado selecciona en Opciones de Menú en la etiqueta “Ciudadanos” de las opciones que ya muestra la interfaz del sistema, puede ser Listado o Nuevo.	1.1 Si escoge la opción “Listado” el submenú: Extranjeros ir a la Sección Listar Extranjeros y si escoge “Nuevo” el submenú: Extranjeros ir a la Sección Nuevo Extranjero.
<b>Sección Nuevo Extranjero</b>	
	1.1 El sistema muestra la interfaz para Agregar Ciudadanos Extranjeros, la cual tiene un formulario con los campos a llenar.
2. El actor llena los campos y presiona el botón “Aceptar”.	2.1 EL sistema busca en la base datos si existen ciudadanos extranjeros con igual carné de identidad, en caso de existir ir a la Sección Duplicar, en caso de no existir coincidencia el sistema actualiza la Base de Datos.
<b>Sección Listar Extranjeros</b>	
	1.1 El sistema muestra una interfaz con las opciones de búsqueda por CI, Nombre, Sexo, Primer Apellido, Segundo Apellido, Estado, Numero Pasaporte y País de procedencia, puede hacer una búsqueda general con el botón “Mostrar Todos”.
3. El actor realiza la búsqueda por los criterios que desee (llenando los datos que aparecen en la interfaz y luego “Buscar” o hace una búsqueda general con “Mostrar Todos”).	3.1 El sistema muestra una lista del resultado de la búsqueda en la Base de Datos, según el criterio seleccionado. La información resultante de la búsqueda se muestra en forma de un listado con los datos que seleccione. Si son más de 10 los resultados encontrados, los muestra de 10 en 10 y brinda la posibilidad al actor de ver los 10 resultados próximos mediante el botón

## CARACTERÍSTICAS DEL SISTEMA

	<p>“Próximo”, los anteriores mediante el botón “Anterior”, los últimos mediante el botón “Último”, los primeros mediante el botón “Primero” y también escogerlos directamente por el número de la página donde estén por el botón “Página #”.</p> <p>3.2 El Sistema brinda opciones de Editar siempre seleccionando un Ciudadano extranjero determinado, ver la sección Modificar, de Agregar un ciudadano nuevo, ver la sección Nuevo Extranjero.</p>
<b>Sección Modificar</b>	
	3.3 El sistema muestra todos los campos que el usuario puede actualizar.
4. El usuario presiona sobre el carné de identidad del ciudadano que quiere modificar.	4.1 EL sistema busca en la base datos si existen ciudadanos con igual carné de identidad, en caso de existir ir a la Sección Duplicar, en caso de no existir coincidencia el sistema actualiza la Base de Datos.
<b>Sección Duplicar</b>	
	2.1 El sistema muestra la interfaz con las opciones: Editar u Aceptar (Duplicar).
5. El usuario escoge una de las opciones.	5.1 Si escoge la opción Editar siempre seleccionando un Ciudadano extranjero determinado, ver la sección Modificar, Si desea duplicarlo (opción: Aceptar) el sistema actualiza la Base de Datos.
<b>Flujos Alternos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Negocio</b>
<b>Poscondiciones</b>	El sistema retorna los datos de los Ciudadanos Extranjeros.

Tabla 2.5 Descripción del caso de uso: GestionarCiudadanoExtranjero.

En este capítulo, debido a la poca estructuración de los procesos del negocio y para una mejor comprensión, se definieron conceptos que fueron relacionados mediante un Modelo de Dominio. Además se definieron los requisitos que debe cumplir el sistema, se seleccionaron los actores y los casos de uso. El empleo de los modelos de casos de uso para describir el sistema propuesto permitió una adecuada captación y modelación de los requerimientos, demostrando su importancia en esta etapa.

## CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

El diseño del sistema es la estrategia de alto nivel para resolver problemas y construir una solución. Éste incluye decisiones acerca de la organización del sistema en subsistemas, la asignación de subsistemas a componentes hardware y software, y decisiones fundamentales conceptuales y de política que son las que constituyen un marco de trabajo para el diseño detallado. La organización global del sistema es lo que se denomina la arquitectura del sistema.

En el presente capítulo se realiza el diseño de la solución propuesta. Se realiza la justificación del uso de patrones y subsistemas de servicios. Se presentan los diagramas de clases de diseño y se describen las clases y atributos. Además se presentan los diagramas de clases persistentes, los modelos de datos y las descripciones de las tablas y atributos del Registro de Ciudadanos 2.0.

### 3.1 Modelo de Análisis.

El Modelo de Análisis es un modelo que se utiliza para obtener una visión del sistema sobre los Requisitos Funcionales, expresados en lenguaje técnico.

#### 3.1.2 Diagramas de Clases de Análisis.

El diagrama de clase es el diagrama principal de diseño y análisis para un sistema. En él, la estructura de las clases del sistema se especifica, con relaciones entre clases y estructuras de herencia. Durante el análisis del sistema, el diagrama se desarrolla buscando una solución ideal. A continuación se presentan algunos de los diagramas de clases de análisis del Registro de Ciudadanos 2.0:

**Modelo de Clases de Análisis**



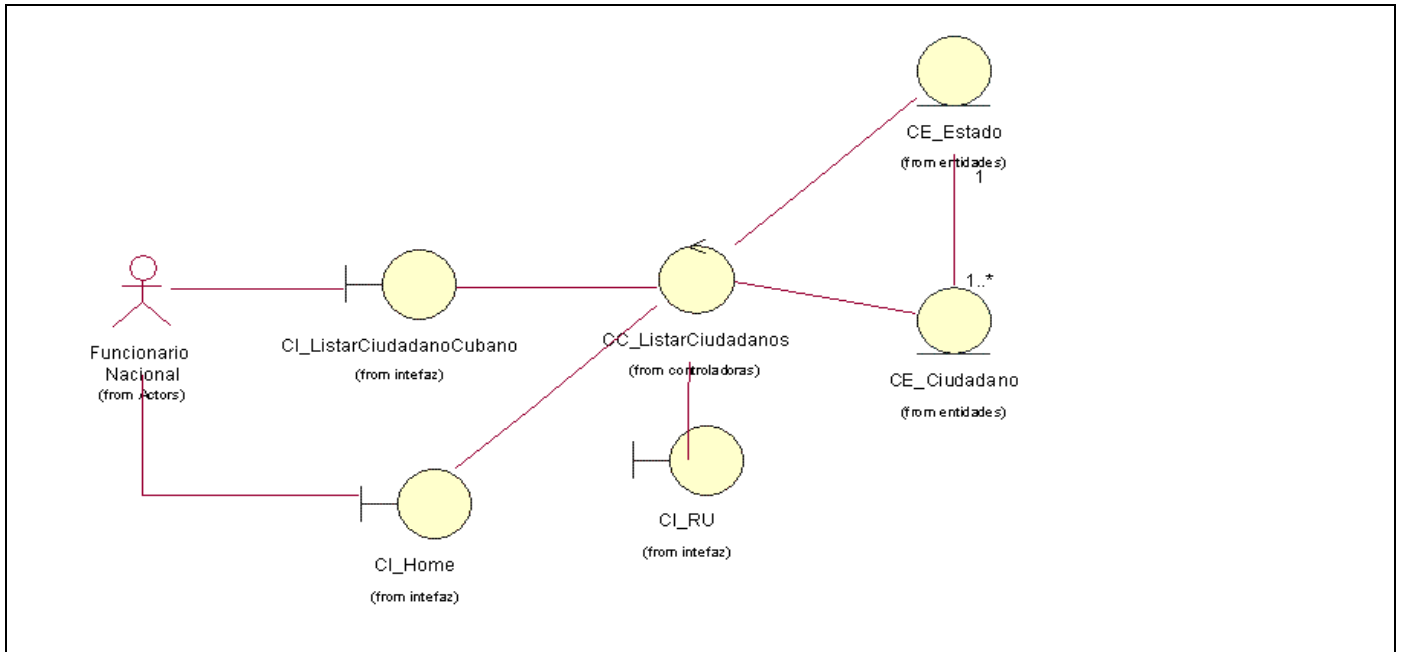


Figura 3.1 Modelo de Clases de Análisis Caso de uso: ListarCiudadanosCubanos.

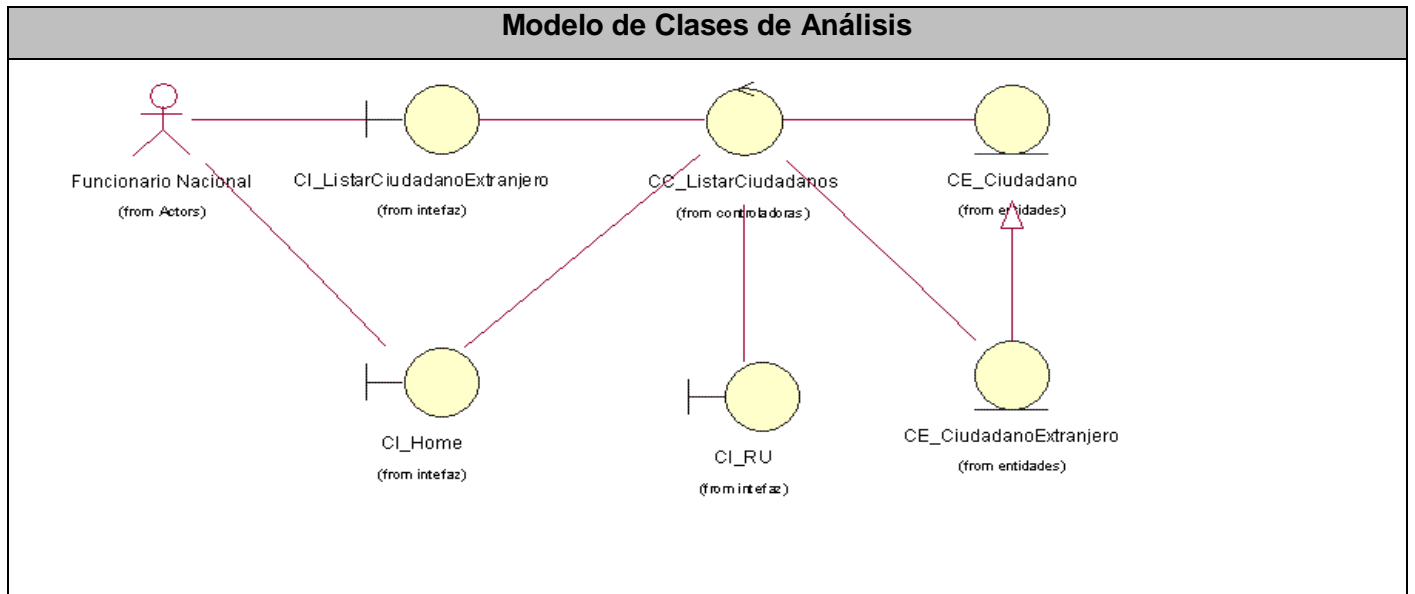


Figura 3.2 Modelo de Clases de Análisis Caso de uso: ListarCiudadanosExtranjeros.

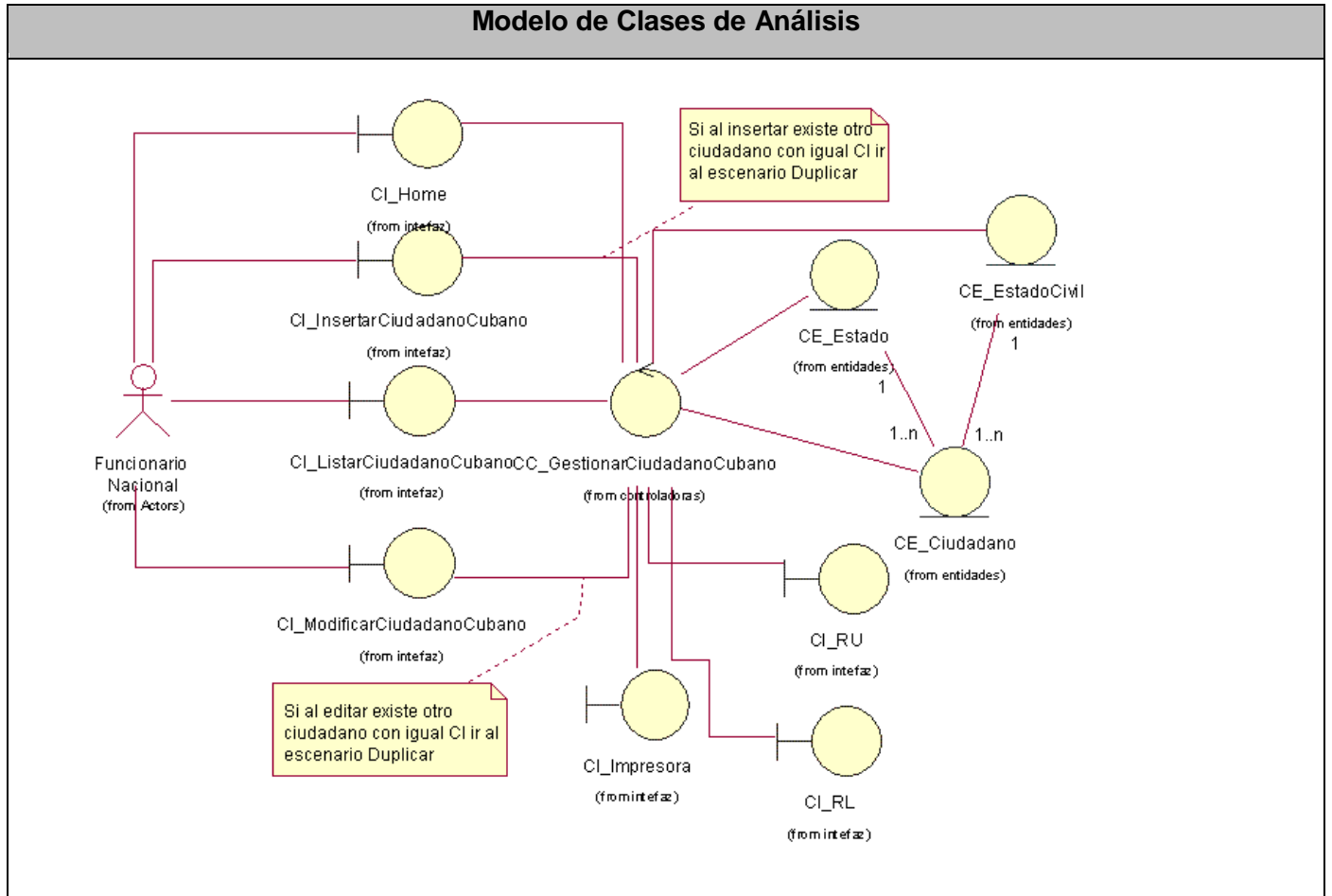


Figura 3.3 Modelo de Clases de Análisis Caso de uso: GestionarCiudadanoCubano.

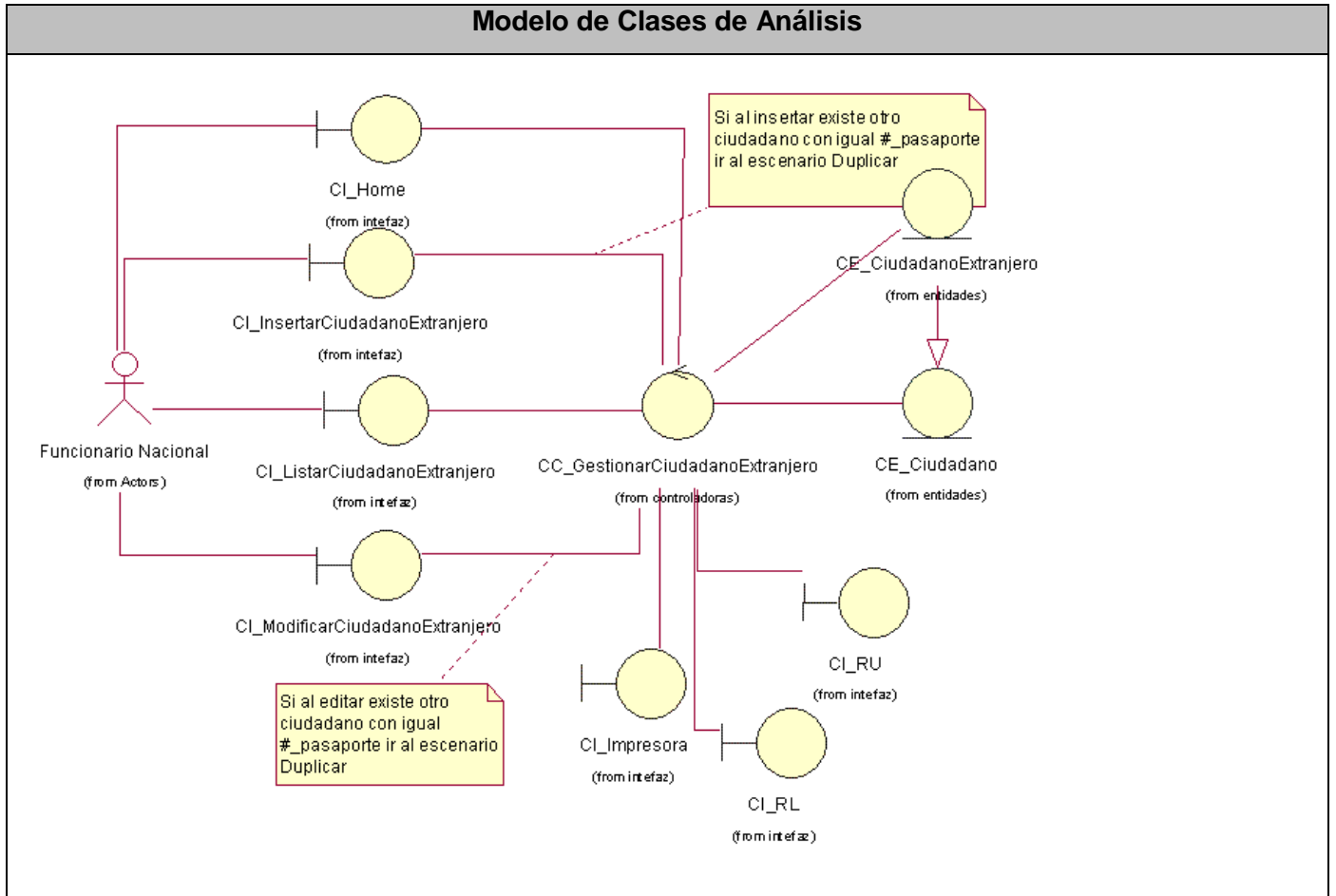


Figura 3.4 Modelo de Clases de Análisis Caso de uso: GestionarCiudadanoExtranjero.

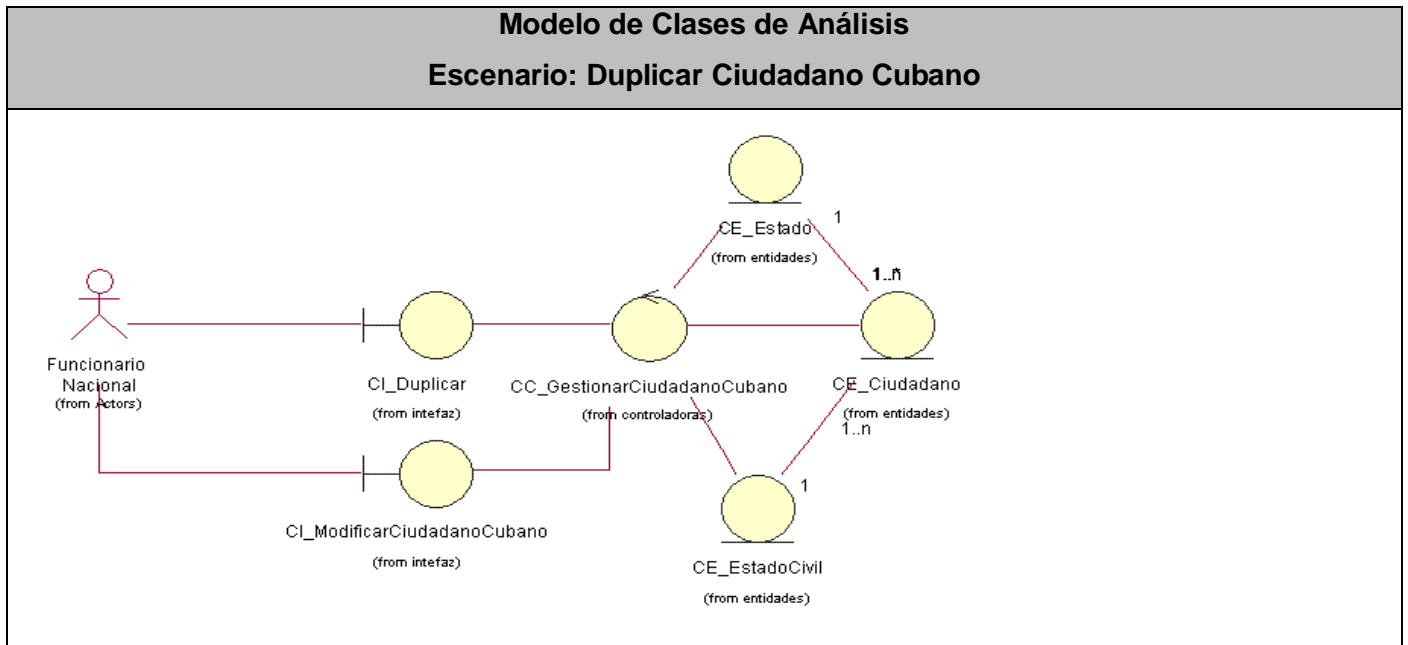


Figura 3.5 Modelo de Clases de Análisis Escenario: Duplicar Ciudadano Cubano.

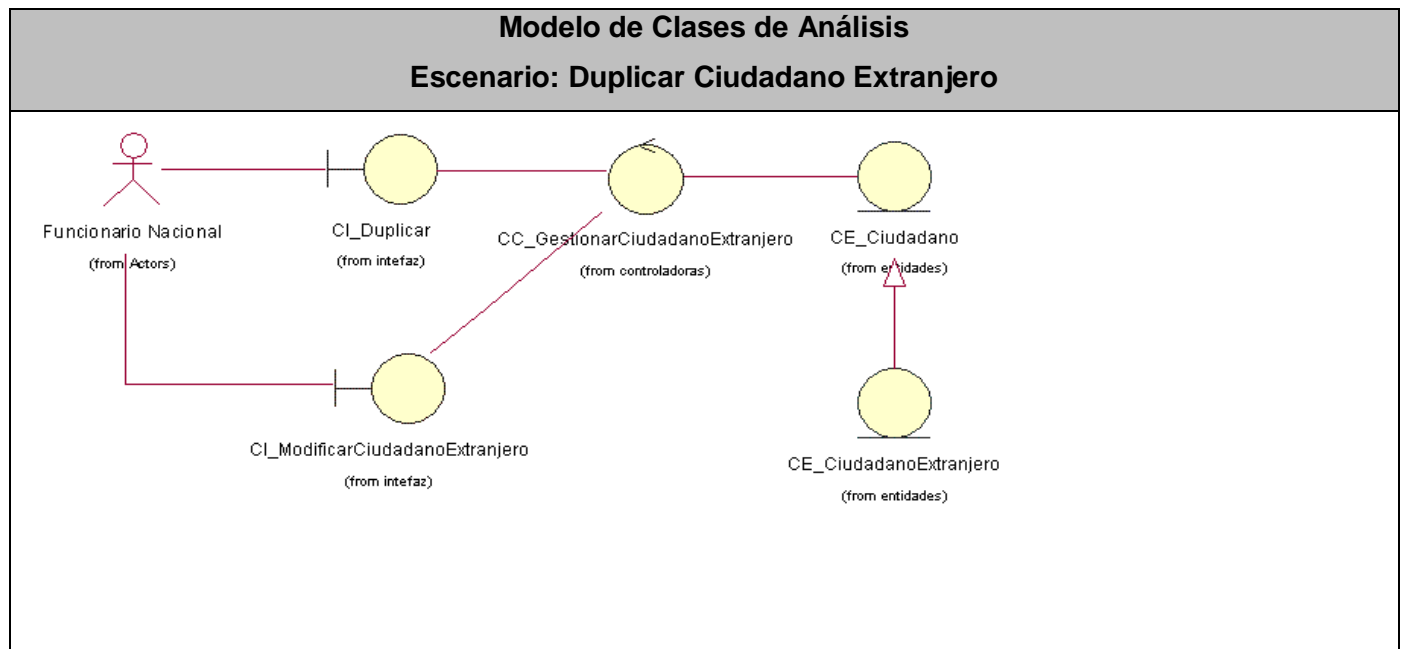


Figura 3.6 Modelo de Clases de Análisis Escenario: Duplicar Ciudadano Extranjero.

## 3.1.3 Diagramas de Interacción.

Los Diagramas de Interacción se utilizan para modelar los aspectos dinámicos de un sistema, lo que conlleva modelar instancias concretas o prototípicas de clases interfaces, componentes y nodos, junto con los mensajes enviados entre ellos. Se dividen en diagramas de Secuencia y diagramas de Colaboración.

En el presente trabajo se definió realizar diagramas de Colaboración. Muestra la interacción entre varios objetos y los enlaces que existen entre ellos. Representa las interacciones entre objetos organizadas alrededor de los objetos y sus vinculaciones. A diferencia de un diagrama de Secuencias, un diagrama de Colaboraciones muestra las relaciones entre los objetos, no la secuencia en el tiempo en que se producen los mensajes. Los diagramas de Secuencias y los diagramas de Colaboraciones expresan información similar, pero en una forma diferente.

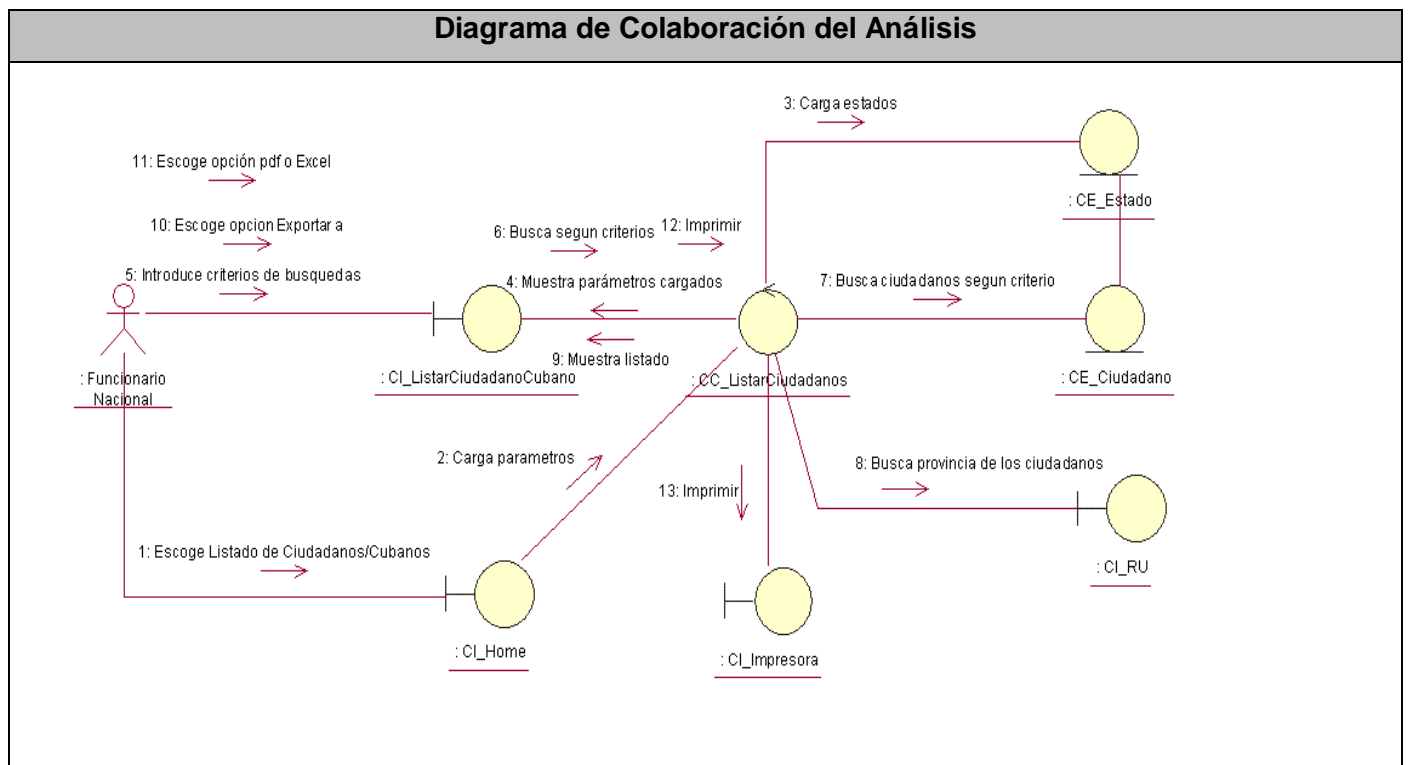


Figura 3.7 Diagrama de Colaboración del Análisis Caso de Uso: ListarCiudadanosCubanos.

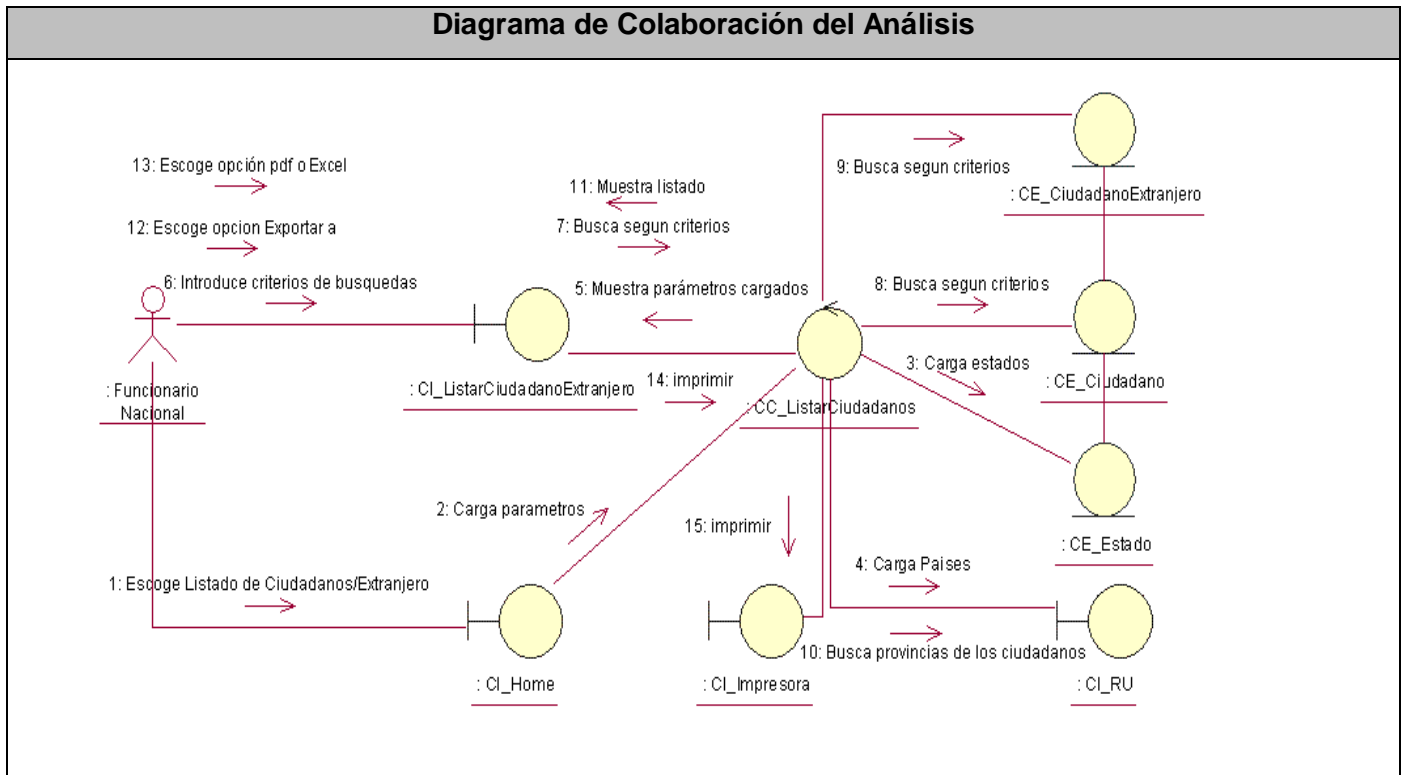


Figura 3.8 Diagrama de Colaboración del Análisis Caso de Uso: ListarCiudadanosExtranjeros.

## Diagrama de Colaboración del Análisis

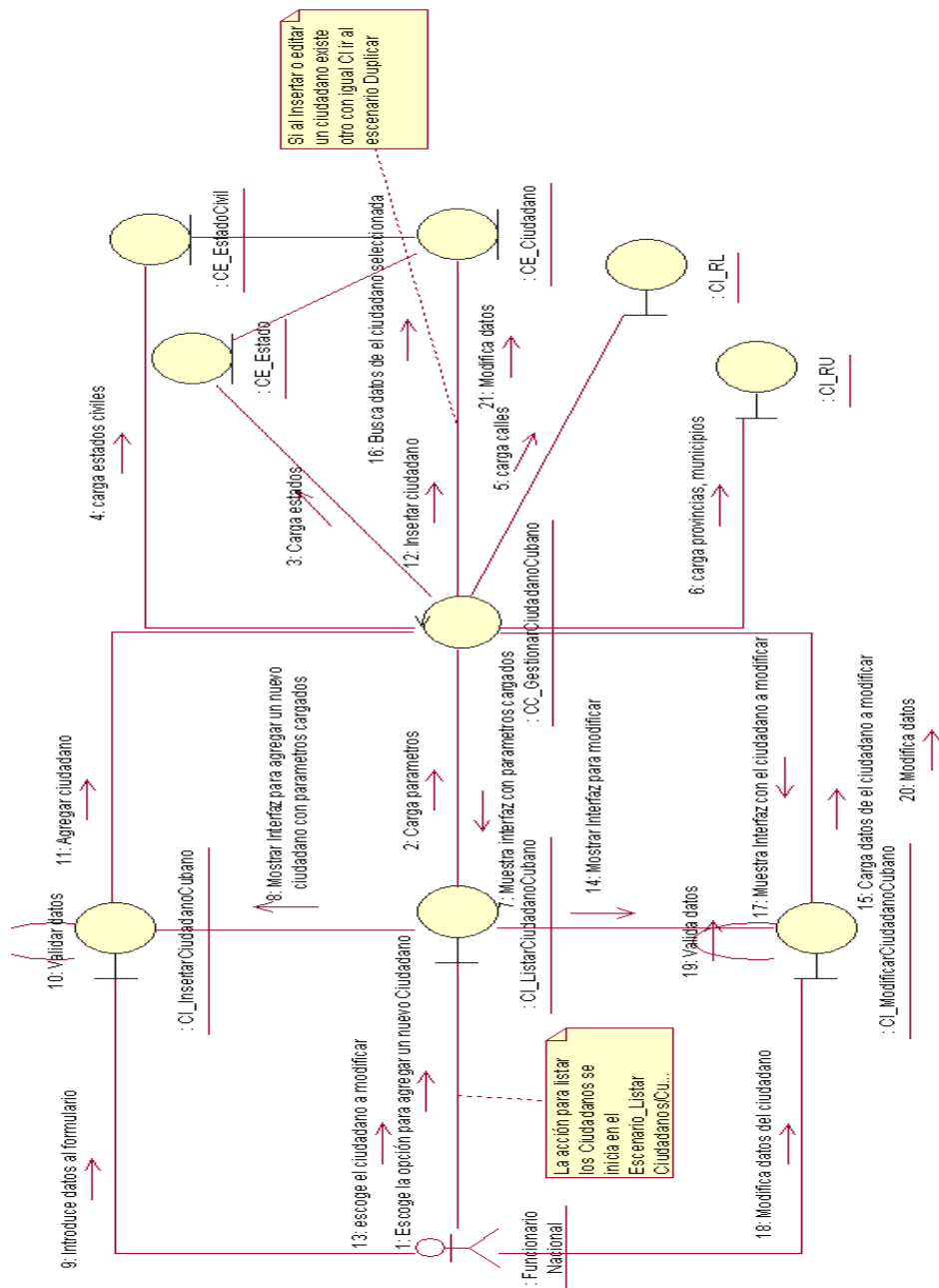


Figura 3.9 Diagrama de Colaboración del Análisis Caso de Uso: GestionarCiudadanoCubano.

Diagrama de Colaboración del Análisis

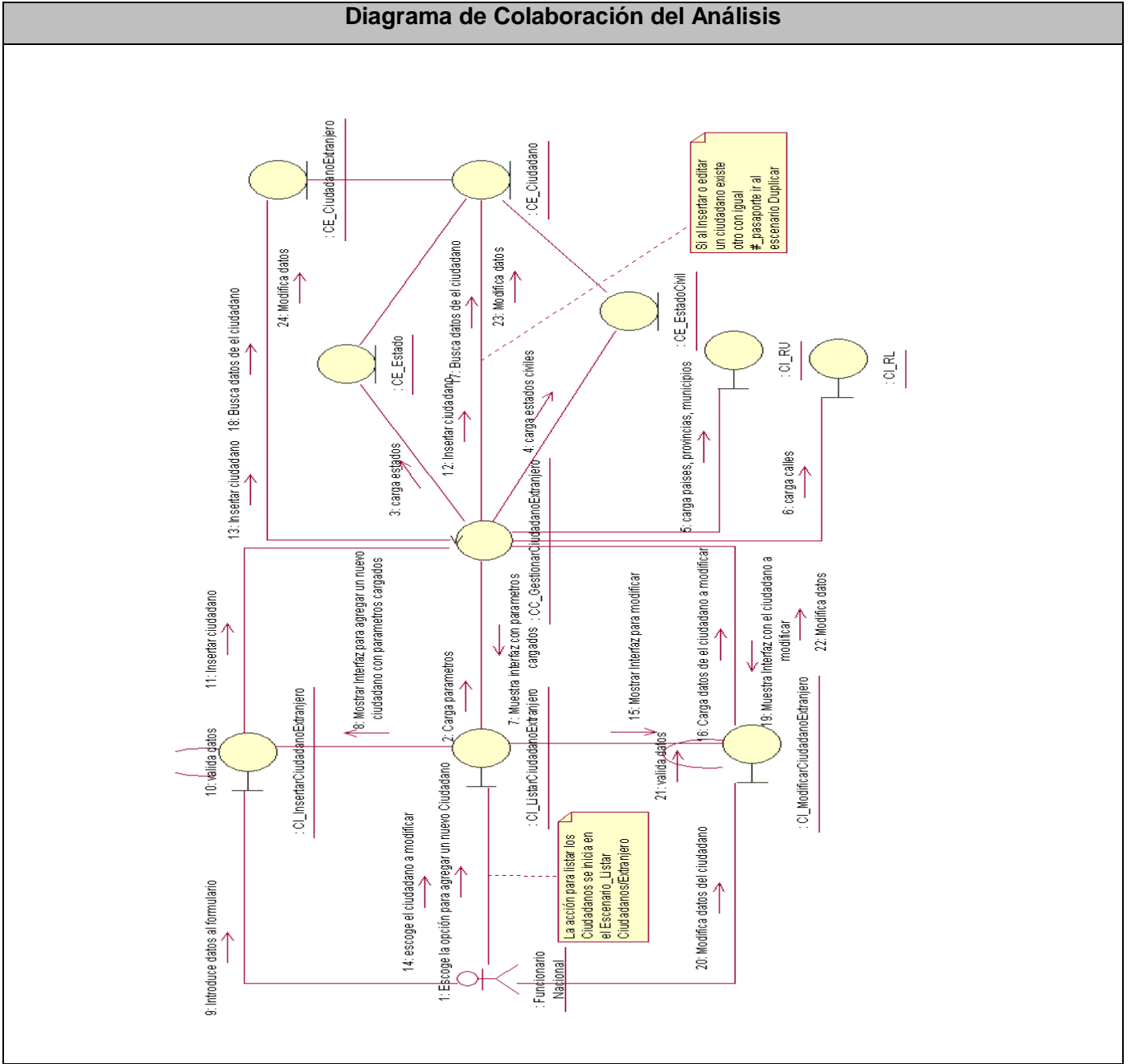


Figura 3.10 Diagrama de Colaboración del Análisis Caso de Uso: GestionarCiudadanoExtranjero.



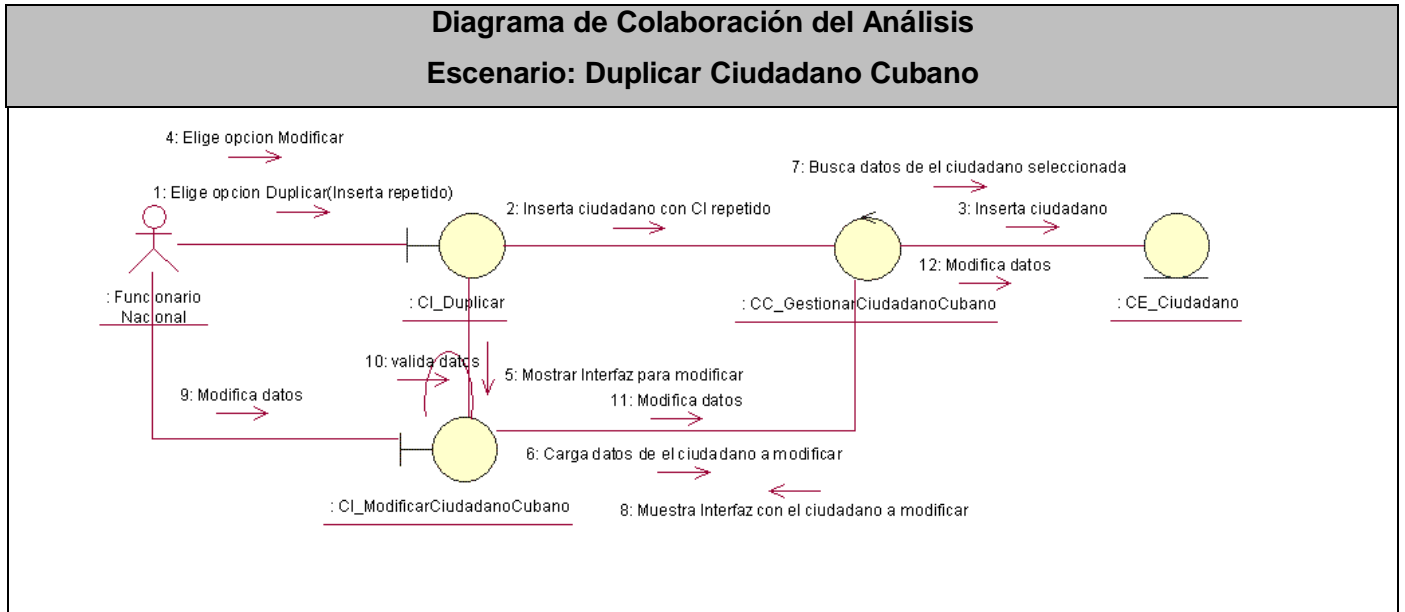


Figura 3.11 Diagrama de Colaboración del Análisis Escenario: Duplicar Ciudadano Cubano.

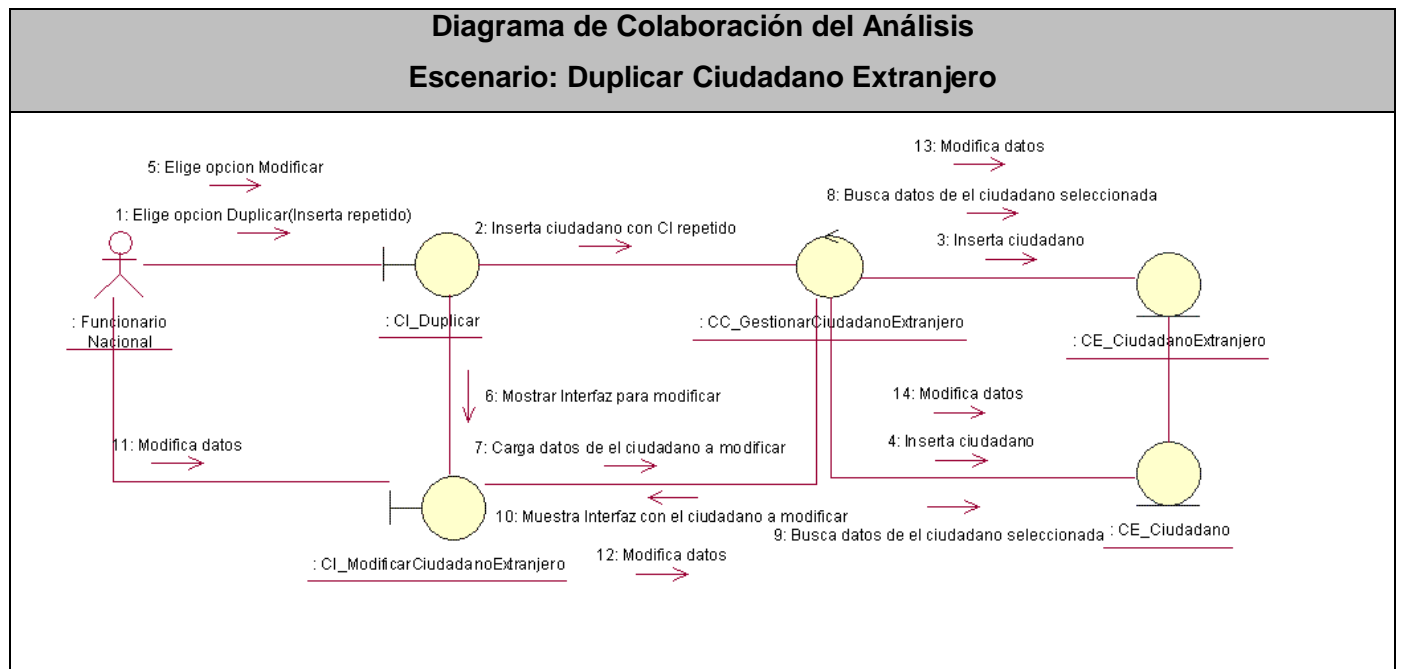


Figura 3.12 Diagrama de Colaboración del Análisis Escenario: Duplicar Ciudadano Extranjero.

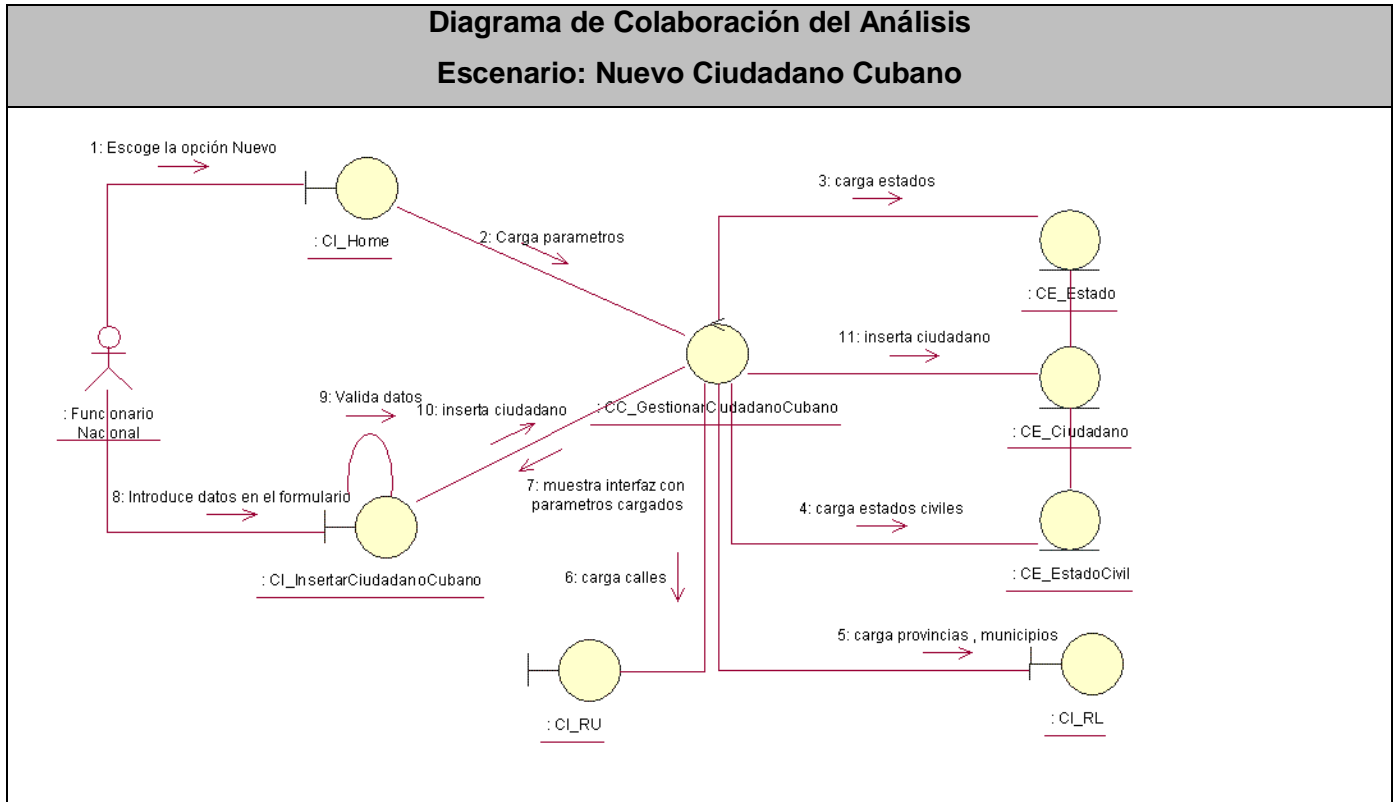


Figura 3.13 Diagrama de Colaboración del Análisis Escenario: Nuevo Ciudadano Cubano.

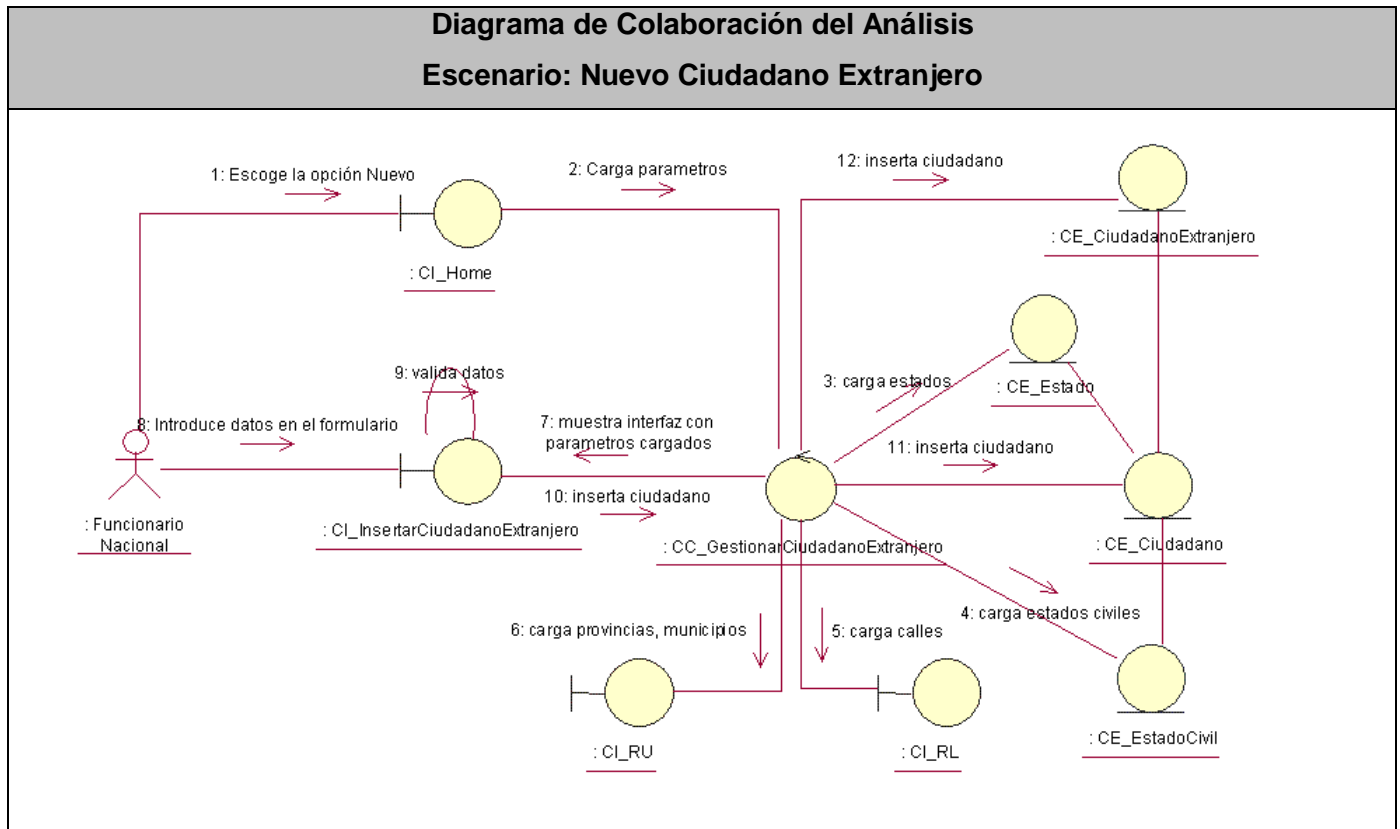


Figura 3.14 Diagrama de Colaboración del Análisis Escenario: Nuevo Ciudadano Extranjero.

### 3.2 Modelo de Diseño.

El Modelo de Diseño permite adquirir una comprensión de los aspectos relacionados con los requisitos no funcionales y restricciones relacionadas con los lenguajes de programación, componentes reutilizables, sistemas operativos, tecnologías de distribución y concurrencia y tecnologías de interfaz de usuario.

Descompone los trabajos de implementación en partes más manejables que puedan ser llevadas a cabo por diferentes equipos de desarrollo. Captura las interfaces entre los subsistemas en el ciclo de vida del software, lo cual es muy útil cuando se utilizan interfaces como elementos de sincronización entre diferentes equipos de desarrollo. Los artefactos generados en el Modelo de Diseño son: Modelo de Despliegue, Descripción de la Arquitectura, Realización de Casos de Uso, Clase del Diseño, Subsistema de Diseño e Interfaz.

## 3.2.1 Justificación del uso de Patrones.

Un patrón es un modelo que se puede seguir para realizar algo. Los patrones surgen de la experiencia de seres humanos de tratar lograr ciertos objetivos promoviendo buenas prácticas.

Los Patrones de Diseño son la base para la búsqueda de soluciones a problemas comunes en el desarrollo de software. Un patrón de diseño es una solución a un problema de diseño, además, estos pretenden proporcionar catálogos de elementos reusables en el diseño de sistemas software, formalizar un vocabulario común entre diseñadores, estandarizar el modo en que se realiza el diseño, facilitar el aprendizaje de las nuevas generaciones de diseñadores condensando conocimiento ya existente.

En el sistema desarrollado se utilizan patrones como: Proxy, Alta Cohesión y Bajo Acoplamiento, los cuales jugaron determinadas funciones que se explican a continuación:

### **Patrón Proxy.**

El patrón Proxy trabaja como un concentrador de otro objeto para controlar el acceso a éste. Frecuentemente este patrón se usa cuando se tienen objetos que consumen muchos recursos o memoria y se quiere instanciar dichos objetos solo en el momento que sean pedidos por el cliente, aunque también se puede usar en otros casos.

Es un patrón muy general que obliga a que las llamadas a métodos de un objeto ocurran indirectamente a través de un objeto proxy, que actúa como sustituto o intermediario del objeto original, delegando luego las llamadas a los métodos de los objetos respectivos.

### **Patrón Alta Cohesión.**

La cohesión es una medida de cuán relacionadas y enfocadas están las responsabilidades de una clase. Una alta cohesión caracteriza a las clases con responsabilidades estrechamente relacionadas que no realicen un trabajo enorme.

El patrón Alta Cohesión mejora la claridad y facilidad con que se entiende el diseño. Se simplifica el mantenimiento y las mejoras de funcionalidad. A menudo se genera un bajo acoplamiento. Soporta mayor capacidad de reutilización.

## **Patrón Bajo Acoplamiento.**

El acoplamiento es una medida de la fuerza con que una clase está conectada a otras clases, con que las conoce y con que recurre a ellas. El acoplamiento bajo significa que una clase no depende de muchas clases. Mientras que acoplamiento alto significa que una clase recurre a muchas otras Clases. Con el empleo de este patrón no se afectan por cambios de otros componentes, estos sería fáciles de entender por separado y de reutilizar.

## **Modelo Vista Controlador (MVC).**

Es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos:

- Modelo: Esta es la representación específica de la información con la cual el sistema opera.
- Vista: Este presenta el modelo en un formato adecuado para interactuar, usualmente la interfaz de usuario.
- Controlador: Este responde a eventos, usualmente acciones del usuario e invoca cambios en el modelo y probablemente en la vista.

### **3.2.2 Estructura del Diseño.**

La Estructura del Diseño permite ver la relación de los subsistemas de diseño dentro de la arquitectura definida. Un subsistema define una forma coherente de examinar un aspecto del problema. Dividir el diseño de un sistema en subsistemas de diseño da la posibilidad de organizar el modelo de diseño en porciones más manejables. A continuación se presenta de forma gráfica los subsistemas de diseño.

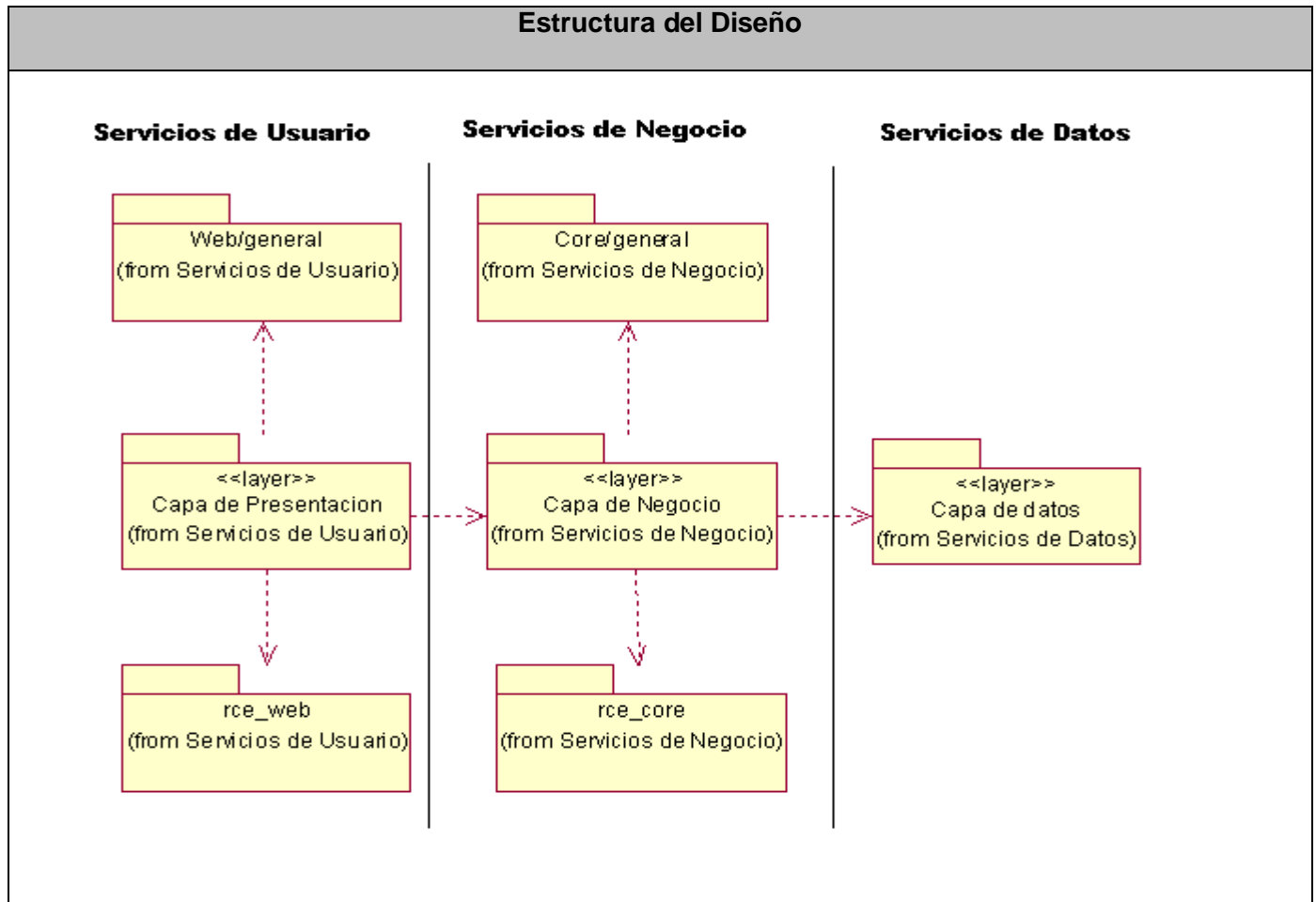


Figura 3.15 Estructura del Diseño.

A continuación se describen los subsistemas de servicios:

**Capa de Presentación:** Esta capa reúne todos los aspectos del software que tienen que ver con las interfaces y la interacción con los diferentes tipos de usuarios. Típicamente incluye el manejo y aspecto de las ventanas, el formato de los reportes, menús, gráficos y elementos multimedia en general. Esta capa se comunica únicamente con la capa de negocio.

**Capa de Negocio:** Esta capa reúne todos los aspectos del software que automatizan o apoyan los procesos de negocio que llevan a cabo los usuarios. Es donde residen los programas que se ejecutan,

recibiendo las peticiones del usuario y enviando las respuestas tras el proceso. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados y con la capa de datos, para solicitar, almacenar o recuperar datos de él.

**Capa de Datos:** Contiene las tablas de la base de datos del Registro de Ciudadanos 2.0, las que son gestionadas por la capa de negocio.

**Core/general:** Esta capa reúne todos los aspectos del software que tienen que ver con el manejo de los datos persistentes. Es donde residen los datos. Está formado por el servidor de bases de datos que realiza todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

**Web/general:** Este subsistema contiene los ficheros de configuración, de seguridad y de acceso a los diferentes módulos, los cuales son utilizados en la Capa de Presentación.

### 3.2.3 Definición de Elementos de Diseño.

La extensión UML para Web presenta como elementos más significativos a 3 clases de UML estereotipadas con los siguientes estereotipos “Server Page”, “Client Page”, “Form” empleados para el código servidor, código cliente y formularios respectivamente.

A continuación se describe en qué consiste cada estereotipo y cómo se utilizan en el diseño:



**<<Client Page>>:** Es una página Web con formato XHTML. Mezcla de datos, presentación y lógica, puede contener scripts interpretados por el browser.

**<<Server Page>>**: Página Web con scripts ejecutados por el servidor, los scripts interactúan con recursos del servidor (Base de datos, lógica del negocio, sistemas externos, etc).

**<<Form>>**: Formulario: colección de campos de entrada de datos, parte de una «Client Page». Los atributos representan campos de entrada del form HTML.

## **Estereotipos de asociación.**

**<<Build>>**: Representa la relación existente entre las páginas cliente, que de forma general expresa cómo las páginas que se encuentran en el servidor construyen las páginas en el cliente.

**<<Call>>**: Se utiliza para llamadas a páginas servidoras que representan métodos de la capa de negocio.

**<<Include>>**: Una página servidor puede incluir a otra página del mismo tipo, pudiendo utilizar todas las funciones brindadas por esta última.

**<<Link>>**: Permite ir de una página cliente a otra página cliente.

**<<Redirect>>**: Una página servidora puede re direccionar el procesamiento a otra página.

**<<Submit>>**: Envía los valores de un formulario a una página servidora.

### **3.2.4 Diagrama de Clases del Diseño.**

Un diagrama de clases es un tipo de diagrama estático que describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos, brindan un mayor acercamiento al contenido de la solución propuesta. El Registro de Ciudadanos 2.0 es una aplicación Web, para realizar estos diagramas de clases se tuvo en cuenta la extensión UML para Web. A continuación se presentan estos diagramas de clases:



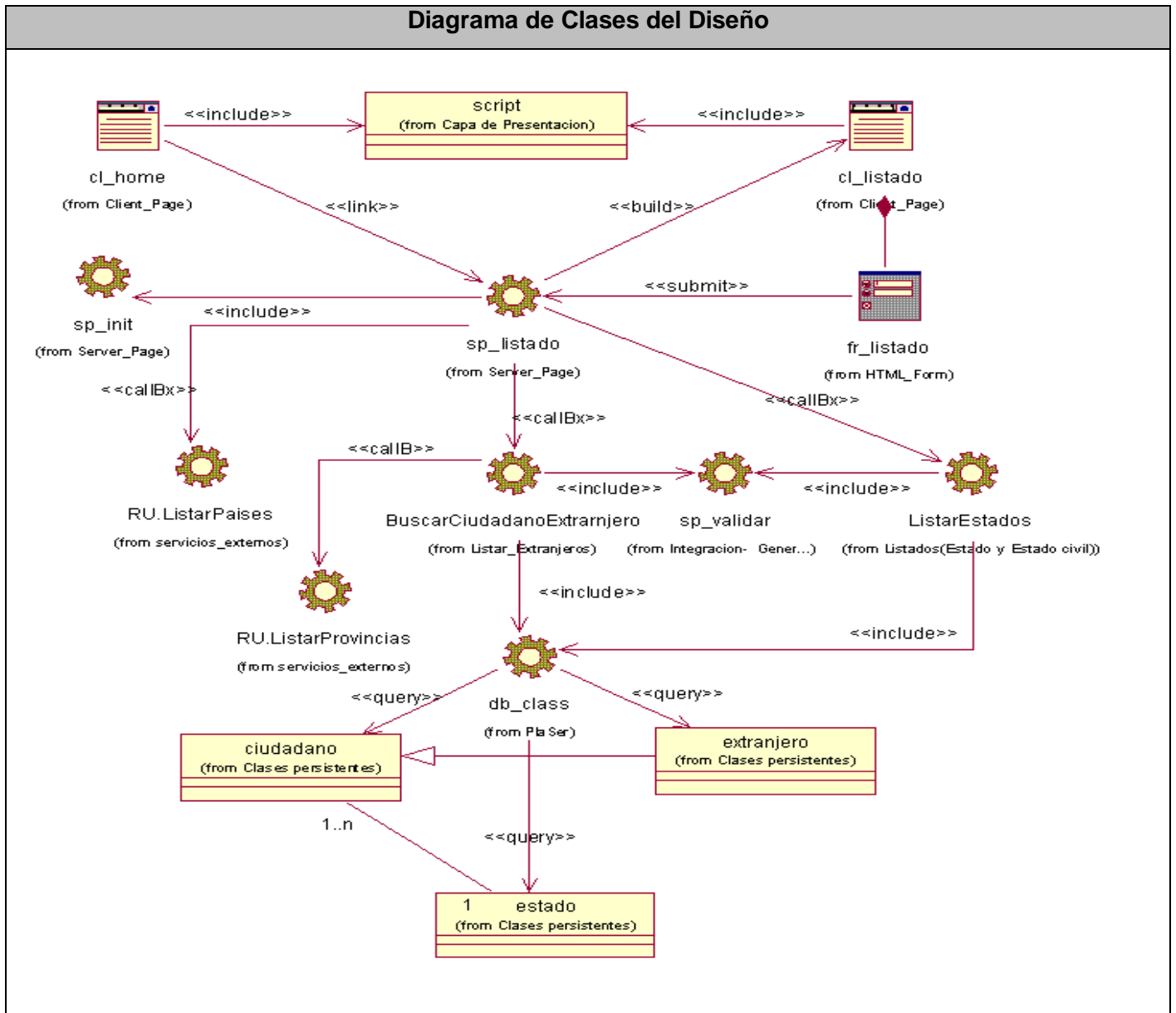


Figura 3.18 Diagrama de Clases del Diseño Caso de Uso: ListarCiudadanosExtranjeros.



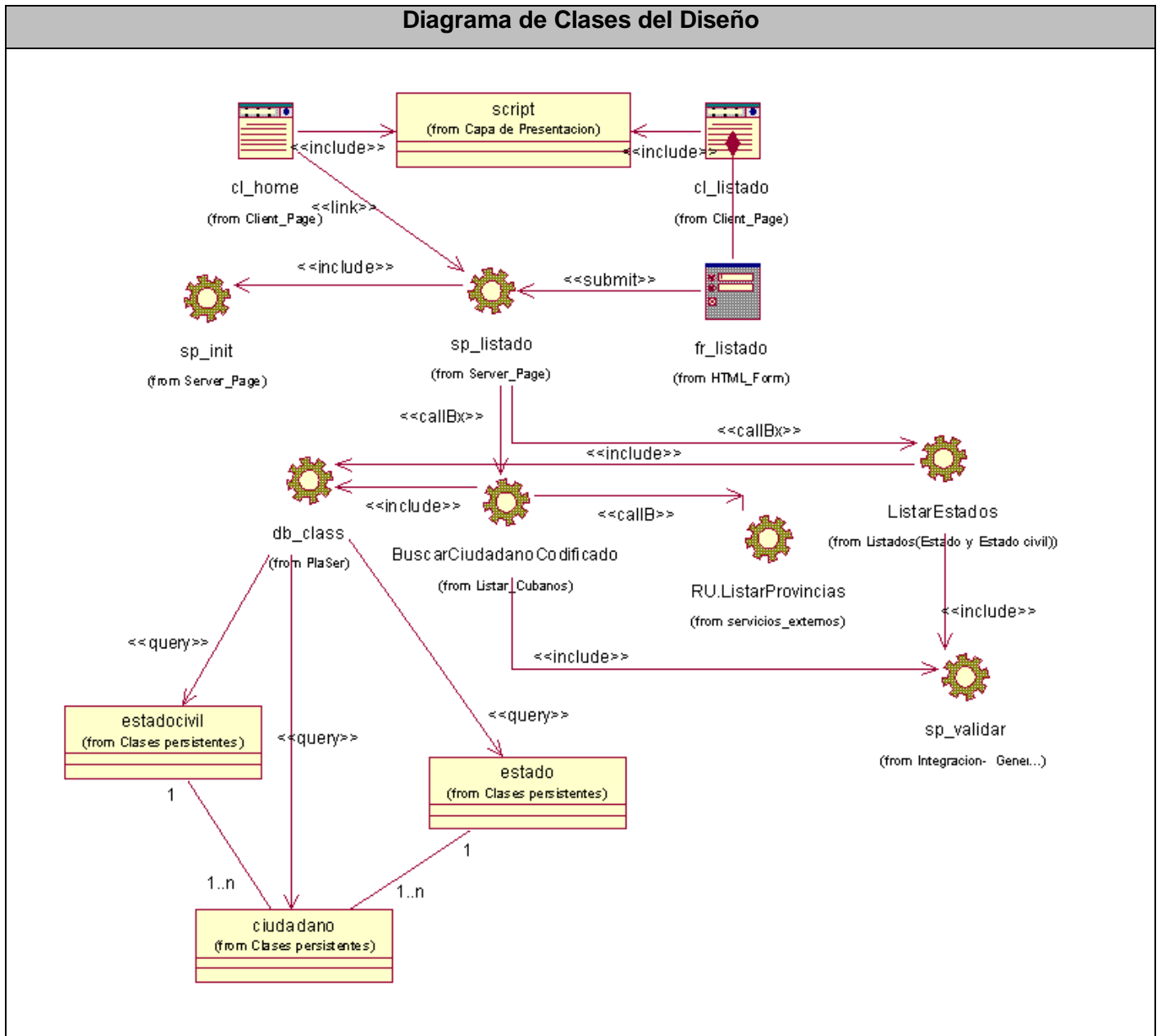


Figura 3.17 Diagrama de Clases del Diseño Caso de Uso: ListarCiudadanosCubanos.

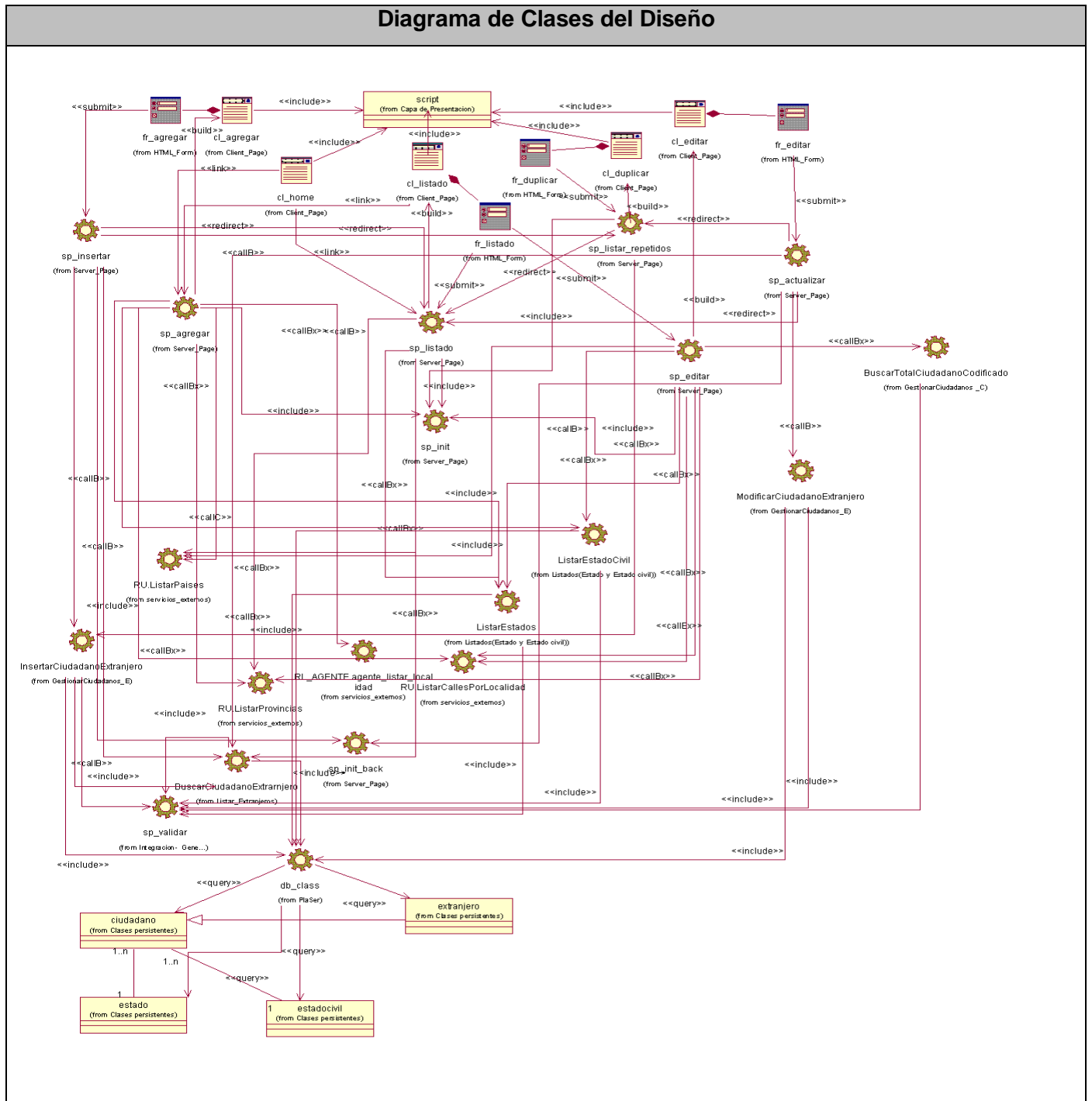


Figura 3.19 Diagrama de Clases del Diseño Caso de Uso: GestionarCiudadanoExtranjero.

### 3.2.5 Descripción de las clases y atributos.

A continuación se describen las clases del diseño mostrados anteriormente.

#### Páginas clientes.

<b>Nombre:</b> cl_listado
<b>Tipo de clase:</b> página cliente
<p><b>Descripción General:</b> La clase <i>cl_listado</i> es una página Web que se ejecuta del lado del cliente sobre un browser o navegador Web. Permite a los editores nacionales la realización de listado y búsquedas de información específica permitiendo generar documentos Portable Document Format (.pdf) y Microsoft Office Excel (.xls) para imprimir. A través de esta página se puede visualizar la información. El resultado de las búsquedas es paginada, permitiendo la movilidad por las diferentes páginas, inclusive ir directamente a la última. Es utilizada en los siguientes casos de uso:</p> <ul style="list-style-type: none"> <li>• ListarCiudadanosCubanos.</li> <li>• ListarCiudadanosExtranjeros.</li> </ul>

Tabla 3.1 Descripción página cliente cl\_listado.

<b>Nombre:</b> cl_editar
<b>Tipo de clase:</b> página cliente
<p><b>Descripción General:</b> La clase <i>cl_editar</i> es una página Web que se ejecuta del lado del cliente sobre un browser o navegador Web. Recibe los valores de la página cliente <i>cl_listado</i> .Permite a los editores nacionales modificar o visualizar información. Es utilizada en los siguientes casos de uso:</p> <ul style="list-style-type: none"> <li>• GestionarCiudadanoCubano.</li> <li>• GestionarCiudadanoExtranjero.</li> </ul>

Tabla 3.2 Descripción página cliente cl\_editar.

<b>Nombre:</b> cl_agregar
<b>Tipo de clase:</b> página cliente
<b>Descripción General:</b> La clase <i>cl_agregar</i> es una página Web que se ejecuta del lado del cliente sobre

un browser o navegador Web. Permite a los editores nacionales insertar información. Es utilizada en los siguientes casos de uso:

- GestionarCiudadanoCubano.
- GestionarCiudadanoExtranjero.

Tabla 3.3 Descripción página cliente cl\_agregar.

<b>Nombre:</b> cl_duplicar
<b>Tipo de clase:</b> página cliente
<p><b>Descripción General:</b> La clase <i>cl_duplicar</i> es una página Web que se ejecuta del lado del cliente sobre un browser o navegador Web. Permite a los editores nacionales la realización de duplicar o no de información de un ciudadano cuyo carné de identidad este repetido. Es utilizada en los siguientes casos de uso:</p> <ul style="list-style-type: none"> <li>• GestionarCiudadanoCubano.</li> <li>• GestionarCiudadanoExtranjero.</li> </ul>

Tabla 3.4 Descripción página cliente cl\_duplicar.

## Páginas servidoras.

<b>Nombre:</b> sp_listado
<b>Tipo de clase:</b> página servidora
<p><b>Descripción General:</b> La clase <i>sp_listado</i> es una clase que se ejecuta del lado del servidor en la capa de presentación. Su responsabilidad es construir la página cliente <i>cl_listado</i>. Aplica un documento (XSL) a otro documento XML para transformarlo y mostrarlo al usuario en formato XHTML para posteriormente recibir los parámetros de búsqueda. Es utilizada en los siguientes casos de uso:</p> <ul style="list-style-type: none"> <li>• ListarCiudadanosCubanos.</li> <li>• ListarCiudadanosExtranjeros.</li> </ul>

Tabla 3.5 Descripción página servidora sp\_listado.

<b>Nombre:</b> sp_actualizar
<b>Tipo de clase:</b> página servidora

**Descripción General:** Recibe los valores de la clase *sp\_editar*, los valida y los envía hacia la capa de negocio al método de modificación de los datos involucrados. Una vez concluida la ejecución de sus responsabilidades devuelve una respuesta a la clase *sp\_actualizar*. Es utilizada en los siguientes casos de uso:

- GestionarCiudadanoCubano.
- GestionarCiudadanoExtranjero.

Tabla 3.5 Descripción página servidora *sp\_actualizar*.

<b>Nombre:</b> <i>sp_insertar</i>
<b>Tipo de clase:</b> página servidora
<b>Descripción General:</b> Recibe los valores de la clase <i>sp_agregar</i> , los valida y los envía hacia la capa de negocio al método de inserción de los datos involucrados y una vez concluida la ejecución de sus responsabilidades devuelve una respuesta a la clase <i>sp_insertar</i> . Es utilizada en los siguientes casos de uso:
<ul style="list-style-type: none"> <li>• GestionarCiudadanoCubano.</li> <li>• GestionarCiudadanoExtranjero.</li> </ul>



Tabla 3.6 Descripción página servidora *sp\_insertar*.

<b>Nombre:</b> <i>sp_listar_repetidos</i>
<b>Tipo de clase:</b> página servidora
<b>Descripción General:</b> es una clase que se ejecuta del lado del servidor en la capa de presentación. Su responsabilidad es construir la página cliente <i>cl_duplicar</i> . Aplica un documento (XSL) a otro documento XML para transformarlo y mostrarlo al usuario en formato XHTML para posteriormente recibir los valores introducidos. Es utilizada en los siguientes casos de uso:
<ul style="list-style-type: none"> <li>• GestionarCiudadanoCubano.</li> <li>• GestionarCiudadanoExtranjero.</li> </ul>



Tabla 3.7 Descripción página servidora *sp\_listar\_repetidos*.

<b>Nombre:</b> <i>sp_agregar</i>
<b>Tipo de clase:</b> página servidora

**Descripción General:** La clase *sp\_agregar* es una clase que se ejecuta del lado del servidor en la capa de presentación. Su responsabilidad es construir la página cliente *cl\_agregar*. Aplica un documento (XSL) a otro documento XML para transformarlo y mostrarlo al usuario en formato XHTML para posteriormente recibir los valores introducidos. Es utilizada en los siguientes casos de uso:

- GestionarCiudadanoCubano.
- GestionarCiudadanoExtranjero.

Tabla 3.8 Descripción página servidora *sp\_agregar*.

<b>Nombre:</b> <i>sp_editar</i>
<b>Tipo de clase:</b> página servidora
<b>Descripción General:</b> La clase <i>sp_editar</i> es una clase que se ejecuta del lado del servidor en la capa de presentación. Su responsabilidad es construir la página cliente <i>cl_editar</i> . Aplica un documento (XSL) a otro documento XML para transformarlo y mostrarlo al usuario en formato XHTML para posteriormente recibir los valores modificados. Es utilizada en los siguientes casos de uso:
<ul style="list-style-type: none"> <li>• GestionarCiudadanoCubano.</li> <li>• GestionarCiudadanoExtranjero.</li> </ul>

Tabla 3.9 Descripción página servidora *sp\_editar*.

<b>Nombre:</b> <i>InsertarCiudadanoCodificado</i>	
<b>Tipo de clase:</b> página servidora	
<b>Descripción General:</b> La clase <i>InsertarCiudadanoCodificado</i> es una clase que se ejecuta del lado del servidor en la capa de negocio. Su función es insertar en la base de datos un ciudadano determinado con los datos pasados como parámetros. Es utilizada en el siguiente caso de uso:	
<ul style="list-style-type: none"> <li>• GestionarCiudadanoCubano.</li> </ul>	
Parámetros de Entrada:	
nombre	string
ape1	string
ape2	string
CI	string
talla	string
Peso	string
piel	string



ojos	string
sexo	string
numero	int
apto	string
padre	string
madre	string
tomo	string
folio	string
calle	string
entrecalles	string
fecha	datetime
idestado	int
municipioNac	string
municipioDir	string
localidadDir	string
id_entrecalle2	int
id_calle	int
id_entrecalle1	int
alias	string
id_pais	int
nacionalidad	string
idestadocivil	int
militar	int
nopasaporte	string
fechavencepas	datetime
lugar	string
municipioOrigen	string
provinciaOrigen	string
localidadYCalle	string

Tabla 3.10 Descripción página servidora InsertarCiudadanoCodificado.

<b>Nombre:</b> ModificarCiudadanoCodificado
<b>Tipo de clase:</b> página servidora
<b>Descripción General:</b> La clase <i>ModificarCiudadanoCodificado</i> es una clase que se ejecuta del lado del servidor en la capa de negocio. Su función es modificar en la base de datos un ciudadano según los parámetros de entrada. Es utilizada en el siguiente caso de uso: <ul style="list-style-type: none"> <li>• GestionarCiudadanoCubano.</li> </ul>

Parámetros de Entrada:	
idciudadano	int
idestado	int
nombre	string
ape1	string
ape2	string
padre	string
madre	string
carnetident	string
talla	string
peso	string
piel	string
ojos	string
sexo	string
alias	string
numero	int
calle	string
entre	string
apto	string
idmunicipio	int
id_localidad	int
id_entrecalle1	int
id_entrecalle2	int
id_calle	int
tomo	string
folio	string
idmunicipionacimiento	int
idestadocivil	int
fechanacimiento	datetime
militar	int
id_pais	int
idnacionalidad	int
nopasaporte	string
fechavencepas	datetime
lugar	string
municipioOrigen	string
provinciaOrigen	string
localidadYCalle	string

Tabla 3.11 Descripción página servidora ModificarCiudadanoCodificado.

<b>Nombre:</b> BuscarCiudadanoCodificado	
<b>Tipo de clase:</b> página servidora	
<p><b>Descripción General:</b> La clase <i>BuscarCiudadanoCodificado</i> es una clase que se ejecuta del lado del servidor en la capa de negocio. Su función es buscar en la base de datos un o los ciudadano según el o los parámetros de entrada. Es utilizada en el siguiente caso de uso:</p> <ul style="list-style-type: none"> <li>• ListarCiudadanosCubanos.</li> </ul>	
Parámetros de Entrada:	
idciudadano	int
idestado	int
nombre	string
ape1	string
ape2	string
carnetident	string
sexo	string
edad1	Int
edad2	int
ordenarpor	string
orden	string
offset	int
cantidad	int
tipociudadano	int
idpais	int
nopasaporte	string

Tabla 3.12 Descripción página servidora BuscarCiudadanoCodificado.

<b>Nombre:</b> InsertarCiudadanoExtranjero	
<b>Tipo de clase:</b> página servidora	
<p><b>Descripción General:</b> La clase <i>InsertarCiudadanoExtranjero</i> es una clase que se ejecuta del lado del servidor en la capa de negocio. Su función es insertar en la base de datos un o los ciudadanos extranjeros según el o los parámetros de entrada. Es utilizada en el siguiente caso de uso:</p> <ul style="list-style-type: none"> <li>• GestionarCiudadanoExtranjero.</li> </ul>	
Parámetros de Entrada:	
idciudadano	int
nopasaporte	string
fechavencepas	datetime

lugar	string
municipioOrigen	string
provinciaOrigen	string
localidadYCalle	string

Tabla 3.13 Descripción página servidora InsertarCiudadanoExtranjero.

<b>Nombre:</b> ModificarCiudadanoExtranjero.	
<b>Tipo de clase:</b> página servidora	
<b>Descripción General:</b> La clase <i>ModificarCiudadanoExtranjero</i> es una clase que se ejecuta del lado del servidor en la capa de negocio. Su función es modificar en la base de datos un ciudadano extranjero según los parámetros de entrada. Es utilizada en el siguiente caso de uso: <ul style="list-style-type: none"> <li>• GestionarCiudadanoExtranjero.</li> </ul>	
<b>Parámetros de Entrada:</b>	
idciudadano	int
nopasaporte	string
fechavencepas	datetime
lugar	string
municipioOrigen	string
provinciaOrigen	string
localidadYCalle	string

Tabla 3.14 Descripción página servidora ModificarCiudadanoExtranjero.

<b>Nombre:</b> BuscarCiudadanoExtranjero.	
<b>Tipo de clase:</b> página servidora	
<b>Descripción General:</b> La clase <i>BuscarCiudadanoExtranjero</i> es una clase que se ejecuta del lado del servidor en la capa de negocio. Su función es buscar en la base de datos un o los ciudadanos extranjeros según el o los parámetros de entrada. Es utilizada en el siguiente caso de uso: <ul style="list-style-type: none"> <li>• ListarCiudadanosExtranjeros.</li> </ul>	
<b>Parámetros de Entrada:</b>	
idciudadano	int
idestado	int
nombre	string
ape1	string
ape2	string
Provincia	string

Municipio	string
Localidad	string
Calle	string
Entre1	string
Entre2	string
edad1	Int
edad2	int
ordenarpor	string
orden	string
offset	int
cantidad	int

Tabla 3.15 Descripción página servidora BuscarCiudadanoExtranjero.

<b>Nombre:</b> RL_AGENTE.agente_listar_localidad
<b>Tipo de clase:</b> página servidora
<b>Descripción General:</b> La clase <i>RL_AGENTE.agente_listar_localidad</i> un servicio externo que se consume, en este caso de Registro de Localidad(RL), para traer el listado correspondientes a las localidades necesarias para conformar las direcciones de residencia de los ciudadanos. Es utilizada en los siguientes casos de uso: <ul style="list-style-type: none"> <li>• GestionarCiudadanoCubano.</li> <li>• GestionarCiudadanoExtranjero.</li> </ul>

Tabla 3.16 Descripción página servidora RL\_AGENTE.agente\_listar\_localidad.

<b>Nombre:</b> RU.ListarCallesPorLocalidad
<b>Tipo de clase:</b> página servidora
<b>Descripción General:</b> La clase <i>RU.ListarCallesPorLocalidad</i> un servicio externo a consumir, en este caso de Registro de Ubicación (RU), para traer el listado correspondientes a las calles de las localidades necesarias para conformar las direcciones de residencia de los ciudadanos. Es utilizada en los siguientes casos de uso: <ul style="list-style-type: none"> <li>• GestionarCiudadanoCubano.</li> <li>• GestionarCiudadanoExtranjero.</li> </ul>

Tabla 3.17 Descripción página servidora RU.ListarCallesPorLocalidad.

<b>Nombre:</b> RU.ListarPaises
<b>Tipo de clase:</b> página servidora
<p><b>Descripción General:</b> La clase <i>RU.ListarPaises</i> un servicio externo a consumir, en este caso de Registro de Ubicación (RU), para traer el listado correspondientes a los países. Es utilizada en los siguientes casos de uso:</p> <ul style="list-style-type: none"> <li>• GestionarCiudadanoExtranjero.</li> <li>• ListarCiudadanosExtranjeros.</li> </ul>

Tabla 3.18 Descripción página servidora RU.ListarPaises.

<b>Nombre:</b> RU.ListarProvincias
<b>Tipo de clase:</b> página servidora
<p><b>Descripción General:</b> La clase <i>RU.ListarProvincias</i> un servicio externo consumido, en este caso de Registro de Ubicación (RU), para traer los datos correspondientes a las provincias y los municipios del RU. Es utilizada en los siguientes casos de uso:</p> <ul style="list-style-type: none"> <li>• GestionarCiudadanoCubano.</li> <li>• GestionarCiudadanoExtranjero.</li> <li>• ListarCiudadanosCubanos.</li> <li>• ListarCiudadanosExtranjeros.</li> </ul>

Tabla 3.19 Descripción página servidora RU.ListarProvincias.

<b>Nombre:</b> ListarEstadoCivil
<b>Tipo de clase:</b> página servidora

<p><b>Descripción General:</b> La clase <i>ListarEstadoCivil</i> es una clase que se ejecuta del lado del servidor en la capa de negocio. Su función es listar los estados civiles de la base de datos. Es utilizada en el siguiente caso de uso:</p> <ul style="list-style-type: none"> <li>• GestionarCiudadanoCubano.</li> <li>• GestionarCiudadanoExtranjero.</li> <li>• ListarCiudadanosCubanos.</li> <li>• ListarCiudadanosExtranjeros.</li> </ul>	
Parámetros de Entrada:	
cantidad	int
offset	int

Tabla 3.20 Descripción página servidora ListarEstadoCivil.

<b>Nombre:</b> ListarEstados	
<b>Tipo de clase:</b> página servidora	
<p><b>Descripción General:</b> La clase <i>ListarEstados</i> es una clase que se ejecuta del lado del servidor en la capa de negocio. Su función es listar los estados de la base de datos. Es utilizada en el siguiente caso de uso:</p> <ul style="list-style-type: none"> <li>• GestionarCiudadanoCubano.</li> <li>• GestionarCiudadanoExtranjero.</li> <li>• ListarCiudadanosCubanos.</li> <li>• ListarCiudadanosExtranjeros.</li> </ul>	
Parámetros de Entrada:	
cantidad	int
offset	int
tipociudadano	int
Arreglold	array

Tabla 3.21 Descripción página servidora ListarEstados.

## Otras páginas servidoras.

Clase	Propósito
sp_init	Clase encargada de la seguridad. Extiende clases de la librería PlaSer como: client, xml, xls y utiles_gui.
sp_init_back	Verifica los derechos del usuario. Extiende clases de la librería PlaSer como: client y utiles_gui.
sp_dbz_class	Clase que realiza la conexión con las bases de datos MySQL, usa el modulo dbx de PHP para su funcionalidad. Además crea un objeto conexión que permite hacer consultas, y recuperar los resultados; insertar, eliminar y actualizar datos. Esta clase se encuentra en la capa de negocio, la misma está dentro de la librería PlaSer.

En el capítulo se describieron y justificaron los patrones de diseño utilizados. Se definió la estructura y los elementos de diseño. Además, se muestran los Diagramas de Clases del Análisis, los Diagramas de Iteración y los Diagramas de Clases del Diseño, describiéndose las clases y atributos involucrados en los mismos.



## CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA

En este capítulo se describe el proceso de implementación a partir del diseño de la solución anteriormente propuesta. Se justifica la integración con otros componentes del Registro Informatizado de Salud (RIS). Se muestran los diagramas de componentes del Registro de Ciudadanos 2.0, así como el diagrama de despliegue. Se describen los métodos del negocio más complejos o agentes de ambos registros. Se detallan las pautas que se definieron para estandarizar las interfaces, así como los estándares de codificación y tratamientos de errores.

### 4.1 Justificación de integración con otros sistemas.

El Registro de Ciudadanos 2.0 al igual que todos los componentes que se integren al Sistema de Información para la Salud tiene relación con el Módulo de Administración, el cual tiene implementado el Componente de Seguridad (SAAA).

#### Componente de Seguridad (SAAA).

Está basado en el modelo de Autenticación, Autorización y Auditoría (AAA). La autenticación debe ser la primera acción del usuario en el sistema y consiste en suministrar un nombre de usuario único y una contraseña que debe ser de conocimiento exclusivo de la persona que se autentica. Si el usuario autenticado no se encuentra registrado se reporta un error de acceso. En caso contrario, se autoriza su acceso y se crea un certificado digital y se retornan todos los datos y permisos del usuario, desglosado por módulos. Cada Petición de usuario, autorizada o no, es registrada, así como el día, mes, año, hora, minuto, segundo en que se registra y si fue o no autorizada.<sup>31</sup>

---

<sup>31</sup> Ídem referencia 10

## **Registro de Ubicación.**

El Registro de Ciudadanos 2.0 se relaciona con el Registro de Ubicación, del cual toma la información de los Países, las Provincias, Municipios y Calles. Se utilizan estos datos para ubicar en ellos a los ciudadanos.

## **Registro de Localidades.**

El Registro de Localidades organiza el país de forma estratégica, permitiendo al estado cubano la toma de decisiones; y gestiona información de los Consejos Populares, Circunscripciones, Zonas y Comités de Defensa de la Revolución (CDR). El Registro de Ciudadanos 2.0 se relaciona con el Registro de Localidades, del cual toma la información de las localidades.

## **4.2 Modelo de Implementación.**

Los diagramas de Despliegue y Componentes conforman lo que se conoce como modelo de implementación. Este describe los componentes a construir, su organización y la dependencia entre nodos físicos, en los que funcionará el sistema o aplicación.

El modelo de implementación del Registro de Ciudadanos 2.0 expone una organización en capas, jerarquías de paquetes y subsistemas de implementación. Los mismos contienen componentes y sus relaciones, dividiendo al sistema en trozos más manejables. Esto posibilita la reutilización, que se pueda implementar por separado y disminuye el impacto que pueda traer consigo un cambio. A continuación se presenta una vista global de la estructura y organización de la implementación del sistema:

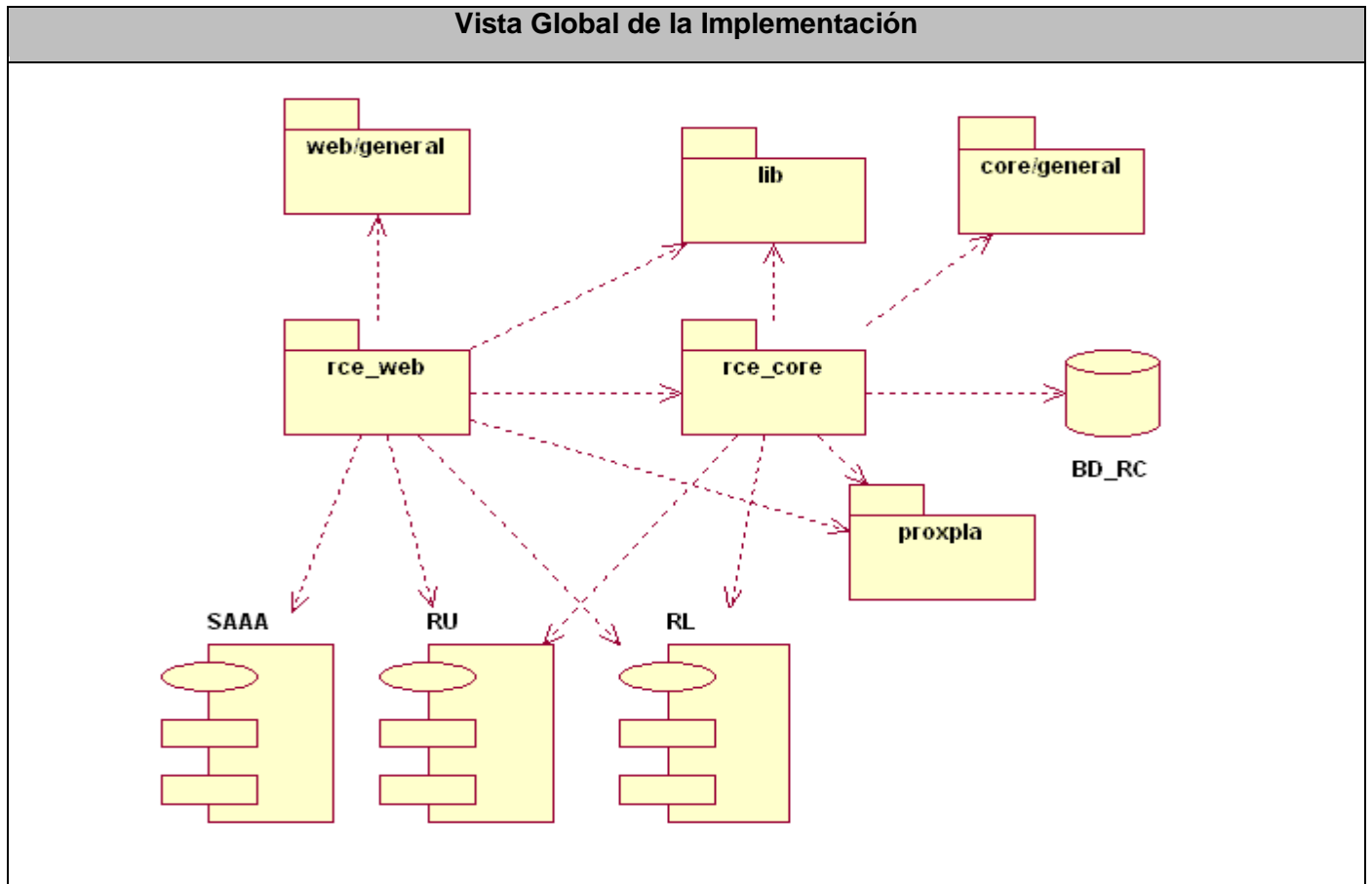


Figura 4.1 Vista Global de la Implementación.

#### 4.2.1 Diagramas de Componentes.

Un Diagrama de Componentes muestra las organizaciones y dependencias lógicas entre componentes software, sean éstos componentes de código fuente, binarios o ejecutables. Desde el punto de vista del Diagrama de Componentes se tienen en consideración los requisitos relacionados con la facilidad de desarrollo, la gestión del software, la reutilización, y las restricciones impuestas por los lenguajes de programación y las herramientas utilizadas en el desarrollo. Los elementos de modelado dentro de un diagrama de componentes serán componentes y paquetes.

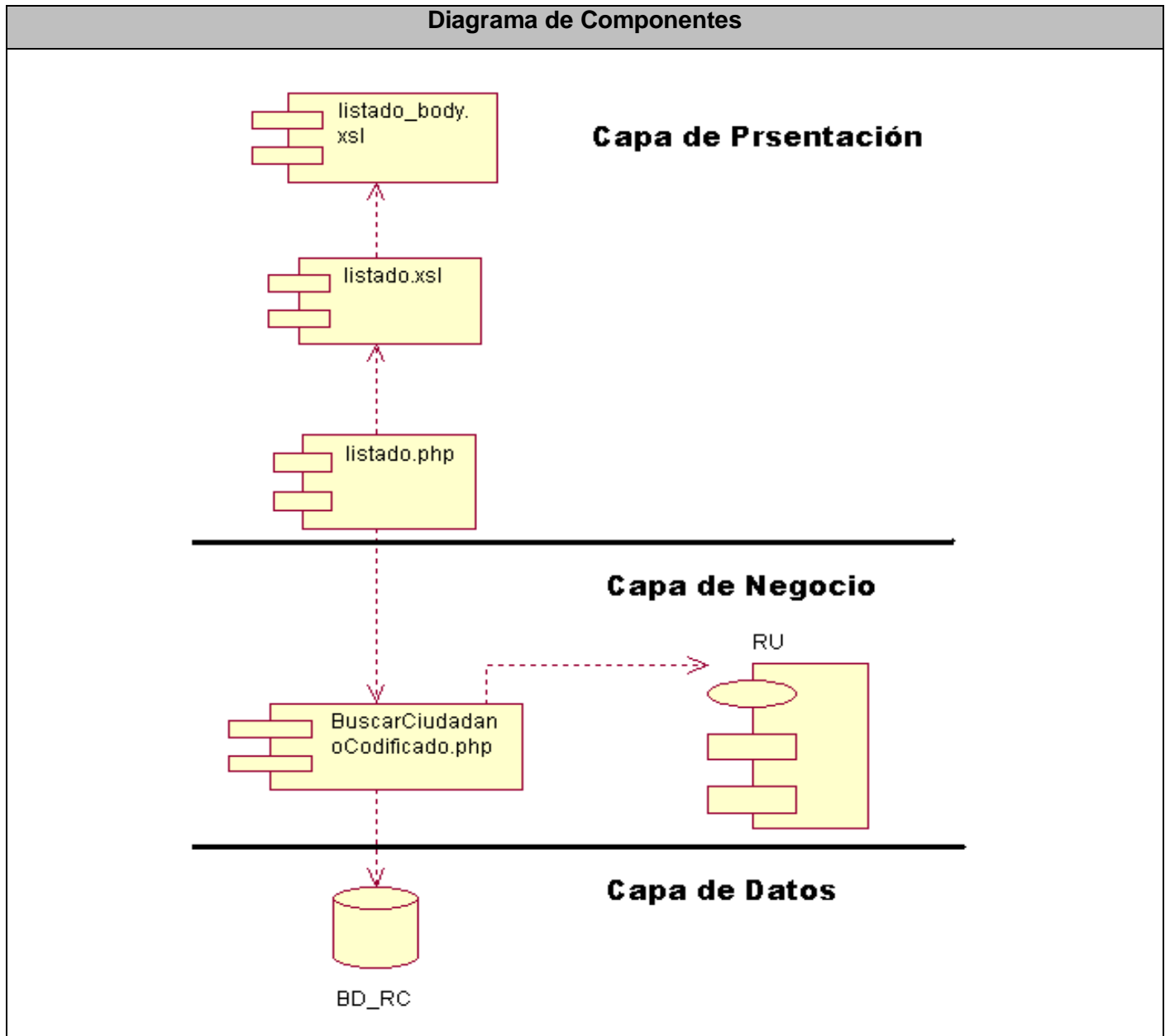


Figura 4.2 Diagrama de Componentes del Caso de Uso: ListarCiudadanosCubanos.

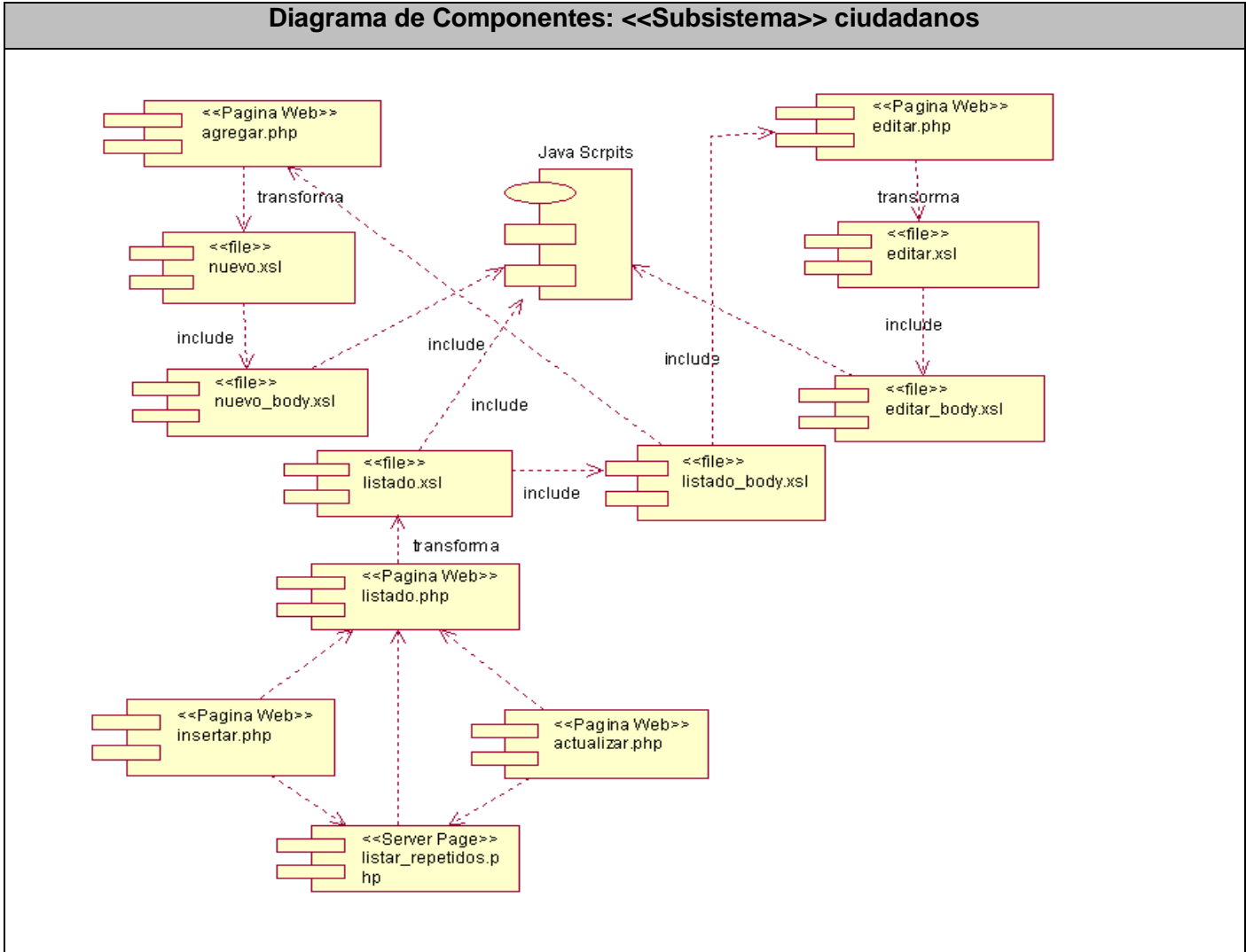


Figura 4.3 Diagrama de Componentes: <<Subsistema>> ciudadanos.

**4.2.2 Diagrama de Despliegue.**

Un Diagrama de Despliegue muestra las relaciones físicas entre los componentes hardware y software en el sistema final, es decir, la configuración de los elementos de procesamiento en tiempo de ejecución y los componentes software (procesos y objetos que se ejecutan en ellos). Están formados por instancias de los componentes software que representan manifestaciones del código en tiempo de ejecución (los

componentes que sólo sean utilizados en tiempo de compilación deben mostrarse en el diagrama de componentes). Un Diagrama de Despliegue es un grafo de nodos unidos por conexiones de comunicación.

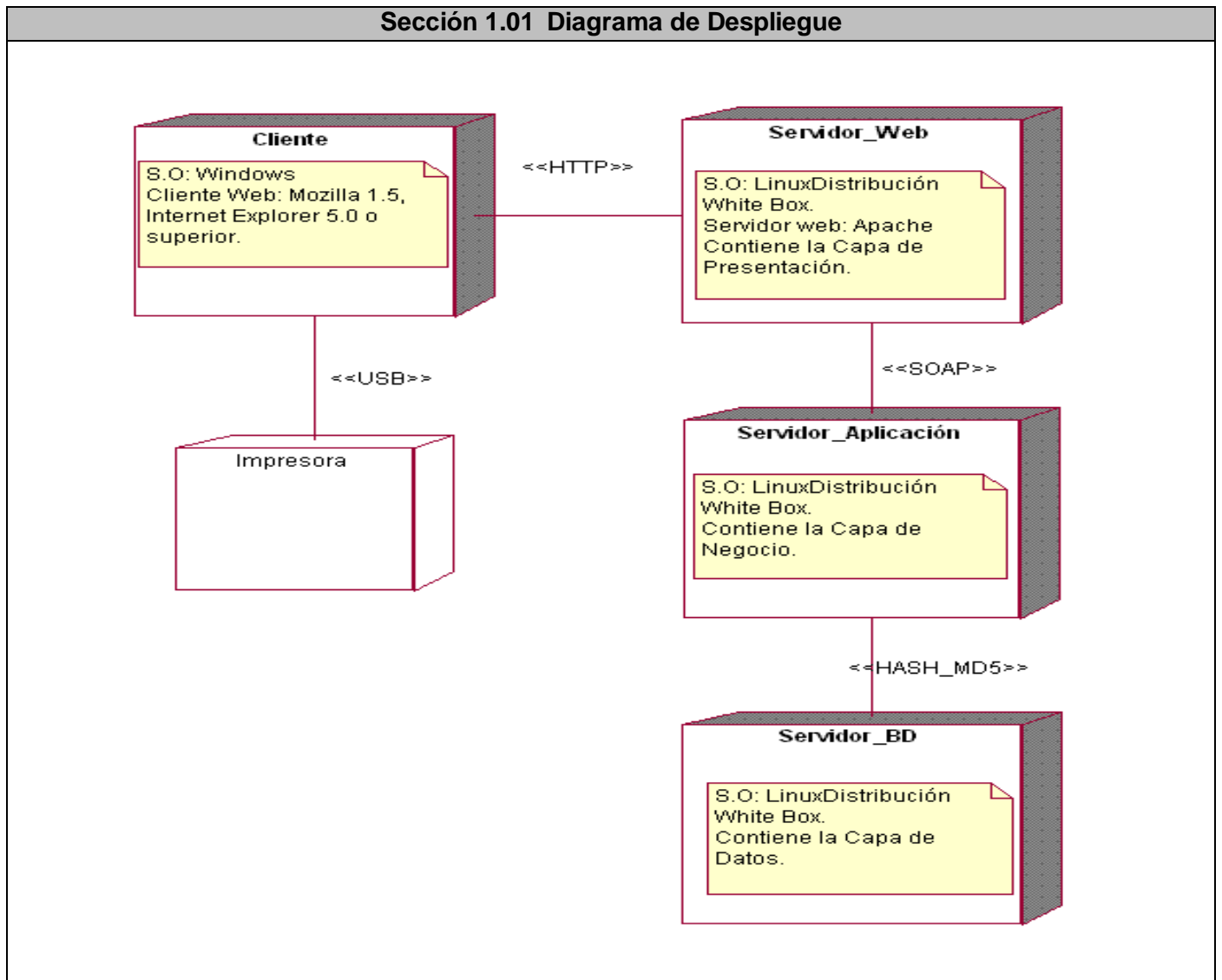


Figura 4.4 Diagrama de Despliegue.

### 4.2.3 Descripción de los Métodos.

A continuación se presenta la descripción de algunos de los métodos del negocio del Registro de Ciudadanos 2.0.

**BuscarCiudadanoCodificado.php:** Este método lista la información de los Ciudadanos cubanos según los parámetros de entrada. Recibe los siguientes parámetros: idciudadano, idestado, nombre, ape1, ape2, carnetident, sexo, edad1, edad2, ordenarpor, orden, offset, cantidad, tipociudadano (puede ser 1 es cubano y 2 es extranjero), idpais y nopasaporte. El método retorna los datos de los Ciudadanos cubanos que cumplan con los criterios de búsqueda especificados. Pueden acceder a él usuarios editores a nivel Nacional.

**InsertarCiudadanoCodificado.php:** este método permite insertar los ciudadanos cubanos. Recibe los siguientes parámetros: nombre, ape1, ape2, padre, madre, CI, talla, peso, piel, ojos, calle, numero, entre, apto, municipioNac, MunicipioDir, LocalidadDir, id\_calle, id\_entrecalle1, id\_entrecalle2, id\_pais, tomo, folio, sexo, idestado, fecha, alias, idestadocivil, militar, idnacionalidad, nopasaporte, fechavencepas, lugar, municipioOrigen, provinciaOrigen, LocalidadYCalle. El método retorna el idciudadano para validar que el ciudadano fue insertado.

**InsertarCiudadanoExtranjero.php:** este método permite insertar los ciudadanos extranjeros. Recibe los siguientes parámetros: idciudadano, nopasaporte, fechavencepas, lugar, municipioOrigen, provinciaOrigen, localidadYCalle. El método retorna el idciudadano para validar que el ciudadano fue insertado.

**ModificarCiudadanoCodificado.php:** este método permite modificar los ciudadanos cubanos. Recibe los siguientes parámetros: nombre, ape1, ape2, padre, madre, carnetident, talla, peso, piel, ojos, sexo, alias, numero, calle, entre, apto, idmunicipio, id\_localidad, id\_entrecalle1, id\_entrecalle2, id\_calle, tomo, folio, dmunicipionacimiento, idestadocivil, fechanacimiento, militar, id\_pais, idnacionalidad, nopasaporte, fechavencepas, lugar, municipioOrigen, provinciaOrigen, localidadYCalle. Pueden acceder a él usuarios editores a nivel Nacional.

**ModificarCiudadanoExtranjero.php:** este método permite modificar los ciudadanos extranjeros. Recibe los siguientes parámetros: idciudadano, nopasaporte, fechavencepas, lugar, municipioOrigen, provinciaOrigen, localidadYCalle. Pueden acceder a él usuarios editores a nivel Nacional.

### 4.3 Estándares de diseño, codificación y tratamiento de errores.

Los esfuerzos de diseño están orientados a lograr una interfaz clara y fácil de usar para los usuarios, ya que a través de ésta es que interactúan con los recursos que se ponen a su disposición en el ordenador.

El Registro de Ciudadanos 2.0 siguen las pautas de diseño del Registro Informatizado de Salud. Estas pautas permiten lograr una mayor eficiencia en el proceso de trabajo, y sobre todo lograr una coherencia formal entre todos los módulos del sistema, y que sean identificados así como parte de un todo.

La resolución utilizada para el diseño de la aplicación es de 800 x 600 pixel. Se definió un estilo CSS (del inglés, Cascading Style Sheets, Hoja de Estilo en Cascada) para permitir separar el contenido de la presentación y lograr una homogeneidad en las letras, imágenes, tablas, controles, encabezados, etc. El CCS permite manejar los cambios que se hagan al diseño de forma centralizada ya que un cambio en él modifica el estilo de todos los implicados.

Para el diseño de cualquier interfaz que esté en el RIS se debe tener en cuenta que su diseño va a estar dividido en 4 partes, el header o cabecera, menú, región editable y footer o pie.

En el header se encuentra el nombre de la aplicación que aparece en la parte superior izquierda (A); tres vínculos: Páginas de Inicio, Acerca de y Ayuda, los cuales permiten entrar a las páginas de Inicio, una explicación del RIS y la ayuda de cada página respectivamente (B). Cuenta con una imagen que identifica al módulo en la parte izquierda (C); en la parte central el nombre y una breve descripción del módulo (D). En la parte derecha aparece el nombre del Usuario, Derechos (Administrador, Visualizador o Editor), Nivel (Nacional, Provincial, Municipal o Unidades de Salud) y módulos a los cuales el usuario que se autenticó en el RIS tiene privilegios (E).

En la parte izquierda se encuentra el menú donde se listan un conjunto de vínculos a los cuales el usuario tiene permisos para entrar (F).



En la parte central se encuentra la región editable en la cual se realiza el proceso de gestión de la información, ya sea, un listado, agregar o editar (G).

En la parte inferior se encuentra el footer el cual describe los derechos y dice el ministerio al que pertenece la aplicación (H).

El logo siempre estará ubicado en el extremo superior izquierdo de la página, es una imagen que cuenta con un ancho de 270 px y se corresponde con el nombre de cada módulo.

La tipografía será siempre Tahoma, por su amplia legibilidad y por las facilidades conocidas que brinda para la lectura digital. <sup>32</sup>

---

<sup>32</sup> *Pautas de diseño RIS*. La Habana, Proyecto APS – SOFTEL.

**Interfaz de usuario del Registro de Ciudadanos 2.0**

**RIS.CU** Registro Informatizado de Salud  
programa de informatización de la sociedad cubana

Página de Inicio | Acerca de

**MODULO**  
**Registro de Ciudadanos**  
Control de todos los Ciudadanos del país.

Usuario: **reynaldoce**  
Derechos: **Editor**  
Nivel: **Nacional**  
Módulos: Registro de Ciudadanos(Tesis) | Cerrar Sesión

**Opciones de Menú**

**Ciudadanos**

- Listado
- Nuevo

**Listado general - CIUDADANOS CUBANOS**

CI:  Sexo: << Seleccione >> Estado: << Seleccione >>

Nombre:  Primer Apellido:  Segundo Apellido:

Ordenar por: Nombre  DESC  Buscar

Exportar a  Página # 1

CI	Nombre (s)	Apellidos	Sexo	Provincia
111	AAA	SSS FFF	M	Ciudad de La Habana
59093002202	Alexander	Paneque O´ Reilly	M	Cienfuegos
87081431223	Alfredo Ernesto	Reytor Batista	M	Ciudad de La Habana
56091900750	Alicia María	Montero Montero	F	Cienfuegos
71060800001	Amaylid	Arteaga Arteaga	F	Cienfuegos
68091605657	Ana Amelia	Pino Pino	F	Cienfuegos
43072411807	Angel	Padrón Soto	M	Cienfuegos
83030600664	Ariel	Delgado Nuñez	M	Cienfuegos
08031720413	Camilo	Perez Alonso	F	Ciudad de La Habana
48122508516	Carelia Natividad	Betancourt Betancourt	F	Cienfuegos

1 - 10/67

Exportar a  Página # 1

Copyright © 2003. RIS - Registro Informatizado de Salud Cubano. MINSAP. Ministerio de Salud Pública.

Figura 4.5 Interfaz de usuario del Registro de Ciudadanos 2.0.

## Estándares de codificación.

Actualmente se encuentran estándares de codificación para la mayoría de los lenguajes existentes. El uso de los mismos, partiendo de las convenciones definidas, permite una mejor comunicación entre los programadores, creando las condiciones para la reusabilidad y el mantenimiento de los sistemas. Para definir el estilo de codificación a seguir en la aplicación se utilizó la notación estándar establecida para

aplicaciones desarrolladas en PHP (PHP CODIG Standard), que mayormente está basada en el estándar de código para aplicaciones en C++ (C++ CODIG Standard).

Las etiquetas de apertura y cierre del lenguaje serán de la forma `<?php ?>`, ya que siempre están disponibles en cualquier configuración.

Se hará uso de los arreglos predefinidos para el manejo de los valores enviados por el usuario `$_GET`, `$_POST`, `$_FILES` evitando el uso de `$_REQUEST`.

### **Convenciones específicas.**

- Para nombrar las variables se seguirá la regla de escribir los identificadores con letras minúsculas y en lenguaje español, utilizando como separador para las palabras el carácter “\_”, tratando de usar nombres sugerentes a la acción de la variable.

Todos los campos identificadores van a comenzar con el identificador (id) seguido del nombre del campo.  
Ejemplo: id \_ país.

- Los arreglos empezarán con el identificador array y las palabras no se separarán con el carácter “\_”.
- Las estructuras se identificarán poniendo al final del nombre struct. Ejemplo: paginadostruct.
- En el caso de las clases se pondrá delante la letra C. Ejemplo: CFachada.
- El idioma de las clases auxiliares como sesión y error, será el inglés para garantizar la homogeneidad con las programadas en este ámbito en el mundo, en el caso de los Servicios Web y la interface de administración se usará el español para esclarecer los objetivos de cada método o script a utilizar.
- En los métodos no se usarán abreviaturas y las palabras continuas deben comenzar con mayúsculas. Ejemplo: BuscarCiudadanoCodificado.
- Cada método que se defina debe seguir la siguiente estructura, acrónimo del módulo. Nombre del método. Ejemplo: RC.ListarEstados.

- Para comentar el código se utilizará, en el caso de una línea, al final de la misma el carácter “//” y seguido el comentario y en el caso de un bloque se utilizará los caracteres “/\* \*/”.
- Se usará una indentación en el código de cuatro espacios para facilitar la lectura de éste. Las llaves se usarán poniendo la llave inicial en una línea para ella sola, y en su respectiva columna la llave final también en una línea.
- El almacenamiento de la información será en scripts SQL para construir la base de datos e interactuar con ella desde las aplicaciones.

Para la capa de datos tienen que nombrar la base de datos poniendo el identificador del proyecto “APS” seguido del carácter “\_” y del nombre del módulo. Ejemplo: bd\_RC

Las tablas se identificarán con el acrónimo tb\_ciudadano, ejemplo: tb\_estado.

Los nombres de las tablas deben ir en minúsculas y cada palabra separada por línea abajo “\_”. (Ejemplo: Id\_nombre\_tabla)

Los campos de la base de datos se nombrarán igual que las variables.

Las palabras correspondientes a las sentencias SQL y sus parámetros deben ir en mayúsculas Ej: SELECT \* FROM tb\_estado

- En el caso de los XSL será con el mismo nombre que el fichero de la capa de presentación.

Los controles seguirán el siguiente tratamiento:

Control	Prefijo	Ejemplo
Botón	Btn	btnAceptar
Etiqueta	Lbl	lblNombre
Lista/Menú	Mn	mnPrincipal
Campo de Texto	Txt	txtFecha
Botón de Opción	Opt	optSexo
Casilla de Verificación	Chx	chxBorrar

Grid o rejilla	Grid	grUsuario
----------------	------	-----------

Tabla 4.1 Estándares para los controles.

Las páginas HTML se harán sin incluir código y todas las funciones JavaScript que se usarán se escribirán dentro de ficheros “.js”.

### Tratamiento de errores.

Se utiliza JavaScript para depurar los errores de parte del cliente validando los formularios y evitando consultas a la base de datos sin sentido o que tengan malas intenciones como son las inyecciones SQL, en cualquiera de estas situaciones el JavaScript funciona como una barrera y no deja pasar estos valores lanzando alertas o excepciones.



Figura 4.6 Ejemplo de una alerta en JavaScript.

Otros errores en la capa de negocio serán tratados devolviendo un SOAP\_FAULT, cuyos elementos FaultCode, FaultString, FaultAutor se describen a continuación:

**FaultCode:** Código de texto utilizado para indicar la clase de error, será codificado de la siguiente manera. Código del proyecto-código del módulo (:) número del método (.) número del error. Ejemplo: APS-RC: 1.5 que indica error 5 en el método 1 del módulo Registro de Ciudadanos perteneciente al Proyecto APS.

**FaultString:** Una explicación del error asequible al humano (leíble y explicativo). Debe tenerse en cuenta que este texto puede ser mostrado al operador final del sistema. Ejemplo: Formato de entrada no válido para la fecha de cierre estadístico.

**FaultAutor:** Un texto que indica quien provocó el error, siempre será el nombre del método que eleva la excepción. Ejemplo: RC\_BuscarCiudadanoExtramjero.

**Detail:** Este elemento se usa para llevar mensajes de error específicos de aplicaciones, se empleará únicamente en errores cuya resolución depende del Centro de Control, en cualquier otro caso este elemento debe estar vacío.<sup>33</sup>

### **4.4 Prueba.**

La fase de Pruebas del sistema tiene como objetivo verificar el sistema software para comprobar si este cumple sus requisitos. Dentro de esta fase pueden desarrollarse varios tipos distintos de pruebas en función de los objetivos de las mismas. Algunos tipos son pruebas funcionales, pruebas de usabilidad, pruebas de rendimiento, pruebas de seguridad, etc. Estas pruebas verifican que el sistema software ofrece a los actores humanos la funcionalidad recogida en su especificación. Una de las técnicas más empleadas para la especificación funcional de sistemas software son los casos de uso. Las principales ventajas de los casos de uso son que ocultan los detalles internos del sistema, son rápidos de construir, fáciles de modificar y entender por los clientes y futuros usuarios del sistema.

#### **4.4.1 Métodos de Prueba.**

Los Métodos de Prueba fundamentales son: el método de Caja Negra y de Caja Blanca. Al sistema se le realizaron las pruebas de Caja Negra.

#### **Prueba de Caja Negra.**

Las pruebas que se llevan a cabo sobre la interfaz del software, por lo que los caso de prueba pretenden demostrar que las funcionalidades del software son operativas, que la entrada se acepta de forma adecuada y que se produce una salida correcta. Esta prueba examina algunos aspectos del modelo fundamentalmente del sistema si tener mucho en cuenta la estructura interna del software.

---

<sup>33</sup> GÓMEZ VELÁZQUEZ, K. *APS\_Estándares\_Tratamiento\_Errores\_PHP\_BD*. La Habana, Proyecto APS – SOFTEL, 2005.

## 4.4.1.1 Modelo de Prueba.

**Nombre del caso de prueba:** Agregar Ciudadano Cubano.

Entrada	Resultados	Condiciones
En las Opciones de Menú ir a la opción Nuevo, seleccionar el Submenú Cubano y se llena el formulario.	Se inserta en la base de datos el nuevo ciudadano cubano.	

Tabla 4.2 Caso de prueba del caso de uso: GestionarCiudadanoCubano.

**Nombre del caso de prueba:** Modificar Ciudadano Cubano.

Entrada	Resultados	Condiciones
Después de mostrar el listado se seleccionó el ciudadano cubano a editar dando clic en el vinculo en su Carné de Identidad.	Se actualiza en la base de datos el ciudadano cubano y se regresa a la página donde se muestra el listado.	

Tabla 4.3 Caso de prueba del caso de uso: GestionarCiudadanoCubano.

**Nombre del caso de prueba:** Listar Ciudadanos Cubanos.

Entrada	Resultados	Condiciones
En las Opciones de Menú ir a la opción listado, seleccionar el Submenú Cubano.	El sistema muestra la lista de los ciudadanos cubanos existentes.	

Tabla 4.4 Caso de prueba del caso de uso: ListarCiudadanosCubanos.

**Nombre del caso de prueba:** Insertar Ciudadano Extranjero.

Entrada	Resultados	Condiciones
Después de Listado los ciudadanos extranjeros presionar el botón Agregar.	Se inserta en la base de datos el nuevo ciudadano extranjero y se regresa a la página donde se muestra el listado.	

Tabla 4.5 Caso de prueba del caso de uso: GestionarCiudadanoExtranjero.

**Nombre del caso de prueba:** Modificar Ciudadano Extranjero.

Entrada	Resultados	Condiciones
Después de mostrar el listado se seleccionó el ciudadano extranjero a editar dando clic en el vinculo en su Carné de Identidad.	Se actualiza en la base de datos el ciudadano extranjero y se regresa a la página donde se muestra el listado.	

Tabla 4.6 Caso de prueba del caso de uso: GestionarCiudadanoExtranjero.

**Nombre del caso de prueba:** Listar Ciudadanos Extranjeros.

Entrada	Resultados	Condiciones
En las Opciones de Menú ir a la opción listado, seleccionar el Submenú Extranjero.	El sistema muestra la lista de los ciudadanos extranjeros existentes.	

Tabla 4.7 Caso de prueba del caso de uso: ListarCiudadanosExtranjeros.



En el capítulo se mostraron los resultados de la etapa de construcción del sistema. En el proceso de implementación se cumplieron los principios de diseño y estándares de interfaz, elementos que ayudan a una mejor comunicación con el usuario, así como los estándares de implementación. Como culminación al diseño se presentó el modelo de implementación describiendo la distribución física del sistema y sus componentes y se realizaron las pruebas al sistema para evaluarlo y determinar la calidad del mismo.

### CONCLUSIONES

El Registro de Ciudadanos 2.0 se encuentra en proceso de liberación en la Empresa de Soluciones Informáticas (Softel), para posteriormente ser desplegado en Infomed. Partiendo de este hecho se puede concluir que:

- ❖ Se realizó un estudio detallado de la información referente a los ciudadanos cubanos y extranjeros, necesaria para el Sistema de Información para la Salud.
- ❖ Se modelaron los Flujos de Trabajo: Modelamiento del Negocio, Gestión de Requerimientos, Análisis, Diseño, Implementación y Pruebas; quedando completa la documentación técnica en el Expediente del Proyecto.
- ❖ Se implementó el mantenimiento al Registro de Ciudadanos actual, siguiendo las políticas de informatización del Sistema Nacional de Salud, la arquitectura definida por el MINSAP y la integración con otros componentes.

Por lo que, se cumplieron los objetivos del trabajo, logrando llevar a cabo un mantenimiento evolutivo y correctivo del Registro de Ciudadanos, que permite hacer búsquedas dinámicas, gestionar la información y brindar servicios a las demás aplicaciones integradas al SISalud.

### RECOMENDACIONES

Al concluir el presente trabajo y habiendo cumplido el objetivo propuesto, se recomienda:

- ❖ Desplegar el Registro de Ciudadanos 2.0 en el Sistema de Información para la Salud para su uso.
- ❖ Introducir la información de este registro, dada su importancia para las demás aplicaciones del Sistema de Información para la Salud.
- ❖ Implementar los niveles: Provincial, Municipal y Unidades de Salud, cada uno con los tipos de usuarios Editor y Visualizador correspondientes, así como el Visualizador Nacional.

### REFERENCIAS BIBLIOGRÁFICAS

[1] DELGADO RAMOS ARIEL, VIDAL LEDO MARÍA Informática en la salud pública cubana *Revista Cubana Salud Pública*, 2006.

[2] GONZÁLEZ VALDÉS DAMARIS, ALEMÁN SÁNCHEZ PEDRO CARLOS, & LAURA DÍAZ ORTEGA. *Impacto de la informatización en la sociedad y estomatología cubanas* La Habana, Facultad de Ciencias Médicas de La Habana, 2007.

[3] Ídem referencia 1.

[4] MARÍN DÍAZ, M. Consideraciones sobre el proyecto de informatización de la Atención Primaria de Salud *Revista Habanera de Ciencias Médicas*, 2004, 3.

[5] DOTRES, C. *Revista Cubana de Medicina Tropical*, 1997.

[6] INFOMED. Disponible en: [http://www.sld.cu/sistema\\_de\\_salud/aspectos.html](http://www.sld.cu/sistema_de_salud/aspectos.html)

[7] RAMÍREZ MÁRQUEZ ABELARDO, CASTELL-FLORIT SERRATE,& PASTOR MESA GUILLERMO El Sistema Nacional de Salud de Cuba, 2003.

[8] Ídem referencia 1.

[9] DELGADO RAMOS, A. Informatización Sistema Nacional de salud, 2003.

[10] TURRUELLES TEJEDA YOSVANI, GONZALEZ DIAZ MAIKEL *Registro de Ubicación Geográfica y Registro de Localidades para el Sistema de Información para la Salud*. Ciudad de La Habana, Universidad de las Ciencias Informáticas, 2007.

[11] DELGADO RAMOS ARIEL , CABRERA HERNÁNDEZ MIRNA ,& JUNCAL VIRGINIA Registro Informatizado de Salud (RIS) *Temas Estadísticos de Salud*, 2005.

[12] Ídem referencia 11.

[13] CABRERA HERNÁNDEZ, M. *Propuesta de Esquema del Sistema de Información para la Salud (SISalud)*. La Habana, SOFTEL, 2006.

[14] Ídem referencia 13.

[15] Ídem referencia 1.

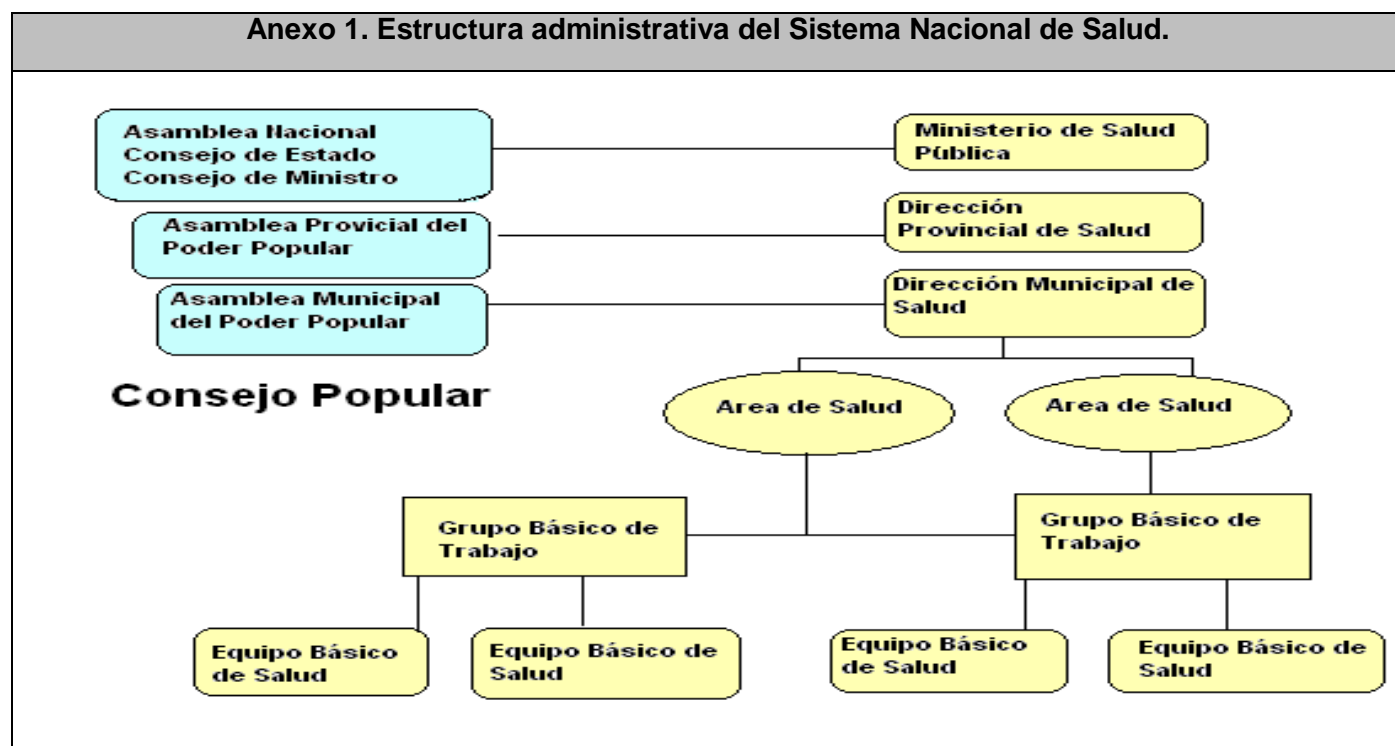
- [16] GNU. *Sitio Web de GNU*. Disponible en: <http://www.gnu.org/philosophy/free-sw.es.html>
- [17] GONZÁLEZ C., B. *XML Web Services*. *DesarrolloWeb.com*
- [18] REYNOSO CARLOS, B. *Introducción a la Arquitectura de Software* Universidad de Buenos Aires, 2004.
- [19] ESTR@TEGIAMAGAZINE. Disponible en: [www.e-estrategia.com](http://www.e-estrategia.com)
- [20] *Arquitectura de Software a emplear en los componentes del Sistema de Información para la Salud*. La Habana, SOFTEL, 2006.
- [21] FUENTES LIDIA, TROYA JOSÉ, &, VALLECILLO ANTONIO *Desarrollo de Software Basado en Componentes* Universidad de Málaga.
- [22] DACCACH T, J. SOA, 2007.
- [23] LEÓN GARCÉS , J. and Y. FROMETA MORENO. *Plataforma de Servicios PlaSer*. La Habana, Universidad de las Ciencias Informáticas, 2008.
- [24] WWW.CIBERAULA.COM. 2006. Disponible en: [http://linux.ciberaula.com/articulo/linux\\_apache\\_intro/](http://linux.ciberaula.com/articulo/linux_apache_intro/)
- [25] *ASP en Castellano*. Disponible en: [http://www.programacion.net/asp/articulo/aspnet\\_quees](http://www.programacion.net/asp/articulo/aspnet_quees)
- [26] *Tutorial de Perl en español*. Disponible en: <http://geneura.ugr.es/~jmerelo/tutoperl/>
- [27] E. VILLATE, J. *Introducción al XML*, 2001.
- [28] GUERVOS, J. M. *Introducción al lenguaje XML*.
- [29] *Extended Style Language* Disponible en: <http://www.ulpgc.es/otros/tutoriales/xml/XSL.html>
- [30] EGUÍLUZ PÉREZ , J. *Introducción a JavaScript*.
- [31] Ídem referencia 10.
- [32] *Pautas de diseño RIS*. La Habana, Proyecto APS – SOFTEL.
- [33] GÓMEZ VELÁZQUEZ, K. *APS\_Estándares\_Tratamiento\_Errores\_PHP\_BD*. La Habana, Proyecto APS – SOFTEL, 2005.

## BIBLIOGRAFÍA

1. *Arquitectura de Software a emplear en los componentes del Sistema de Información para la Salud. La Habana, SOFTEL, 2006.*
2. *ASP en Castellano.* Disponible en: [http://www.programacion.net/asp/articulo/aspnet\\_quees](http://www.programacion.net/asp/articulo/aspnet_quees)
3. CABRERA HERNÁNDEZ, M. *Propuesta de Esquema del Sistema de Información para la Salud (SISalud).* La Habana, SOFTEL, 2006.
4. DACCACH T, J. SOA, 2007.
5. DELGADO RAMOS, A. *Informatización Sistema Nacional de salud,* 2003.
6. DELGADO RAMOS ARIEL , CABRERA HERNÁNDEZ MIRNA ,&, JUNCAL VIRGINIA *Registro Informatizado de Salud (RIS) Temas Estadísticos de Salud,* 2005.
7. DELGADO RAMOS ARIEL, VIDAL LEDO MARÍA *Informática en la salud pública cubana Revista Cubana Salud Pública,* 2006.
8. DOTRES, C. *Revista Cubana de Medicina Tropical,* 1997.
9. E. VILLATE, J. *Introducción al XML,* 2001.
10. EGUÍLUZ PÉREZ , J. *Introducción a JavaScript.*
11. ESTRATEGIAMAGAZINE. Disponible en: [www.e-estrategia.com](http://www.e-estrategia.com)
12. *Extended Style Language.* Disponible en: <http://www.ulpgc.es/otros/tutoriales/xml/XSL.html>
13. FUENTES LIDIA, TROYA JOSÉ, &, VALLECILLO ANTONIO *Desarrollo de Software Basado en Componentes* Universidad de Málaga.
14. GNU. *sitio Web de GNU.* Disponible en: <http://www.gnu.org/philosophy/free-sw.es.html>
15. GÓMEZ VELÁZQUEZ, K. *APS\_Estándares\_Tratamiento\_Errores\_PHP\_BD.* La Habana, Proyecto APS – SOFTEL, 2005.
16. GONZÁLEZ C., B. *XML Web Services. DesarrolloWeb.com*

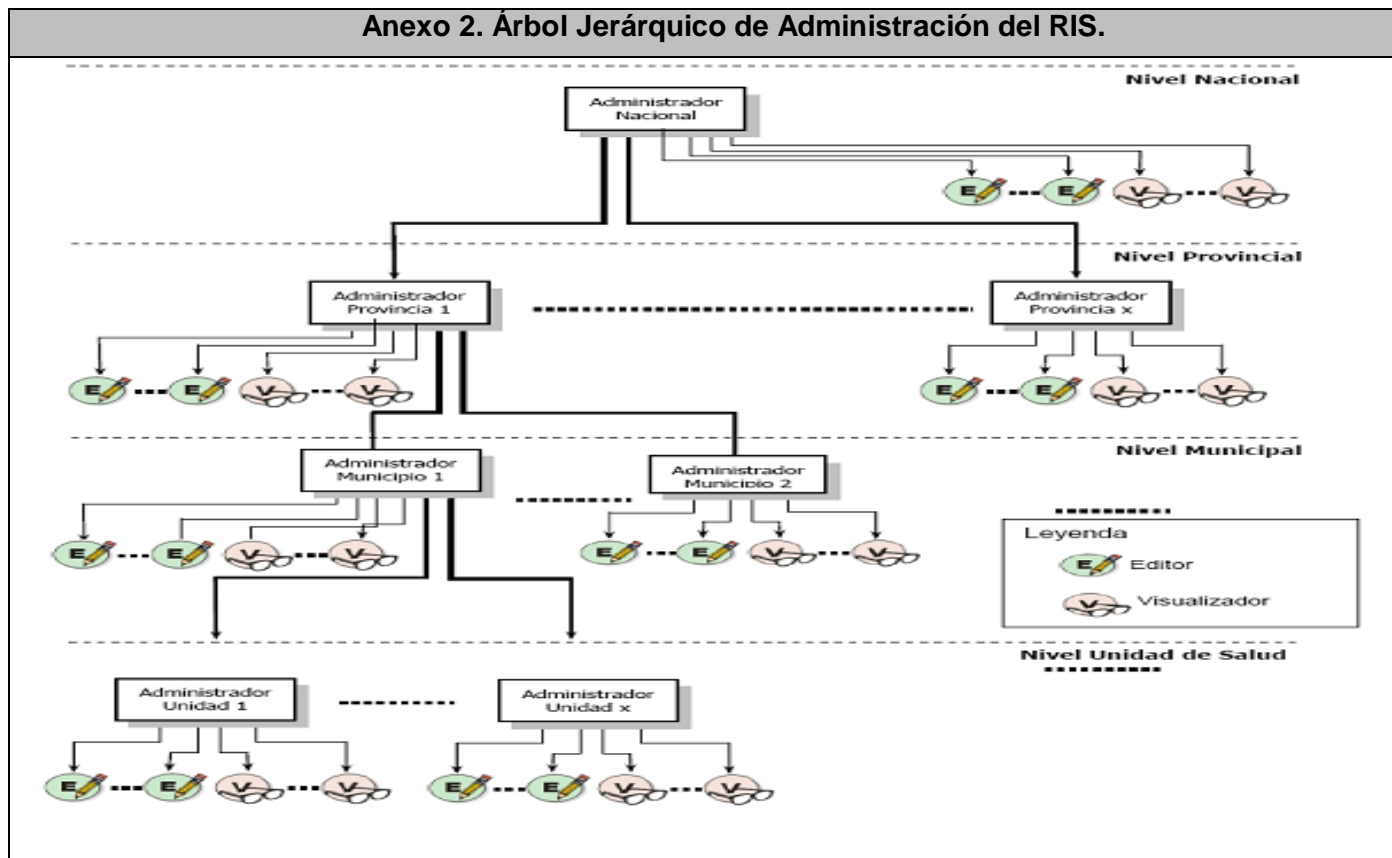
17. GONZÁLEZ VALDÉS DAMARIS, ALEMÁN SÁNCHEZ PEDRO CARLOS ,&, LAURA DÍAZ ORTEGA. *Impacto de la informatización en la sociedad y estomatología cubanas* La Habana, Facultad de Ciencias Médicas de La Habana, 2007.
18. GUERVOS, J. M. Introducción al lenguaje XML.
19. INFOMED. Disponible en: [http://www.sld.cu/sistema\\_de\\_salud/aspectos.html](http://www.sld.cu/sistema_de_salud/aspectos.html)
20. LEÓN GARCÉS, J. and Y. FROMETA MORENO. *Plataforma de Servicios PlaSer*. La Habana, Universidad de las Ciencias Informáticas, 2008.
21. MARÍN DÍAZ, M. Consideraciones sobre el proyecto de informatización de la Atención Primaria de Salud *Revista Habanera de Ciencias Médicas*, 2004, 3.
22. *Pautas de diseño RIS*. La Habana, Proyecto APS – SOFTEL.
23. RAMÍREZ MÁRQUEZ ABELARDO, CASTELL-FLORIT SERRATE, &, PASTOR MESA GUILLERMO *El Sistema Nacional de Salud de Cuba*, 2003.
24. REYNOSO CARLOS, B. *Introducción a la Arquitectura de Software* Universidad de Buenos Aires, 2004.
25. TURRUELLES TEJEDA YOSVANI, GONZALEZ DIAZ MAIKEL *Registro de Ubicación Geográfica y Registro de Localidades para el Sistema de Información para la Salud*. Ciudad de La Habana, Universidad de las Ciencias Informáticas, 2007.
26. *Tutorial de Perl en español*. Disponible en: <http://geneura.ugr.es/~jmerelo/tutoperl/>
27. WWW.CIBERAULA.COM. 2006. Disponible en: [http://linux.ciberaula.com/articulo/linux\\_apache\\_intro/](http://linux.ciberaula.com/articulo/linux_apache_intro/)

## ANEXOS

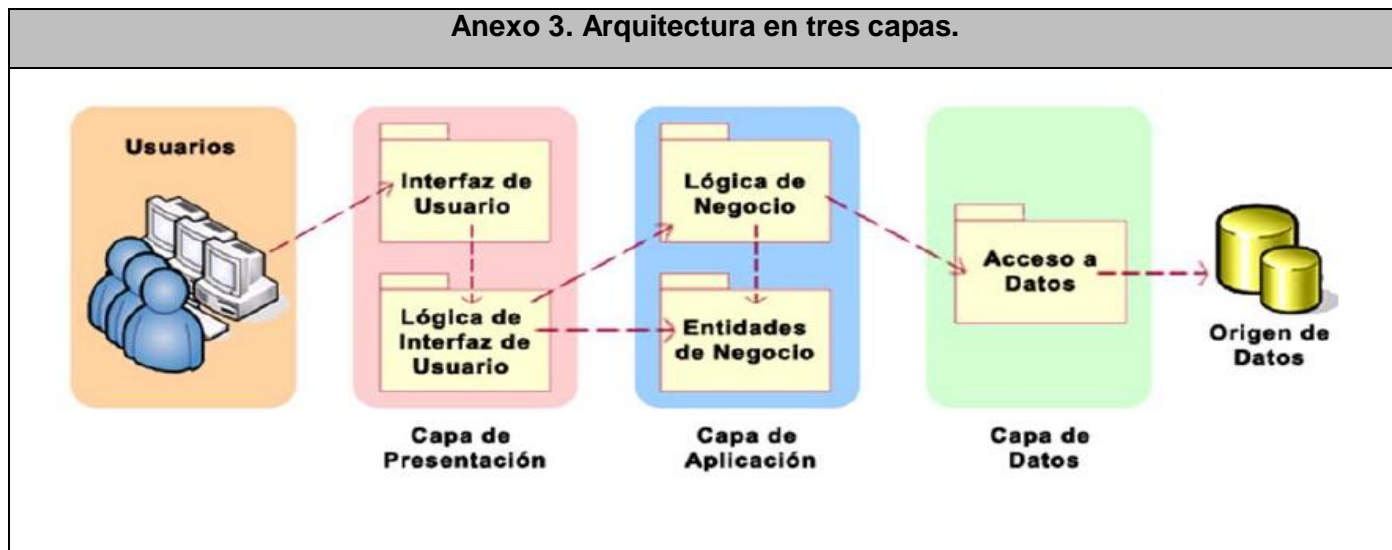




Anexo 2. Árbol Jerárquico de Administración del RIS.



Anexo 3. Arquitectura en tres capas.



## GLOSARIO DE TÉRMINOS

**APS:** Atención Primaria de Salud: Nivel asistencial que constituye la puerta de entrada del paciente al Sistema Nacional de Salud, donde debe darse solución alrededor del 90% de los problemas que afectan a la población. En este nivel se realizan acciones educativas, curativas y de Rehabilitación.

**Aplicación Web:** Es una aplicación informática que los usuarios utilizan accediendo a un servidor Web a través de un navegador o browser. Estas son muy populares debido a la habilidad para actualizar y mantener la información manipulada sin distribuir e instalar el software en miles de potenciales clientes.

**Arquitectura:** Conjunto de decisiones significativas acerca de la organización de un sistema de software, la selección de los elementos estructurales a partir de los cuales se componen el sistema. La misma se interesa no sólo por la estructura y el comportamiento, sino también por las restricciones y compromisos de uso, funcionalidad, funcionamiento, flexibilidad al cambio, reutilización, comprensión, economía y tecnología, así como por aspectos estéticos.

**Base de Datos:** Es un conjunto de datos que pertenecen al mismo contexto almacenados sistemáticamente para su posterior uso.

**Caso de Uso:** Descripción de un conjunto de secuencias de acciones, incluyendo variaciones, que un sistema lleva a cabo y que conduce a un resultado observable de interés para un actor determinado.

**Componente:** Parte física y reemplazable de un sistema que se ajusta a, y proporciona la realización de, un conjunto de interfaces.

**Concurrencia:** Ejecución simultánea de dos o más actividades durante el mismo intervalo de tiempo.

**Dependencia:** Relación semántica entre dos elementos, en la cual un cambio en uno puede afectar al otro.

**Diagrama:** Presentación gráfica de un conjunto de elementos y sus relaciones.

**Dominio:** Área de conocimiento o actividad caracterizada por un conjunto de conceptos y terminología comprendidos por los practicantes de ese dominio.

**INFOMED:** Red telemática del Sistema Nacional de Salud (SNS) de Cuba que funciona como una división del Centro Nacional de Información de Ciencias Médicas (CNICM), y parte de la existencia de una red nacional especializada de centros de información.

**Informática:** Es la disciplina que estudia el tratamiento automático de la información utilizando dispositivos electrónicos y sistemas computacionales.

**Informatizar:** Proceso de aplicar sistemas o equipos informáticos al tratamiento de la información.

**Metalinguaje:** Es un lenguaje usado para hacer referencia a otros lenguajes.

**MINSAP:** Ministerio de Salud Pública en Cuba.

**PDF (del inglés, Portable Document Format, Formato de Documento Portátil):** Es un formato de almacenamiento de documentos, desarrollado por la empresa Adobe Systems.

**Proyecto:** Esfuerzo de desarrollo para llevar un sistema a lo largo de un ciclo de vida.

**SISalud:** Sistema de Información para la Salud, plataforma que brinda servicios web al Sistema Nacional de Salud.

**SOFTTEL (Empresa de Soluciones Informáticas):** Entidad del Ministerio de Informática y las Comunicaciones (MIC).

**Servicio:** Unidad de software que encapsula alguna funcionalidad de negocio y proporciona estas a otros servicios a través de interfaces públicas bien definidas.

**Servicio Web:** Colección de protocolos y estándares que sirven para intercambiar datos entre aplicaciones.

**SOAP (del inglés, Simple Object Access Protocol, Protocolo de Acceso Simple a Objetos):** Estándar que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML. SOAP es uno de los protocolos utilizados en los Servicios Web.

**Software:** Conjunto de programas y procedimientos necesarios para hacer posible la realización de una tarea específica, en contraposición a los componentes físicos del sistema.

**Software Libre:** Es el software que, una vez obtenido, puede ser usado, copiado, estudiado, modificado y redistribuido libremente.

**Subsistema:** Agrupación de elementos, de los que algunos constituyen una especificación del comportamiento ofrecido por los elementos contenidos.

**UNIX:** Es un sistema operativo portable, multitarea y multiusuario.