



Universidad de las Ciencias Informáticas

Facultad 7

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

Componente de identificación centralizada para la solución alas PACS-RIS

Autores: Jorge Luis Díaz González

Ramón Ernesto García Ponce

Tutores: Ing. Arelys Rivero Castro

Ing. Yuniesky Armentero Moreno

La Habana, 26 de junio de 2013

“Año 55 de la Revolución”

DATOS DE CONTACTO

Tutores:

Ing. Arelys Rivero Castro

Graduada de Ingeniero en Ciencias Informáticas, egresada de la UCI en el año 2009. Posee la categoría docente de Instructor. Ha impartido las asignaturas de Sistemas Operativo y Seguridad Informática. Es profesora de la Facultad 7 donde realiza función de Asesora de Seguridad Informática. Actualmente se desempeña como Analista en el Departamento de Software Médico Imagenológico del Centro de Informática Médica de la Universidad de las Ciencias Informáticas. Correo electrónico: arcastro@uci.cu.

Ing. Yuniesky Armentero Moreno

Ingeniero en Ciencias Informáticas, graduado en la Universidad de las Ciencias Informáticas en el año 2007. Posee la categoría docente Instructor. Actualmente es profesor de Debate Histórico Cultural, ha impartido también Introducción a la Programación, Programación I, Gráfico por Computadoras y varios cursos optativos y del segundo perfil en la Facultad 6. Se desempeña como coordinador de PP4 y PP5 en el Departamento de Producción de Software Médico Imagenológico, CESIM. Correo electrónico: yarmentero@uci.cu

DEDICATORIA

A mi mamá y mi papá, por darme su inmenso amor, por ser ejemplos a seguir y por confiar siempre en mí. Por enseñarme que si se puede y ayudarme a seguir adelante. A Adrián Roldos por ser el mejor de mis amigos, por estar ahí conmigo en las buenas y en las malas, por tener tanta paciencia conmigo y brindarme su apoyo incondicional. Cuando muchos me fallaron tu siempre estuviste para ayudarme, gracias mi hermano.

A mi compañero de tesis por soportarme todo este tiempo y darme ánimos para continuar, gracias. A la profesora María de Jesús que cariñosamente todos llamamos Marichu por su ayuda, preocupación y exigencia. A la profesora Arelys que pienso que hubiera sido imposible la culminación de la tesis sin su ayuda. A mis compañeros de grupo por compartir estos 5 años de universidad como mi segunda familia, a los que vinieron y a los que no, Muchas Gracias!

Ernesto

A mi madre y mi padrino que me han apoyado a lo largo de mi carrera, la cual no hubiese podido terminar sin su guía.

A mi abuela que siempre me ha puesto el hombro para apoyarme en el en los más duros momentos y siempre me dio el apoyo necesario para no flaquear y lograr mis objetivos.

A mis hermanos Wilma y Juan Pablo que los quiero con la vida.

A mi novia Dory que tuvo la paciencia y comprensión para en todo momento brindarme su ayuda por difícil que fuera el problema. A todos aquellos que no mencioné y de una forma u otra me ayudaron.

Jorge Luis

AGRADECIMIENTOS

A todas esas personas que de una forma u otra ayudaron en nuestra formación profesional a lo largo de estos 5 años. En particular a Adrián el cual nos dio su ayuda incondicional, también a Pedro, Ariel, Manuel, Raidel y a todos aquellos que nos han acompañado en esta carrera, a todos los del apartamento y el aula por compartir los buenos y malos momentos.

A la Revolución y a la Universidad de las Ciencias Informáticas por darnos la posibilidad de forjarnos como hombres de bien.

A los amigos, los más cercanos, y los que están lejos también, gracias.

¡A todos Muchas Gracias!

RESUMEN

Al incrementarse el alcance de la tecnología, el cuidado de la información se ha tornado crucial para los hombres, las organizaciones y las sociedades. La seguridad de la información se ha convertido en un elemento clave en las empresas de todas las ramas de la sociedad.

En la Universidad de las Ciencias informáticas se desarrolló la solución alas PACS-RIS, la cual está compuesta por un grupo de sistemas que se encargan de la gestión de la información asociada a los pacientes, los estudios y las imágenes médicas. Algunos sistemas como el alas RIS realizan el proceso de identificación de forma independiente y otros como el alas PACSViewer no controlan el acceso de los usuarios a los recursos disponibles, representando una vulnerabilidad de seguridad.

Con la presente investigación se tiene como objetivo desarrollar un componente para gestionar de forma centralizada del proceso de identificación de los usuarios en la solución alas PACS-RIS. Este componente ofrece un único punto de identificación para todos los sistemas que conforman la solución, eliminando la reiteración del código de identificación en cada una de las aplicaciones. Posibilita que el usuario que interactúa con los sistemas solo tenga que autenticarse una vez, evitando retrasos en el proceso de atención al paciente. Para el desarrollo del componente de identificación se utilizó como IDE de desarrollo Microsoft Visual Studio 2012. Como sistema gestor de base de datos PostgreSQL 8.4 y para el proceso de modelado Enterprise Architect 7.5.

Palabras clave: alas PACS-RIS, centralización, control de acceso, identificación, seguridad

ÍNDICE

INTRODUCCIÓN	1
CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA SOBRE EL PROCESO DE IDENTIFICACIÓN CENTRALIZADA.....	3
1.1. Control de acceso.....	3
1.2. Servicio web	3
1.3. Sistemas de Almacenamiento y Transmisión de Imágenes Digitales (PACS).....	4
1.1. Estándar DICOM	5
1.2. Sistema de Información Radiológica (RIS).....	5
1.3. Integrating the Healthcare Enterprise (IHE)	6
1.4. Arquitecturas, protocolos y estándares de autenticación	8
1.5. Sistemas Internacionales que realizan identificación centralizada	12
1.6. Sistemas nacionales que realizan identificación centralizada	14
1.7. Lenguaje Unificado de Modelado.....	17
1.8. Lenguaje de programación	18
1.9. Tecnologías, herramientas y metodologías.....	18
1.10. Modelo de calidad, lenguaje y notación de modelado	20
1.11. Conclusiones.....	20
CAPÍTULO 2. CARACTERÍSTICAS DEL COMPONENTE DE IDENTIFICACIÓN CENTRALIZADA.....	22
2.1. Propuesta del sistema	22
2.2. Modelo de Dominio.....	23
2.1. Diagrama del Modelo de Dominio.....	Error! Bookmark not defined.
2.2. Entidades y Conceptos fundamentales.....	24
2.3. Requisitos Funcionales.....	25
2.4. Requisitos no Funcionales.....	27
2.5. Definición de los actores del sistema	29
2.6. Diagrama de Casos de Uso del Sistema.....	30
2.7. Descripción de los Casos de Uso del Sistema.....	31
2.8. Conclusiones	32

CAPÍTULO 3. DISEÑO E IMPLEMENTACIÓN DEL COMPONENTE DE IDENTIFICACIÓN CENTRALIZADA.....	33
3.1. Diseño	33
3.2. Patrones de diseño.....	33
3.3. Modelo arquitectónico.....	34
3.4. Modelo de Datos.....	36
3.5. Descripción de los componentes del modelo de datos.....	37
3.6. Web Services Description Language (WSDL).....	37
3.7. Diagrama de componentes	38
3.8. Modelo de despliegue.....	39
3.9. Conclusiones	40
CONCLUSIONES	41
RECOMENDACIONES.....	42
REFERENCIAS BIBLIOGRÁFICAS.....	43
BIBLIOGRAFÍA.....	47
ANEXOS	53
Anexo 1. Descripción ampliada de los Casos de Uso del Sistema	53
Anexo 2. Diagramas de clases	59
Anexo 3. Diagramas de Secuencia.....	63
Anexo 4. WSDL del servicio web.....	67

INTRODUCCIÓN

Con el paso del tiempo, la investigación y la aplicación técnica de los conocimientos científicos se han desarrollado a un ritmo tan acelerado que han transformado radicalmente la vida de los seres humanos en todas las esferas de la sociedad. En el campo de la salud, la evolución de los sistemas de diagnóstico por imagen ha sido vertiginosa, la aparición del ultrasonido y después de la resonancia magnética son claras muestras de una evolución permanente (1).

Con el desarrollo de las tecnologías y la ciencia de la computación, se ha logrado obtener imágenes de altísima calidad de zonas del cuerpo que serían inaccesibles sin invasión del organismo. El número de pruebas radiológicas de un paciente es muy variable y la manera de almacenar sus imágenes asociadas está cambiando gracias al uso de las Tecnologías de la Información y las Comunicaciones (TICs).

Para facilitar la gestión de las imágenes de los departamentos de diagnóstico por imágenes surgen los sistemas de Almacenamiento y Transmisión de Imágenes Digitales (PACS, por sus siglas en inglés). Son sistemas que le brindan al personal médico una amplia gama de herramientas y funcionalidades para la visualización y procesamiento de las mismas. Existen diversos fabricantes de equipos médicos y para lograr una estandarización en las imágenes que producen de forma tal que un mismo PACS pueda gestionarlas eficientemente surge el estándar de Imagen Digital y Comunicación en Medicina (DICOM, por sus siglas en inglés). El estándar describe el formato de archivos y la especificación de los datos de un paciente en la imagen brindando un lenguaje común a distintos sistemas médicos. De esta forma las imágenes pueden venir acompañadas de mediciones, cálculos e información descriptiva relevante para diagnósticos (2).

Para complementar el trabajo que realizan los PACS se utilizan los Sistemas de Información Radiológica (RIS, por sus siglas en inglés). Estos sistemas gestionan la información administrativa de los departamentos de diagnóstico por imágenes permitiendo el registro de pacientes y sus citas para estudios o consultas de imagenología. También permiten realizar búsquedas por pacientes y diagnósticos médicos facilitando la realización de estudios de morbilidad (3).

La existencia de diversos sistemas para la gestión de la información de los pacientes en los departamentos de diagnóstico por imagen, hace necesario que se comuniquen entre sí para evitar redundancia de la información y al final del proceso brindar mejor atención al paciente. La Iniciativa de Integración de Empresas de Salud (IHE, por sus siglas en inglés) pretende que todos los sistemas de información que intervienen en el

proceso sanitario compartan información de forma eficaz y transparente (4). Para ello cuenta con perfiles de integración que describen una necesidad clínica de integración de sistemas y la solución para llevarla a cabo. Define también los componentes funcionales, los que se denominan actores IHE, y especifica con el mayor grado de detalle posible las transacciones propias para cada actor, basadas siempre en estándares como DICOM y Health Level Seven (HL7) (5).

Para las compañías que se dedican al desarrollo de las aplicaciones como los PACS y los RIS la protección de la información que manejan sus sistemas es de vital importancia. En estos sistemas se maneja información referente al estado de salud de los pacientes por lo que solo el personal autorizado debe de tener acceso a esa información y las instituciones cubanas no son la excepción. En la Universidad de las Ciencias Informáticas (UCI) el Centro de Informática Médica (CESIM), se dedica al desarrollo de aplicaciones informáticas para el área de salud. El centro cuenta con departamentos especializados en diferentes niveles y áreas de atención, tales como el departamento de Software Médico Imagenológico (SWMI), donde se desarrolla la solución alas PACS-RIS. Esta solución está formada por un grupo de módulos, subsistemas o sistemas de mediana o alta complejidad como el sistema alas PACSViewer, alas PACSReporter y el alas RIS.

En la solución alas PACS-RIS existen un grupo de operaciones que no son controladas según el acceso de cada usuario:

- Las secretarias, actualmente pueden reservar, reprogramar y cancelar citas para cualquier departamento de diagnóstico por imágenes.
- Los especialistas, pueden visualizar imágenes de cualquier modalidad y crear informes independientemente del servicio al que pertenezcan.
- Existen actualmente médicos residentes que tienen la posibilidad de aprobar informes, a esta operación solo deberían tener acceso solo los médicos especialistas en radiología.

El proceso de identificación dentro de la solución alas PACS-RIS no se hace de forma centralizada. Los sistemas alas PACSViewer y alas PACSServer en la actualidad no realizan ningún proceso de identificación de usuario, siendo esto una debilidad en la seguridad de los sistemas. Los únicos que realizan el proceso de identificación son los sistemas alas PACSReporter y alas RIS los cuales cuentan con roles definidos.

Debido a que cada sistema gestiona la identificación de forma independiente, entonces una persona tiene que recordar más de un usuario y contraseña lo que figura un problema para los especialistas que son de avanzada

edad que se les puede dificultar este procedimiento. El básico conocimiento en informática de los usuarios y la existencia de normativas de seguridad para la protección de contraseñas (Longitud de ocho caracteres, que contenga letras mayúsculas, números y caracteres especiales) influyen en errores de autenticación provocando una pérdida de tiempo, elemento que puede ser determinante en la institución donde se encuentra implantado el sistema.

Por lo que se plantea como **problema a resolver**: ¿Cómo controlar el proceso de identificación en la solución alas PACS-RIS?

Este problema se enmarca en el **objeto de estudio**: El proceso de control de acceso. Delimitando el **campo de acción** en: El proceso de identificación en la solución alas PACS-RIS.

Para dar solución al problema se plantea como **objetivo general**: Desarrollar un componente que controle de forma centralizada el proceso de identificación en la solución alas PACS-RIS.

Para dar cumplimiento al objetivo planteado se proponen las siguientes **tareas investigativas**:

- Realizar un análisis crítico y valorativo de los sistemas informáticos que realizan identificación de manera centralizada.
- Generar los artefactos que corresponden al flujo de trabajo: Modelamiento del negocio, Requerimientos, Análisis y Diseño e Implementación.
- Realizar un estudio de estándares, protocolos y arquitecturas relacionadas a la identificación centralizada.
- Desarrollar un componente informático siguiendo lo establecido en la especificación de Requisitos de Software.

Los métodos científicos utilizados en el trabajo de investigación son:

Histórico Lógico: para realizar un análisis crítico valorativo de la información contenida en las fuentes bibliográficas consultadas, con el objetivo de conocer cuáles son las tendencias y como se ha comportado el desarrollo de software en torno al proceso de identificación centralizada en los sistemas informáticos. Como parte inicial de la investigación se realiza un estudio del estado del arte de la problemática planteada. Se analizan diversas soluciones existentes y se identifican fuentes de datos confiables.

Analítico Sintético: se hace uso del mismo para descomponer el problema de la investigación en elementos concretos de la solución. Se analizan los casos de forma particular.

Inductivo Deductivo: teniendo en cuenta el funcionamiento del proceso de identificación en los sistemas informáticos y específicamente en los sistemas PACS y RIS, se evalúa la problemática existente y de esta forma se determinan los aspectos particulares para desarrollar una propuesta de solución.

Modelación: con el fin de crear abstracciones e interpretar la realidad. Se utiliza al crear modelos de funcionamiento de los flujos de trabajos de los sistemas PACS y RIS y para la confección de diagramas que posibilitaron un mejor entendimiento de los procesos desarrollados.

Beneficios

La puesta en funcionamientos del componente de identificación centralizada como parte de la solución alas PACS-RIS, propicia un conjunto de beneficios:

- Ofrecer un único punto de identificación para todos los sistemas que conforman la solución, posibilitando que el usuario solo tenga que autenticarse una vez.
- Evitar la reiteración del código de identificación en cada una de las aplicaciones que conforman la solución.
- Optimizar los flujos de trabajos de los sistemas y con ello se agilizar el acceso a la información de los pacientes.

Estructura de la Tesis

Capítulo 1: Fundamentación teórica sobre el proceso de identificación centralizada. Aborda los diferentes elementos y características que forman la base teórica conceptual para el desarrollo de un componente de seguridad que permite la identificación centralizada en la solución alas PACS-RIS. Se

presentan los principales antecedentes existentes sobre el tema, sistemas que realizan identificación centraliza, metodologías, herramientas y tecnologías que se utilizaron en la investigación.

Capítulo 2: Características del componente de identificación centralizada. Describe los procesos que se relacionan en el entorno donde se encuentra funcional la solución alas PACS-RIS, sus requisitos funcionales y no funcionales. Se ofrece una propuesta del componente para la identificación centralizada del proceso de identificación de los usuarios de la solución alas PACS-RIS y se exponen las características del mismo.

Capítulo 3: Diseño e implementación del componente de identificación centralizada. Aborda las fases de diseño e implementación del software. Se muestran los diagramas de clases del diseño y los diagramas de secuencia. Se define el modelo arquitectónico utilizado para el desarrollo del componente para la identificación centralizada y se describen los diagramas de componente, despliegue y el modelo de datos.

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA SOBRE EL PROCESO DE IDENTIFICACIÓN CENTRALIZADA

En este capítulo se presenta una breve descripción de los sistemas que conforman la solución alas PACS-RIS y los resultados obtenidos en la investigación de los sistemas tanto nacionales como internacionales que realizan identificación centralizada. Se presentan conceptos de interés para la investigación y se fundamentan las herramientas y tecnologías que se utilizarán para el desarrollo de la misma.

1.1. Control de acceso

Es el proceso de conceder permisos a usuarios o grupos para acceder a objetos tales como ficheros o documentos. El control de acceso se basa en tres conceptos fundamentales: identificación, autenticación y autorización. Con el uso de estos tres principios un administrador del sistema puede controlar qué recursos están disponibles para los usuarios de un sistema (6).

La identificación es el proceso a través del cual un usuario muestra su identidad, presenta un conjunto de datos o cualidades que lo caracterizan. La autenticación, consiste en la confirmación de la identidad de un usuario; es decir, la garantía para cada una de las partes de que su emisor es realmente quien dice ser (7). La autorización es el proceso donde se comprueba que los usuarios con identidad válida, solo tengan acceso a aquellos recursos sobre los cuales se le asignaron privilegios (8).

1.2. Servicio web

Un servicio web es un conjunto de aplicaciones o tecnologías con capacidad de interactuar en la web. Estas tecnologías permiten la relación entre dos sistemas informáticos, de manera que se pueda llevar a cabo un intercambio de información y transacciones entre dichos sistemas independientemente de la plataforma en que se hayan desarrollado cada uno. (9)

El potencial de que un programa informático pueda comunicarse prácticamente con cualquier otro programa, en cualquier momento y lugar, es desde luego el cambio más revolucionario introducido por los servicios web. Los servicios web son independientes de la tecnología utilizada para su suministro y utilización, lo cual facilita su integración. (9)

1.3. Sistemas de Almacenamiento y Transmisión de Imágenes Digitales (PACS)

En los hospitales existen actualmente diversos equipos de adquisición de imágenes médicas que generan un elevado volumen de información. La administración de dichos datos se hace engorrosa, principalmente durante el proceso de impresión y almacenamiento. Para la gestión de las imágenes médicas dentro del entorno de trabajo de los departamentos de imagenología se utilizan los sistemas PACS (10). Estos sistemas están compuestos por cuatro componentes fundamentales, los equipos de adquisición de imágenes médicas, las estaciones de visualización, los servidores de almacenamiento y la infraestructura de red que es capaz de interconectar los elementos anteriormente mencionados. En las estaciones de visualización se brindan un grupo de técnicas digitales para mejorar la calidad de la imagen. Estas funcionalidades incluyen aplicar brillo y contraste a la imagen, ampliar y resaltar regiones de interés, así como realizar un conjunto de transformaciones espaciales, posibilitando de esta manera un aumento de la capacidad de diagnóstico (11). Utilizar el sistema PACS posibilita la reducción del costo por la utilización de las placas radiológicas o por el daño o la pérdida de información y mejora la accesibilidad de las imágenes debido a que las agrupa en una misma base de datos.

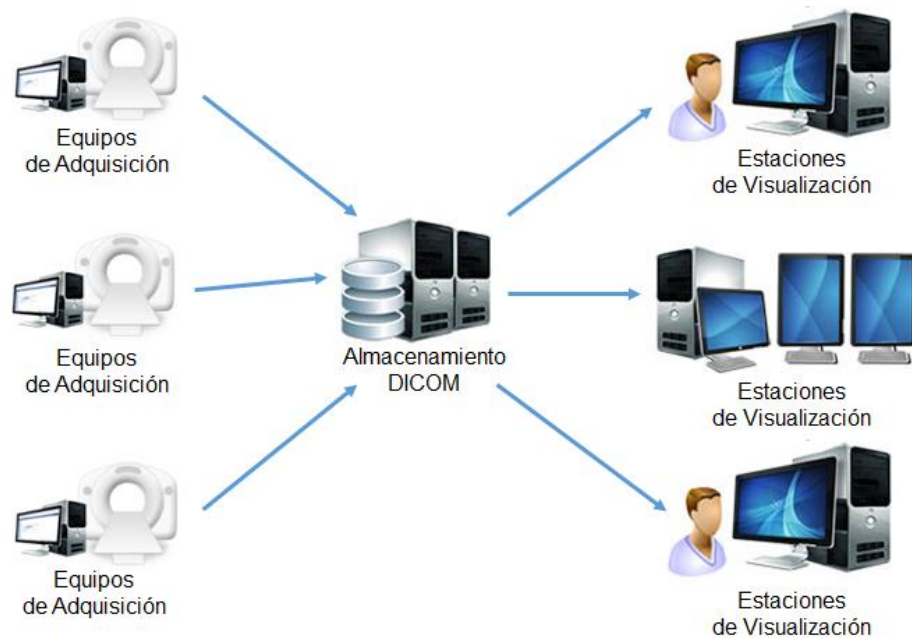


Figura 1. Componentes básicos de un PACS (fuente: creada por los autores)

1.1. Estándar DICOM

Con la introducción de la tomografía computarizada, seguida de otras modalidades de imágenes digitales de diagnóstico en la década de 1970, y el creciente uso de las computadoras en las aplicaciones clínicas, el Colegio Americano de Radiología y el National Electrical Manufacturers Association (NEMA por sus siglas en inglés) reconoció la emergente necesidad de un método estándar para la transferencia de imágenes y la información asociada entre dispositivos fabricados por diferentes proveedores. Estos dispositivos producen una variedad de formatos de imágenes digitales (12). Por lo que surge en 1993 la versión real del estándar DICOM.

El estándar permite que distintos sistemas puedan interactuar en conjunto, mediante la especificación de normas para la manipulación, el almacenamiento, la impresión y la transmisión de información digital en la imagenología médica. Los equipos médicos en conformidad con DICOM podrán generar imágenes con la capacidad de ser archivadas y visualizadas en un mismo PACS, sin importar la modalidad o el fabricante al cual pertenezcan (12).

1.2. Sistema de Información Radiológica (RIS)

Un RIS tiene la responsabilidad de gestionar la actividad clínica y administrativa de los departamentos de diagnóstico por imágenes, manejar la información demográfica de los pacientes, programar las citas y la entrega de reportes de diagnóstico. Este sistema informatiza toda la actividad radiológica de un paciente, desde la petición del estudio, al informe del mismo, pasando por la recogida de las incidencias y consumos de materiales que conlleva la realización de dicha exploración (13). Permite realizar búsquedas por pacientes, estudios y diagnósticos médicos, proporcionando al personal que lo utiliza una mejor organización en la información que se maneja.

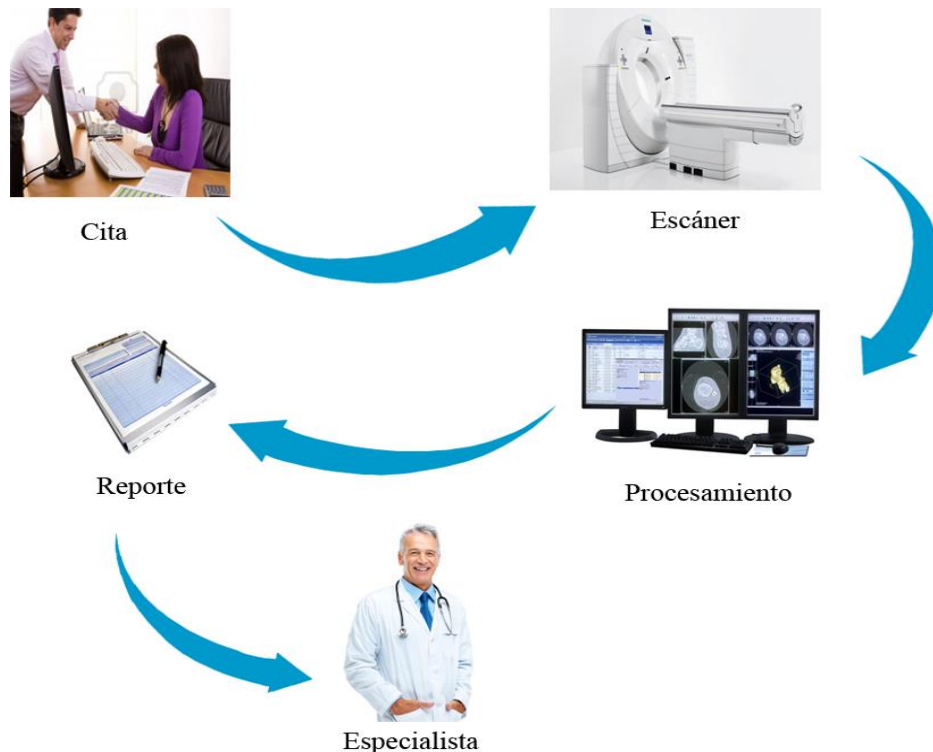


Figura 2. Flujo de trabajo del RIS (fuente: creada por los autores)

1.3. Integrating the Healthcare Enterprise (IHE)

IHE se creó en 1998 por parte de usuarios y empresas de Estados Unidos, es una iniciativa de profesionales de la salud y empresas proveedoras de equipos y software médicos, cuyo objetivo es mejorar la comunicación entre los sistemas de información que se utilizan en la atención al paciente. La iniciativa IHE trabaja en definir y documentar soluciones basadas en estándares para resolver necesidades operacionales y clínicas de integración. No es un nuevo estándar que los equipos deban cumplir para realizar determinada función, y tampoco viene a sustituir a ninguno de estos estándares; por lo que se considera un marco de implementación, una armazón sustentada por los estándares existentes, pero con la forma de documento de consenso acerca de cómo pensar, discutir y solucionar satisfactoriamente los problemas de integración de la forma más eficiente y económica, los denominados "Marcos Técnicos IHE". Brindando así la oportunidad de demostrar que los sistemas pueden operar eficazmente bajo estos estándares y que pueden comunicarse con los de otros fabricantes sin necesidad de desarrollar otros diferentes para cada instalación (14).

1.3.1. ¿Qué es un perfil de integración?

Es una descripción precisa de cómo debe implementarse un estándar para cubrir una necesidad determinada de integración clínica. Cada perfil de integración incluye definiciones de casos de uso clínicos, información clínica que se intercambia y el flujo de trabajo a tener en cuenta, así como el conjunto de Actores y Transacciones que cubrirían dicha necesidad. Los perfiles de integración incluyen también con gran detalle las especificaciones de integración que están definidas en el Marco Técnico IHE y las descripciones de productos de manera que puedan ser utilizadas en las propuestas de adquisición.

1.3.2. ¿Cuáles son los contenidos de un perfil de integración?

Un determinado perfil de integración definirá actividades significativas en el contexto del flujo de trabajo del “caso de uso” relacionado. La transmisión de la información se describe en términos de “transacciones” entre “actores”. Un actor se implementa como parte de una aplicación informática. Todas las transacciones entre actores requeridas para completar el flujo de trabajo son especificadas claramente. Las especificaciones describen cómo se van a utilizar partes específicas de los estándares y proporcionan pautas técnicas para la implementación de las aplicaciones.

1.3.3. ¿Qué beneficios tienen los usuarios al utilizar los perfiles de IHE?

Los perfiles de integración proporcionan una descripción de las funcionalidades básicas que un usuario puede esperar de una determinada aplicación. Los perfiles de integración pueden, por lo tanto ser usados como un punto de partida para el desarrollo de propuestas de adquisición. IHE abre una vía para actualizar sistemas ya existentes, cuando es necesario e integrarlos con nuevas aplicaciones de un proveedor diferente.

1.3.4. Perfiles referentes al tema de identificación

Autenticación de Usuarios Empresariales (EUA, por sus siglas en inglés): la utilización de este perfil facilita la gestión de la autenticación centralizada y proporciona a los usuarios la comodidad y la velocidad de un inicio de sesión único. Este perfil define un método para establecer un único identificador por usuario que le permita acceder de forma transparente a cualquier sistema que participa en este perfil de integración. EUA hace uso del protocolo Kerberos durante el proceso de autenticación de usuarios, mejorando el flujo de trabajo.

Cross-Enterprise User Profile aserción (XUA): apoya a las empresas que han optado por tener su propio directorio de usuario con su método único de autenticación de los usuarios, así como otras que han determinado utilizar a un tercero para realizar la autenticación (15).

1.4. Arquitecturas, protocolos y estándares de autenticación

1.4.1. *Arquitectura Single Sign-On (SSO)*

Se refiere al acceso a múltiples recursos por medio de un único proceso de ingreso. El principal objetivo de la arquitectura Single Sign-On es transferir la funcionalidad y complejidad de todos los componentes de seguridad a un solo servicio, siendo éste el único punto de autenticación y registro en el sistema a pesar de que continúan interactuando con múltiples componentes de seguridad en el sistema. Una implementación real de SSO, deberá contar con un agente SSO que se encarga de almacenar en una base de datos o directorio protegido los passwords que le permiten al usuario acceder a cada una de las aplicaciones o servicios, en el momento que lo desee, porque el proceso de autenticación se realiza de manera transparente para el usuario, una vez que éste ha sido autenticado por medio de la arquitectura SSO (16).

Las diferentes arquitecturas SSO están compuestas por tres componentes básicos:

- **Interface:** el modo en que el SSO interactúa con una determinada aplicación. Usualmente reside en el cliente, y es conocido como Agente SSO.
- **Administración:** el mecanismo que permite configurar, mantener y monitorear el proceso de SSO.
- **Credenciales:** cada aplicación a la que se accede requiere información confidencial (nombre de usuario, contraseña, etc.), que agrupada recibe el nombre de credenciales. Las credenciales deben almacenarse de manera protegida para que sea únicamente el agente SSO quien pueda acceder a ellas (16).

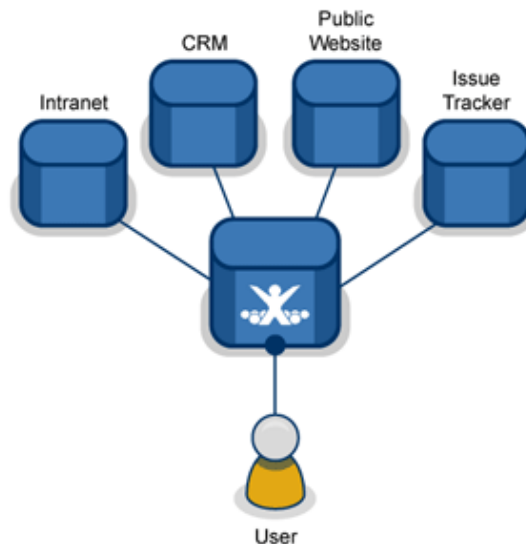


Figura 6. Single Sing-On (17)

1.4.2. Kerberos

Es un protocolo de seguridad creado en el Massachusetts Institute of Technology (MIT) como una solución a los problemas de seguridad de la red, usa una criptografía de claves simétricas para validar usuarios con los servicios de red evitando tener que enviar contraseñas a través de la red. Al validar los usuarios para los servicios de la red por medio de Kerberos, los intentos de usuarios no autorizados de interceptar contraseñas en la red quedan frustrados (18).

Es un método de autenticación que sigue la arquitectura Single-Sign-On (SSO) basada en el paso de tokens, llamados tickets. Heimdal Kerberos es una implementación de la versión 5, y fue desarrollada intencionalmente fuera de los EEUU para sortear las regulaciones de exportación y por eso puede incluirse en versiones no comerciales de UNIX (18).

La autenticación según el método Kerberos funciona de la siguiente manera:

- El usuario se valida a un servidor de autenticación Kerberos que le devuelve una clave de sesión, es un ticket general de comunicación con el servidor de autenticación.
- Cada vez que el cliente quiere acceder a un recurso, el servidor de autenticación genera un ticket para el recurso determinado.
- El servidor del recurso comprueba que el ticket enviado por el cliente es válido y permite el acceso.

Ventajas de Kerberos

- Autenticación mutua: el cliente puede validar la identidad de la entidad de seguridad del servidor y el servidor puede validar al cliente.
- Vales de autenticación seguros: solo se utilizan vales cifrados y las contraseñas nunca están incluidas en el vale.
- Autenticación integrada: una vez que el usuario haya iniciado sesión, no necesitará iniciar sesión nuevamente para tener acceso a cualquiera de los servicios que admite la autenticación Kerberos, siempre y cuando el vale del cliente no haya expirado. Cada vale tiene una vigencia, que está determinada por las directivas del dominio Kerberos que genera el vale (19).

Desventajas de Kerberos

- Para que una aplicación use Kerberos, el código de las aplicaciones debe ser modificado para hacer las llamadas apropiadas a las librerías de Kerberos. Para algunas aplicaciones, esto puede suponer un esfuerzo excesivo para los programadores. Para otras aplicaciones incompatibles, los cambios se deben realizar en el protocolo usado entre el servidor de red y sus clientes. Por defecto, las aplicaciones de código cerrado que no tienen soporte de Kerberos son usualmente las más problemáticas.
- Si se decide utilizar Kerberos en la red y se transmite cualquier contraseña a un servicio que para autenticar no hace uso de este protocolo, se corre el riesgo de que el paquete pueda ser interceptado. Para asegurar la red con Kerberos, se debe kerberizar (hacer trabajar con Kerberos) todas las aplicaciones que mandan las contraseñas en texto plano o parar el uso de estas aplicaciones en la red (20).

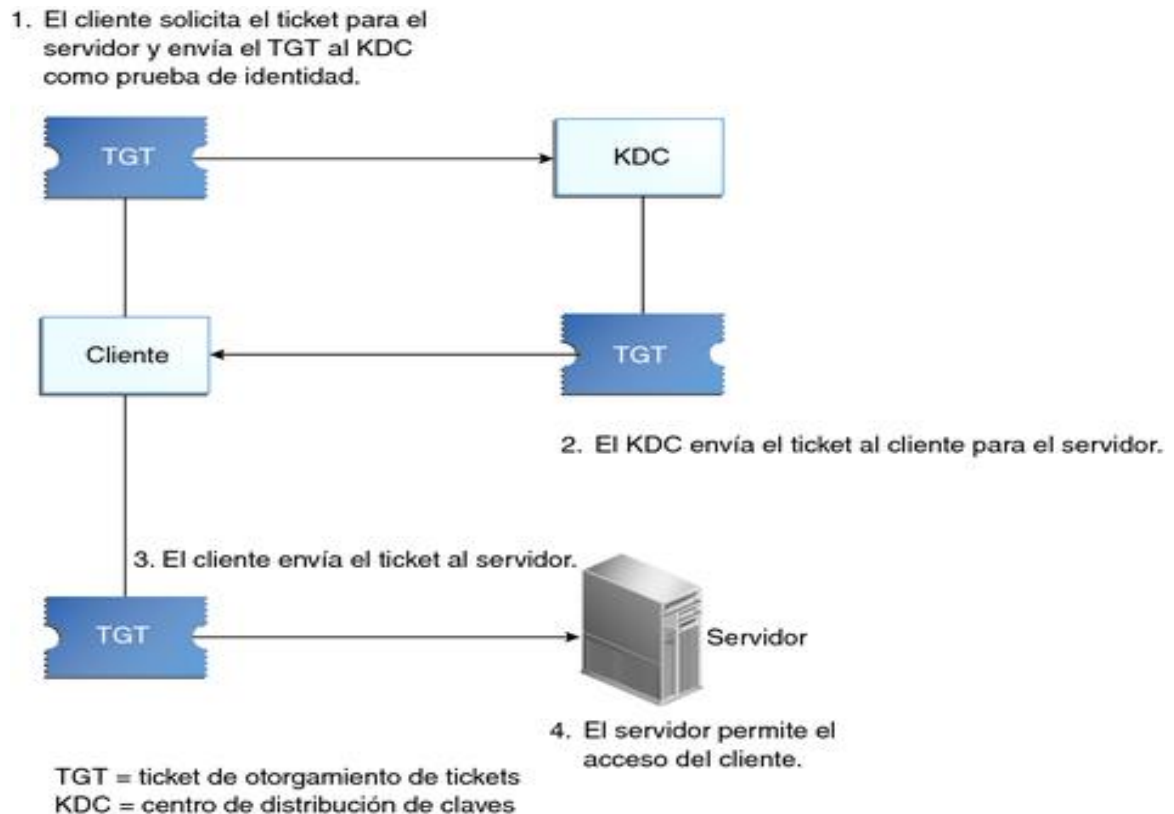


Figura 3. Diagrama del flujo de información de Kerberos (21)

1.4.3. OpenID

Es un estándar para la identificación digital descentralizada. Los usuarios se registran en una página web y pueden verificar el estado de conexión desde cualquier servidor que sea compatible. Los distintos sitios web que tienen soporte para OpenID no requieren que el usuario cree una nueva identidad. En lugar de registrarse utilizan un **idP** (proveedor de identidad). Se trata de un identificador que crea el propio servidor y facilita la autenticación de OpenID. A diferencia de otros servicios y arquitecturas del tipo Single Sign-On no tiene un mecanismo de autenticación específica. La seguridad de la conexión depende de la confianza en el proveedor de identidad (22).

Ventajas de utilizar OpenID

- Con un único nombre de usuario podrás acceder a cientos de sitios webs, sin recordar decenas de claves de acceso y usuarios diferentes.

- También es un sistema de gestión de sus datos, de forma que al autenticarte en un sitio web te preguntará que datos quieres compartir con este sitio donde te vas a registrar.
- Es un estándar abierto y libre que te permite confiar en un servidor que aloje el servidor de autenticación o incluso montarte uno propio de tu máxima confianza, pues lo controlas y gestionas tú mismo (23).

Desventajas de OpenID

- **Es un protocolo muy vulnerable al phishing.** El phishing consiste en la suplantación maliciosa de una página de autenticación con el objetivo de conseguir el identificador y la contraseña de un usuario. Habitualmente los sitios maliciosos tienen que engañar al usuario para que llegue a la página suplantada. En un servidor que utilice OpenID ese acceso es más sencillo de conseguir, basta con ofrecer al usuario la autenticación mediante OpenID para acceder a una aplicación determinada que puede ser falsa, y después redirigir al usuario a una página que suplanta la página del proveedor de OpenID para conseguir su contraseña (24).

1.5. Sistemas Internacionales que realizan identificación centralizada

1.5.1. PasswordBank Single Sign-On

PasswordBank es una institución dedicada al desarrollo y comercialización de un producto de identificación centralizada diseñado para solucionar múltiples problemas de identificación y gestión de accesos. La solución PasswordBank Single Sign-On permite a los empleados de la empresa donde se encuentre este sistema acceder a todos los recursos (web, internos y aplicaciones Java) usando solo el nombre de usuario y la contraseña que utilizan para su autenticación inicial en Windows. De manera que un sistema centralizado y administrable de forma remota, pueda identificar a dicho usuario, autenticarle, cifrar ese acceso, y finalmente autorizarle a usar dicha aplicación. Los usuarios finales solo necesitan recordar una contraseña, lo que posibilita una mejora en su experiencia, reduciendo costes, impulsando la productividad y mejorando la seguridad. Este sistema está basado en la plataforma MONO, puede ser usado para proporcionar interoperabilidad segura entre los sistemas operativos Windows, Mac y Linux (25).

Las empresas que utilizan esta solución tienen beneficios como:

- Elevado grado de la seguridad y confidencialidad de la información relacionada con la identidad digital.
- Disponen de un directorio corporativo con una estructura adecuada y coherente que organice las identidades digitales.
- Gestionan de una manera eficiente la identidad digital de los empleados públicos y en un futuro de la ciudadanía.

Este sistema resuelve los problemas de identificación centralizada, pero a pesar de ser multiplataforma no deja de ser una solución comerciable, tiene su sede central en California Estados Unidos (25) por lo que la adquisición de este producto se vería afectada por las regulaciones que impone el bloqueo económico de ese país contra Cuba.

1.5.2. *Centricity PACS*

Es una solución desarrollada sobre la plataforma web para el entorno sanitario. Cumple con los estándares de integración de IHE lo cual le permite la integración con otros sistemas de información hospitalaria mejorando el flujo de información y permitiendo un inicio de sección centralizado (26). Utiliza protocolos web de cifrado para la protección de los datos de las instituciones donde se encuentra instalado y posee una interfaz flexible que se puede integrar con todas las empresas que utilicen Single Sign On.

La solución de control de acceso que utiliza Centricity PACS es viable, pero implica adquirir el sistema PACS completo, lo cual involucra un gasto innecesario para Cuba que lleva años desarrollando y perfeccionando el sistema alas PACS.

1.5.3. *Imprivata OneSign*

Este sistema realiza el proceso de autenticación utilizando la arquitectura “single sign-on” para crear un único punto de gestión de acceso y distribuir la información para todas las aplicaciones y estaciones de trabajo (27). Utiliza la tecnología de reconocimiento facial para identificar a los usuarios autorizados, cuando el usuario se aleja de la estación de trabajo el sistema cierra la sesión actual, si el usuario autorizado vuelve, la sesión se desbloquea automáticamente. Los técnicos o radiólogos no tienen que estar pendientes de bloquear o cerrar la sesión en las estaciones de trabajo y los datos del paciente están protegidos contra el acceso inapropiado (28).

Para hacer uso de la solución que propone Imprivata OneSign es necesario adquirir tecnología que permita el reconocimiento facial e instalarlo en cada una de las estaciones clientes que conforman la solución alas PACS-RIS, esa tecnología es costosa y al estar actualmente la solución alas PACS-RIS instalada en varias instituciones sanitarias encarecería el costo de dicha solución.

1.6. Sistemas nacionales que realizan identificación centralizada

1.6.1. Sistema de autenticación y autorización centralizado

El sistema se desarrolló en la UCI en el ambiente de las aplicaciones y servicios con que cuenta la infraestructura de la corporación PDVSA. Permite al usuario autenticarse una sola vez, además proporciona un mecanismo de autenticación que permite el acceso a los usuarios a múltiples aplicaciones, así como también establecer un componente de seguridad y un alto nivel de confiabilidad a los usuarios, a las aplicaciones y a los servicios que se ejecutan en el sistema.

Una vez registrado el usuario y después de haberse validado el registro y contraseña del mismo, el sistema le da un token de seguridad y es almacenado junto con las credenciales del usuario. El sistema devuelve a la aplicación el token de seguridad junto con el rol o roles que se le fueron asignados al usuario para así saber hasta donde el usuario tiene los permisos necesarios para hacer uso de los servicios que brinda la aplicación (29).

El componente de autenticación y autorización centralizado cuenta con una lista de las aplicaciones que pueden hacer uso de él. Las aplicaciones de la corporación PDVSA envían un identificador y el componente verifica si se encuentra registrado, de ser así le permite a la aplicación consumir de los servicios que brinda el mismo. Debido a esto para que una aplicación que no pertenezca a la corporación PDVSA pueda consumir de los servicios del componente se tienen que realizar modificaciones en el código del servicio.

1.6.2. Propuesta de procesos de Autenticación, Autorización y Auditoría (AAA) para aplicaciones basadas en servicios web XML

En la tesis de maestría realizada en La Habana el 20 de noviembre del 2010 por el ingeniero Karel Gómez Velázquez se propone un sistema para aplicaciones basadas en servicios web xml. En esta propuesta el producto de software proporciona una gestión centralizada de usuarios, asignación de roles y privilegios de acceso a todos los sistemas externos que consuman sus servicios. Brinda un óptimo proceso de trazabilidad y auditoría, de manera que se lleve un control estricto de las operaciones en que se involucran los usuarios

de los sistemas. Además con la publicación de estos servicios web xml se evita que se realicen acciones innecesarias fuera de cada negocio y se logra la reutilización de código.

1.6.3. *SISalud*

SISalud es una nueva estrategia del MINSAP alineada al proceso de informatización de la sociedad cubana, poniendo como centro de la misma al paciente y utilizando para su construcción las Tecnologías de la Información y las Comunicaciones (TIC) para el desarrollo de aplicaciones y componentes con un alto nivel de integración e interoperabilidad. SISalud es más que un sistema, es una plataforma de aplicaciones que se basa en una arquitectura orientada a servicios, abierta y con una interfaz de programación que posibilita incorporar nuevos módulos compatibles entre sí, permitiendo la integración de componentes. Este sistema conforma su estructura a partir de los derechos del usuario que se autentica, dándole solamente la posibilidad de acceso a los módulos o aplicaciones que este requiera (30).

Entre las ventajas de SISalud se encuentran:

- Es una solución fácil de mantener, actualizar y soportar técnicamente, porque la aplicación es centralizada.
- Permite la integración con todos los procesos relacionados con la informatización de la sociedad cubana.
- Facilita a todos los niveles de salud (Primario, Secundario y Terciario) la información estandarizada y homogénea.



Figura 4. Estructura de SISalud (30)

La propuesta del sistema de autenticación, autorización y auditoría para aplicaciones web y SISalud realizan identificación centralizada y proporcionan un mecanismo que permite el acceso de los usuarios a múltiples aplicaciones, pero solo creadas en la plataforma web.

1.6.4. Acaxia

Es un sistema que gestiona la seguridad de forma centralizada en un entorno de varias aplicaciones, fortaleciendo una parte muy importante de la arquitectura de seguridad como es la protección de la información almacenada en los sistemas que utilizan sus servicios disminuyendo así el tiempo de desarrollo de las aplicaciones, el costo total de los proyectos y los riesgos que puedan ser aprovechados para realizar un ataque. Acaxia está desarrollado sobre software libre y brinda sus servicios a todos los sistemas que se suscriben a él. A diferencia de la mayoría de las aplicaciones existentes, incorpora procesos tan importantes como la administración de conexiones y perfiles. Además la solución incorpora la gestión dinámica del multi-tema, multi-idioma, multi-entidad, multi-escritorio y multi-sistema. De esta forma el usuario tendrá acceso a un conjunto de sistemas en una o varias entidades garantizándose siempre la compartimentación de la información, además cada usuario solo podrá gestionar la información a la que tenga acceso (31).

En la seguridad de aplicaciones juegan un papel fundamental los procesos de autenticación y autorización, porque permiten un mejor control en el acceso a la información. La mayoría de los sistemas solo incorporan estos dos procesos y en el caso de Acaxia implementa cinco procesos para convertirse en una solución integral.



Figura 5. Procesos de la seguridad (31)

Este sistema realiza la identificación centralizada en aplicaciones desarrolladas en diferentes plataformas, pero no se rige por los perfiles de IHE los cuales plantean un método para establecer un único identificador por usuario que le permita acceder de forma transparente a cualquier sistema que participa en esta integración. Por lo que se reafirma la necesidad de crear un sistema propio que realice las mismas prestaciones pero incorporando los perfiles IHE.

1.7. Lenguaje Unificado de Modelado

El lenguaje para el modelamiento unificado (UML, por sus siglas en inglés) se utiliza para la especificación, visualización, construcción y documentación de los artefactos de un proceso de sistema intensivo. Fue originalmente concebido por la corporación Rational Software y ha ganado un significativo soporte de varias organizaciones (32).

Características más importantes de UML:

- El lenguaje para modelamiento unificado le permite a los creadores de sistemas generar diseños que capturen sus ideas en una forma convencional y fácil de comprender para comunicarlas a otras personas.
- Su necesidad radica principalmente en el hecho de que es un lenguaje que permite desarrollar un plan bien analizado que pueda ser comprensible tanto como para el cliente como para los o el realizador explicarlo, analizarlo y desarrollarlo (33).

1.8. Lenguaje de programación

Como lenguaje de programación se utilizó C# 4.0 que es un lenguaje orientado a objetos y permite al desarrollador crear una amplia gama de aplicaciones que se ejecuten en .NET Framework, su sintaxis es sencilla, fácil de aprender y permite los conceptos de encapsulación, herencia y polimorfismo.

1.9. Tecnologías, herramientas y metodologías

Las siguientes herramientas son las utilizadas en el proyecto de Software Médico Imagenológico por su gran flexibilidad y robustez en cuanto al desarrollo de software se refiere. Estas presentan entre sus ventajas una gran variedad de funcionalidades, las cuales hacen que el trabajo en el proyecto sea mucho más rápido y eficiente.

1.9.1. *Windows Communication Foundation (WCF)*

1.9.1.1. *¿Qué es Windows Communication Foundation?*

Es un marco de trabajo para la creación de aplicaciones orientadas a servicios. Con WCF, es posible enviar datos como mensajes asincrónicos de un extremo del servicio a otro. Un extremo del servicio puede formar parte de un servicio disponible continuamente hospedado por Internet Information Service (IIS), o puede ser un servicio hospedado en una aplicación. Un extremo puede ser un cliente que solicita datos del servicio. Los mensajes pueden ser tan simples como un carácter o una palabra que se envía como XML, o tan complejos como una secuencia de datos binarios (34).

1.9.1.2. *Principales características de Windows Communication Foundation*

Orientación a servicios: el uso de los estándares de WCF permite crear aplicaciones orientadas a servicios. SOA, la arquitectura orientada a servicios posibilita el uso de servicios web para enviar y recibir datos. Los

servicios tienen la ventaja general de estar débilmente acoplados entre una aplicación y otra en lugar de incluidos en el código. Una relación de acoplamiento débil implica que cualquier cliente creado en cualquier plataforma puede conectar con cualquier servicio siempre y cuando se cumplan los contratos esenciales. (34)

Interoperabilidad: WCF implementa estándares modernos del sector para la interoperabilidad de servicios web.

Contratos de datos: dado que WCF se basa en .NET Framework, también incluye métodos con código sencillo para proporcionar los contratos que un usuario desee aplicar. Uno de los tipos de contrato universales es el contrato de datos. Básicamente, mientras se escribe el código del servicio usando Visual C# o Visual Basic, la forma más sencilla de controlar los datos consiste en crear clases que representan una entidad de datos con propiedades que pertenecen a la misma. WCF incluye un completo sistema para trabajar con datos. Cuando se han creado las clases que representan los datos, el servicio genera automáticamente los metadatos que permiten a los clientes ajustarse a los tipos de datos que se han diseñado.

Seguridad: es posible cifrar los mensajes para proteger la privacidad, así como obligar a los usuarios a que se autenticuen antes de permitirles recibir mensajes. (34)

1.9.1.3. *Ventaja de utilizar WCF ante ASP.NET Web services*

- WCF soporta más estándares de WS-* que los propios de los servicios web
- WCF admite varios enlaces HTTP, TCP, MSMQ, WS-HTTP, etc. Mientras que los servicios web sólo son compatible con HTTP
- WCF puede mantener transacciones como COM+
- Tienen integración con JSON
- Pueden alojarse en IIS, WAS, servicios Windows o estar auto alojado (35)

1.9.2. *Enterprise Architect 7.5*

Enterprise Architect es una herramienta que ofrece salida de documentación de alta calidad. Posee un entorno fácil de usar, rápido y flexible. Permite la importación de las clases del sistema para la creación de diagramas ahorrando un tiempo considerable (36) .Por estas características es que se decide utilizar como herramienta de modelado.

1.9.3. PostgreSQL 8.4

Es el sistema de gestión de bases de datos de código abierto más potente del mercado y en sus últimas versiones está a la altura de las mejores bases de datos comerciales (37). Es rápido, confiable y muy potente. Es muy versátil por lo que se considera idóneo para el proyecto.

1.9.4. Visual Studio 2012

Como entorno de desarrollo integrado (IDE) se ha seleccionado a Microsoft Visual Studio. Este IDE permite llevar a cabo tareas de modelado, codificado, prueba o depuración, sin salir de su entorno, brindando alta eficiencia y productividad (38). Es la versión más novedosa del mercado por lo que brinda tecnología de avanzada y múltiples funcionalidades.

Estas herramientas permiten un desarrollo fluido y robusto al componente de identificación centralizada, garantizan que el componente sea compatible con el resto de las aplicaciones que se han desarrollado en el departamento debido a que se utilizaron las mismas.

1.10. Modelo de calidad, lenguaje y notación de modelado

1.10.1. Capability Maturity Model Integration CMMI

El Modelo de Madurez de la Capacidad de Integración (CMMI, por sus siglas en inglés) es un enfoque de mejora de procesos que provee a las organizaciones de los elementos esenciales para un proceso efectivo. Fue desarrollado por el SEI (Software Engineering Institute). Mide la madurez del desarrollo del software en una escala del 1 al 5. Integra disciplinas como sistemas y software en un solo marco de trabajo. Describe formas efectivas y probadas de hacer las cosas, no es un enfoque radical (39). En la UCI, el Centro de Informática Médica perteneciente a la facultad 7 certificó el nivel dos de CMMI, lo cual significa un gran paso de avance en la profesionalidad del centro.

1.11. Conclusiones

A partir del estudio realizado sobre las características fundamentales de algunos sistemas informáticos que realizan el proceso de identificación centralizada, se concluyó que ninguno de ellos es factible para ser utilizado en la solución alas PACS-RIS. Se determina que es necesario desarrollar un componente que facilite el proceso de identificación, guiado por los elementos establecidos por IHE en el perfil de autenticación EUA; permitiendo que los usuarios de los sistemas alas PACS y alas RIS se autenticen una sola vez y de esta

forma agilizar el proceso de atención a los pacientes. Se realizó un análisis de las tecnologías y herramientas necesarias para llevar a cabo dicho desarrollo y se determinó que es viable la utilización de las ya establecidas por el departamento SWMI.

CAPÍTULO 2. CARACTERÍSTICAS DEL COMPONENTE DE IDENTIFICACIÓN CENTRALIZADA

En este capítulo a partir de los resultados obtenidos de la investigación previa, se realiza un análisis de las características del componente de identificación centralizada donde es necesario la mención del concepto de modelo de dominio, requisito funcional y no funcional. También se define el diseño del sistema, se realiza la propuesta del mismo y se describe el funcionamiento del componente.

2.1. Propuesta del sistema

Después de haber realizado un estudio del proceso de identificación de los sistemas que conforman la solución alas PACS-RIS se propone desarrollar un componente de seguridad que garantice la identificación centralizada de manera eficiente.

Cuando un usuario intenta acceder a una de las aplicaciones que conforma la solución alas PACS-RIS, esta envía una petición al componente de identificación centralizada con la siguiente estructura (id, usuario, contraseña). Dicho id se corresponde con un identificador registrado previamente en el componente y se utiliza para corroborar que la aplicación que hizo la solicitud pertenece a la solución alas PACS-RIS. La información viaja cifrada. El componente verifica que el usuario a través del cual se realizó la solicitud sea válido y crea un token de seguridad con la información asociada a este, con la estructura (rol, servicio, ip cifrado). Se da respuesta a la aplicación enviando el token de seguridad. Dicho token es utilizado por la aplicación para permitir el acceso a los recursos que posee.

Este componente para la solución alas PACS-RIS tendrá un mecanismo de identificación que permita el acceso a los sistemas que conforman la solución. Los usuarios que trabajen con la solución se autenticarán en el sistema una sola vez, elemento que facilitará el trabajo, constituirá un ahorro de tiempo considerable, fortaleciendo el esquema de seguridad y la interacción con la solución.

2.2. Modelo de Dominio

Debido a que los procesos del negocio no se encuentran bien definidos, se propone construir un modelo de dominio, el cual proporciona una vista del sistema donde se describen las entidades implicadas en la identificación de los usuarios que interactúan con la solución alas PACS-RIS.

Es una representación visual de clases conceptuales o de objetos reales en un dominio de interés. Consiste en un conjunto de diagramas, sin definición de operaciones. El propósito principal de este modelo es generar una terminología común y sentar las bases del entendimiento del desarrollo y no para definir el sistema completo. Para ver la descripción de los componentes que conforman el modelo de dominio ver la Tabla 1.

2.1. Diagrama del Modelo de Dominio

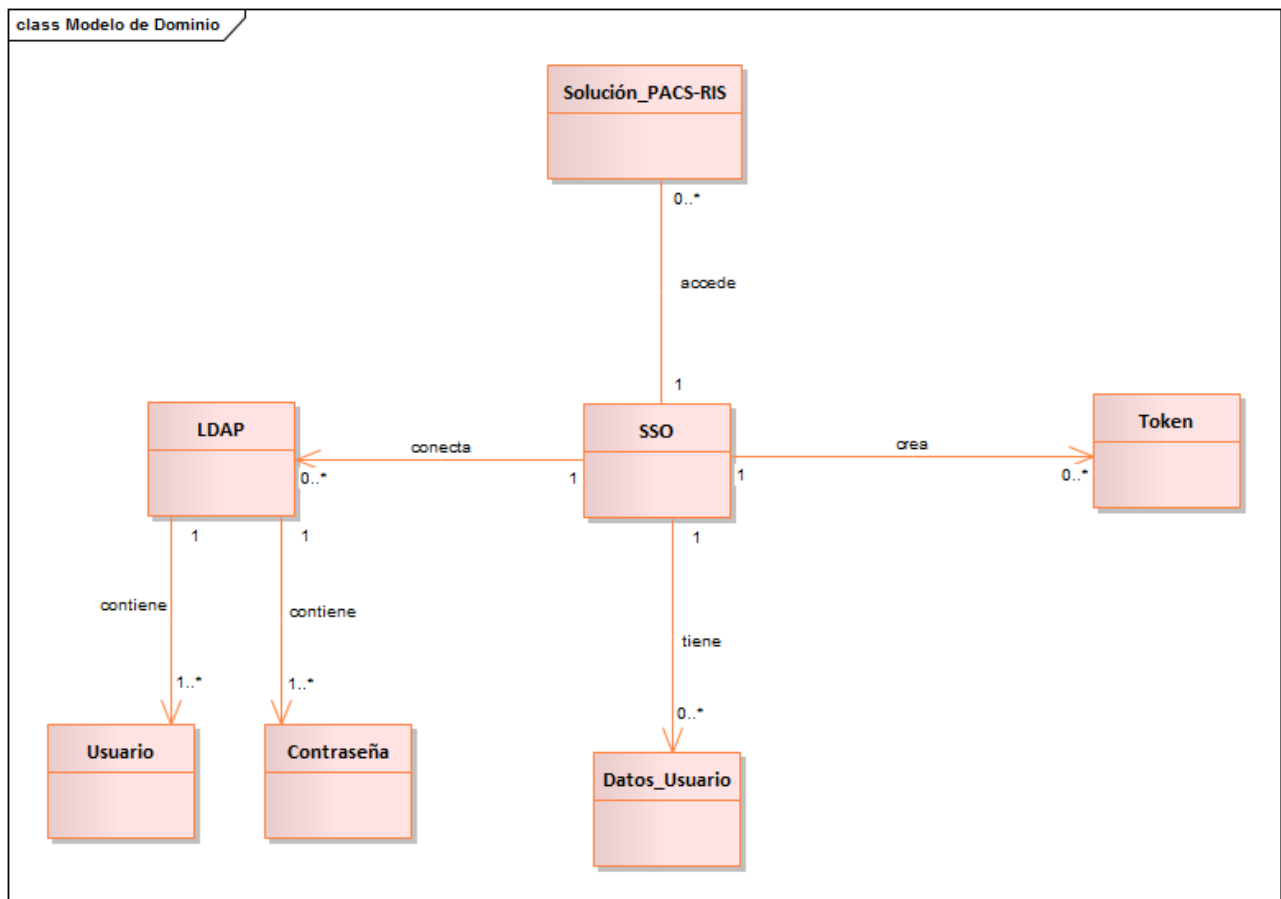


Figura 7. Modelo de Dominio (fuente: creada por los autores)

2.2. Entidades y Conceptos fundamentales

Entidades y Conceptos	Descripción
Solución_PACS-RIS	Está conformada por las aplicaciones que interactúan con el componente de identificación centralizada intercambiando información de los usuarios
LDAP	Servicio de dominio que se utiliza para acceder a información referente a usuarios y contraseñas y otras entidades ofreciendo una amplia capacidad de filtrado y mecanismos de seguridad
SSO	Permite el múltiple acceso a las aplicaciones mediante una sola autenticación
Token	Es una clave de sesión que se le envía a las aplicaciones mediante la cual el sistema sabe los permisos y privilegios de cada usuario
Usuario	Se encuentra identificado en el sistema y puede hacer uso del mismo
Contraseña	Conjunto de caracteres que le permitirá al usuario identificarse
Datos_Usuario	Son los datos que identifican a los usuarios

Tabla 1. Entidades y conceptos fundamentales (fuente: creada por los autores)

2.3. Requisitos Funcionales

Los requisitos funcionales son capacidades o condiciones que el sistema debe de cumplir, estos especifican las acciones que el sistema debe de realizar. Deben de ser sencillos para que puedan ser comprendidos por los clientes y desarrolladores (40). Deben de tener una sola interpretación y cumplir con las pautas establecidas en el departamento SWMI.

Requisito	Descripción
RF 1 Validar los datos de los usuarios	Verificar que los datos de los usuarios son los que corresponden y no tienen errores
RF 2 Insertar usuario	Insertar los datos de un nuevo usuario
RF 3 Modificar usuario	Modificar los datos de un usuario ya registrado en la base de datos
RF 4 Eliminar usuario	Eliminar los datos de un usuario existente
RF 5 Permitir la autenticación de los usuarios una sola vez	Posibilitar al usuario autenticarse una sola vez en el sistema, una vez que este lo hace en una de las aplicaciones, el componente debe ser capaz de gestionar la identidad del usuario sin necesidad de que este recuerde varias contraseñas o teclee sus datos en reiteradas ocasiones
RF 6 Cifrar los datos del usuario	Cifrar la información que se guarda en la base de datos para la protección de los datos de los usuarios
RF 7 Crear un token de seguridad	Crear un token de seguridad con los elementos necesarios que permitan el acceso a las aplicaciones
RF 8 Insertar rol	Insertar los roles de los usuarios

RF 9 Eliminar rol	Eliminar los roles existentes en la base de datos
-------------------	---

Tabla 2. Requisitos funcionales del componente de identificación centralizada (fuente: creada por los autores)

En la Figura 8 se muestra el diagrama de requisitos funcionales agrupados por paquetes lógicos.

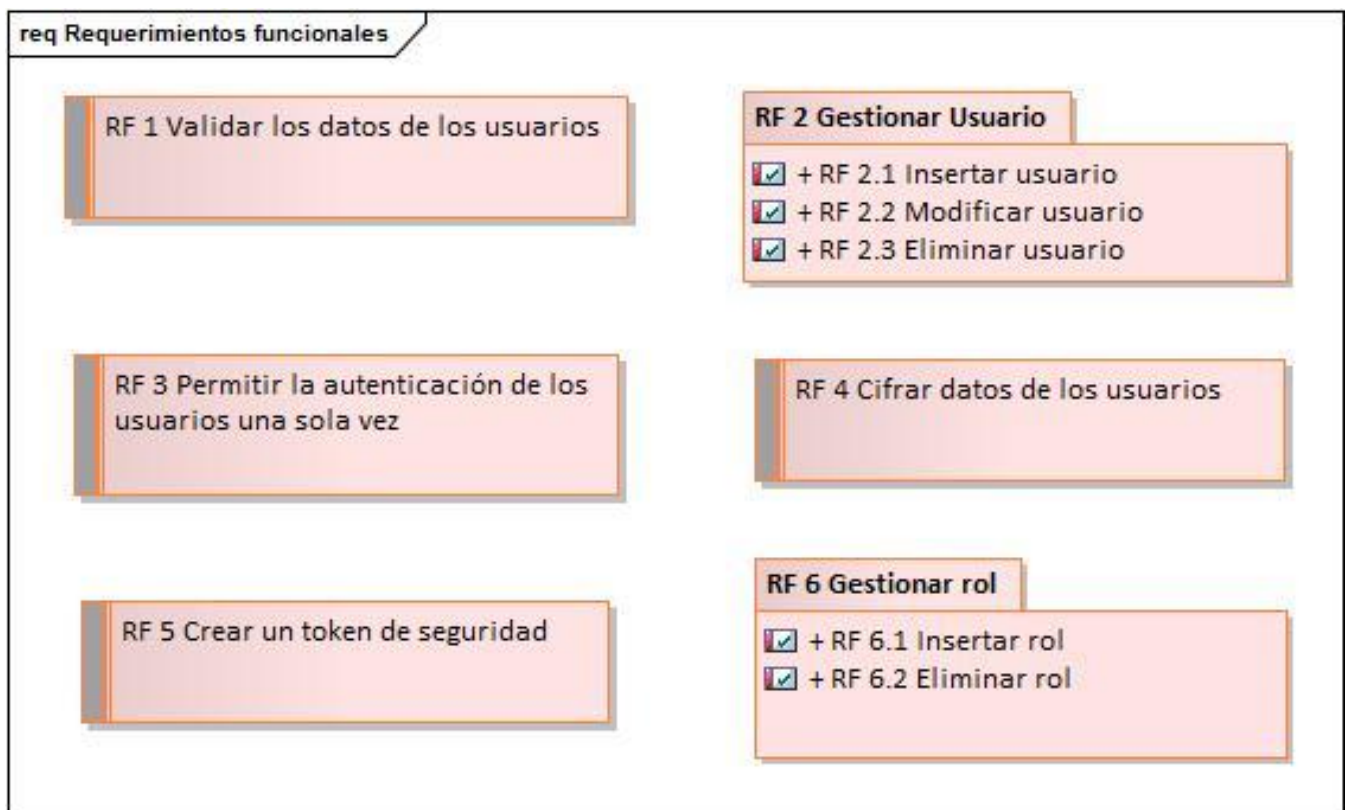


Figura 8. Requisitos funcionales del componente de identificación centralizada (fuente: creada por los autores)

2.4. Requisitos no Funcionales

Los requisitos no funcionales son propiedades o condiciones que el sistema debe de tener. Estas propiedades son las que hacen al producto usable, rápido y confiable.

Requisito	Descripción
RNFO 1. CPU Dual Pentium IV 3.0GHz	Para el correcto funcionamiento del componente, se necesita un CPU Dual Pentium IV 3.0GHz, 2 GB de memoria RAM, 250 GB de disco duro como mínimo y se recomienda una tarjeta de red Gigabit Ethernet NIC
RNFO 2. Memoria RAM de 2GB	
RNFO 3. Gigabit Ethernet NIC	
RNFO 4. 250 GB de disco duro	
RNFO 5. Windows XP SP3 o superior	El componente debe de instalarse sobre el sistema operativo Windows XP SP3 o superior
RNFO 6. Eficiente interacción entre las aplicaciones y el componente	El componente de identificación centralizada debe de ser capaz de brindar respuesta a las peticiones de las aplicaciones en el menor tiempo posible
RNFO 7. Navegador superior o compatible con Internet Explorer 8 o Mozilla Firefox 10	Para interactuar con la interfaz de administración del componente es necesario utilizar un navegador web
RNDI 1. lenguaje de programación C#	Se especifica el uso del lenguaje de programación C# para el desarrollo del componente
RNDI 2. PostgreSQL como gestor de base de datos	Se especifica el uso de PostgreSQL para el desarrollo de la base de datos del componente
RNEA 1: Utilizar los perfiles de IHE	Se utilizan los perfiles de IHE debido a que facilitan la gestión de la autenticación de los usuarios
RNF 1. El sistema debe estar disponible siempre	El componente de identificación centralizada es una pieza clave en el flujo de trabajo de la solución alas PACS-RIS por lo cual debe estar disponible siempre

Tabla 3. Requisitos no funcionales (fuente: creada por los autores)

En la Figura 9 se muestra el diagrama de requisitos no funcionales agrupados por paquetes lógicos.

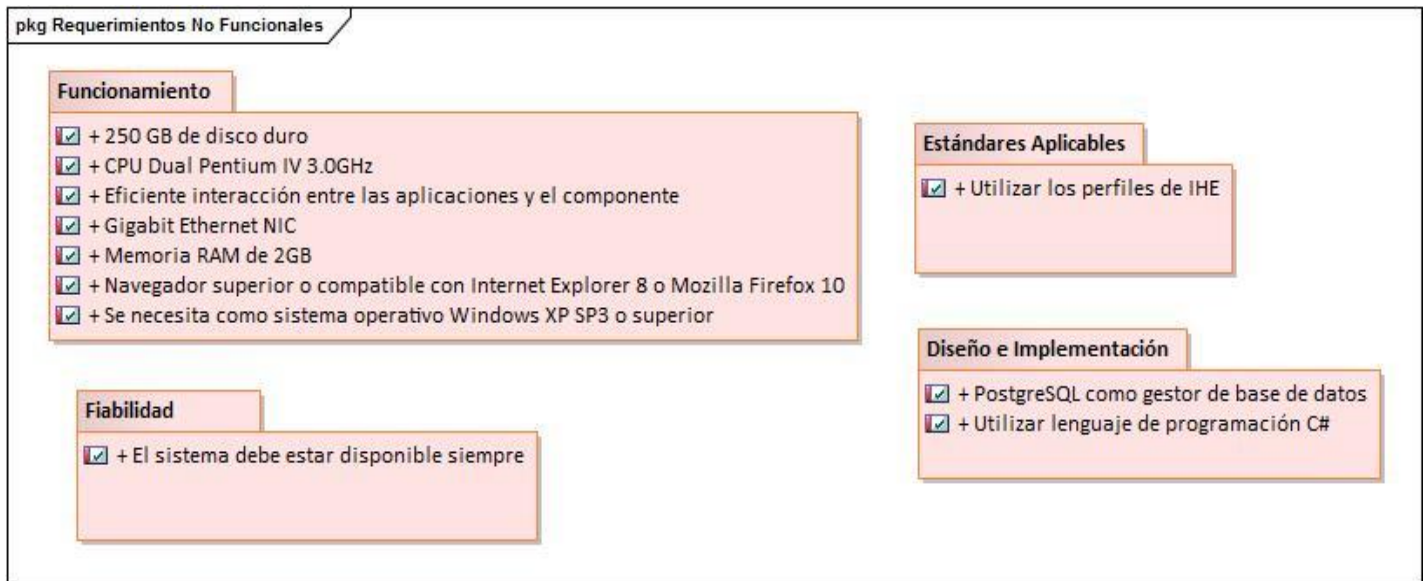


Figura 9. Requisitos no funcionales del componente de identificación centralizada (fuente: creada por los autores)

2.5. Definición de los actores del sistema







Actores	Justificación
 <p>alas_PACSViewer</p>	<p>Representa el visor de imágenes médicas del sistema alas PACS</p>
 <p>alas_PACSReporter</p>	<p>Representa el reportador de informes del sistema alas PACS</p>
 <p>alas_PACSServer</p>	<p>Representa el servidor de almacenamiento de imágenes médicas</p>
 <p>alas_RIS</p>	<p>Representa el sistema alas RIS, encargado de la gestión administrativa, la gestión de citas y del control de los recursos médicos</p>
 <p>Solución_PACS-RIS</p>	<p>Agrupar a las aplicaciones alas PACSViewer, alas PACSReporter, alas PACSServer y alas RIS conformando así la solución PACS-RIS</p>
 <p>Administrador</p>	<p>Representa al administrador general que controla todo lo relacionado a los servicios que consume la solución alas PACS-RIS</p>

Tabla 4. Definición de los actores del sistema (fuente: creada por los autores)

2.6. Diagrama de Casos de Uso del Sistema

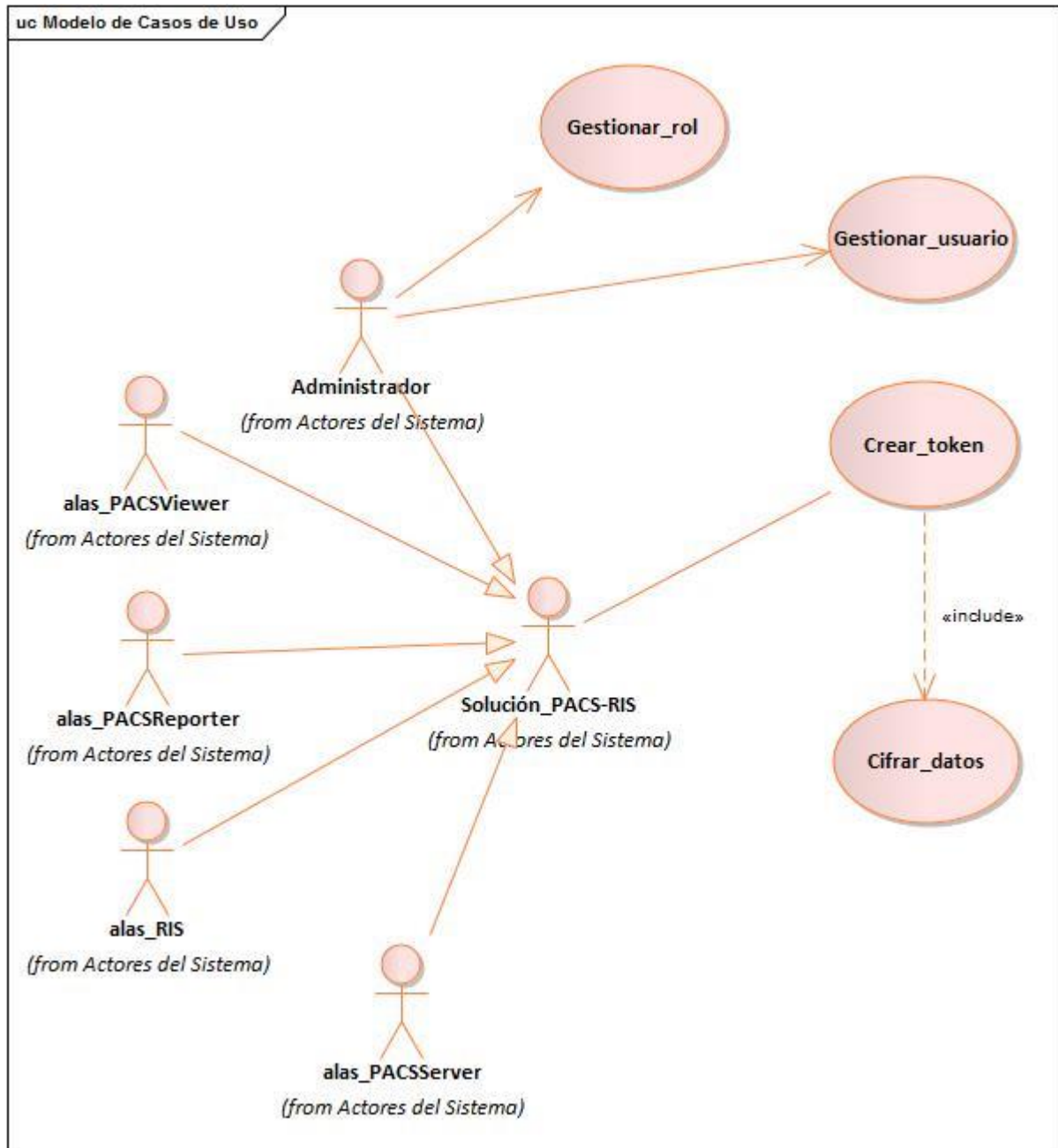


Figura 10. Diagrama de Casos de Uso del Sistema (fuente: creada por los autores)

2.7. Descripción de los Casos de Uso del Sistema

Las tablas que se muestran a continuación muestran un resumen de los casos de uso del sistema. En el [Anexo 1](#) se muestran las descripciones textuales ampliadas de los casos de uso.

CU-1	Gestionar Usuario
Actor	Administrador
Descripción	Permite gestionar las operaciones que se realizan a los usuarios, insertar un nuevo usuario, modificar datos del usuario y eliminar cualquier usuario.
Referencia	RF 2, RF 3, RF 4

Tabla 5. Descripción del caso de uso Gestionar usuario

CU-2	Crear token
Actor	Solución_PACS-RIS
Descripción	Se validan los datos obtenidos de las aplicaciones, se crea el token de seguridad con los datos del usuario que se encuentra autenticado y posteriormente se envía el token a las aplicaciones para que le otorguen los permisos pertinentes al usuario.
Referencia	RF 1, RF 5, RF 7

Tabla 6. Descripción del caso de uso Crear token

CU-3	Cifrar datos
Actor	Solución_PACS-RIS
Descripción	Se cifran los datos obtenidos del proceso de identificación de los usuarios en las aplicaciones que conforman la solución alas PACS-RIS mediante el algoritmo MD5.
Referencia	RF 6

Tabla 7. Descripción del caso de uso Cifrar datos

CU-1	Gestionar rol
Actor	Administrador
Descripción	Permite gestionar las operaciones sobre los roles que se encuentran registrados en el sistema, insertar un nuevo rol o eliminar roles de la base de datos.
Referencia	RF 8, RF 9

Tabla 8. Descripción del caso de uso Gestionar rol

2.8. Conclusiones

Se realizó una propuesta de solución para la problemática planteada de un componente de identificación centralizada para sistemas multientidad (escritorio y web), con las especificaciones descritas en el perfil EUA de IHE. Se identificaron las características que debe cumplir el componente de identificación centralizada para la solución alas PACS-RIS y se describieron en forma de requisitos funcionales y no funcionales; con ello se logró un mayor entendimiento de los procesos a informatizar. Con la definición y descripción de los casos de usos del sistema se obtuvieron las funcionalidades que va cumplir el componente a desarrollar.

CAPÍTULO 3. DISEÑO E IMPLEMENTACIÓN DEL COMPONENTE DE IDENTIFICACIÓN CENTRALIZADA

En este capítulo se presentan los diagramas de clases del diseño y los diagramas de secuencia de los casos de uso arquitectónicamente significativos, Se define la arquitectura utilizada para el desarrollo del componente de identificación centralizada y se muestran los diagramas de componente y despliegue ofreciendo una breve descripción de los elementos que los componen.

3.1. Diseño

El diseño de software es quizás la etapa más importante y definitoria del proceso de desarrollo de software para que el producto que se obtenga sea de calidad. El propósito del diseño se puede resumir en transformar los requisitos en un diseño que describa cómo el sistema debe ser, desarrollar una robusta arquitectura del sistema y adaptar el diseño para que se corresponda con el entorno de implementación, diseñando sus funcionalidades. Debe estar bien especificado para que el sistema pueda ser implementado sin imprecisiones (41).

Mediante los diagramas de clases del diseño es posible alcanzar un mejor entendimiento de la forma en que se encuentra estructurado el sistema, cada clase del diagrama cuenta con atributos y métodos bien definidos lo que facilita al programador el desarrollo de la aplicación. Estos diagramas pueden ser consultados en el [Anexo 2](#).

Como parte de esta fase, la elaboración de los diagramas de secuencia representa en forma gráfica, la entrada de datos al sistema, los procesos y salidas de la información. Se muestra detalladamente y haciendo uso de mensajes la acción realizada entre los objetos que conforman el sistema. Se observan en el [Anexo 3](#).

3.2. Patrones de diseño

Los patrones GRASP (General Responsibility Assignment Software Patterns) describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en formas de patrones (42).

En la implementación del componente de identificación centralizada para la solución alas PACS-RIS se utiliza el patrón de diseño “controlador”. Su uso se hace evidente al utilizar en la implementación una clase controladora capaz de obtener información del usuario y enviarla a cada clase según el método llamado. Esto trae como beneficio la reutilización de código y el establecimiento de un mayor control de la aplicación.

3.3. Modelo arquitectónico

El objetivo de la arquitectura es identificar los requisitos que producen un impacto en la estructura del sistema y reducir los riesgos asociados con la construcción del mismo. La arquitectura debe soportar los cambios futuros del software, del hardware y de las funcionalidades demandada por los clientes (43).

El componente de identificación y control de acceso está desarrollado para brindar servicios, de los cuales las aplicaciones que conforman la solución alas PACS-RIS consumen y obtienen información sobre los permisos con que cuentan los usuarios para acceder a los sistemas de la solución. Para su desarrollo se utilizaron elementos de la arquitectura orientada a servicios (SOA, por sus siglas en inglés).

Esta arquitectura se define como un conjunto de proveedores de servicios que exponen su funcionalidad mediante interfaces públicas (que pueden estar también protegidas). Las interfaces expuestas por los proveedores de servicios pueden ser consumidas individualmente o bien agregando varios servicios y formando proveedores de servicios compuestos (44).

SOA proporciona la infraestructura necesaria para que el intercambio de información y la participación en los procesos de negocio se lleve a cabo con total independencia de la plataforma hardware-software sobre la que trabajan: sistema operativo, lenguaje de programación, características de los equipos, entre otros (44). En la Figura 11 se muestra el esquema de la arquitectura.

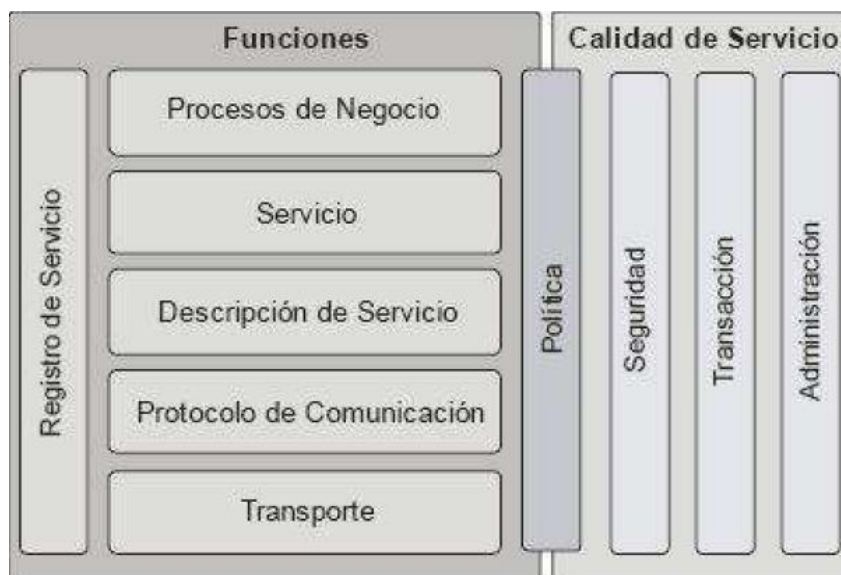


Figura 11. Esquema de la Arquitectura Orientada a Servicio SOA (45)

El esquema se encuentra dividido en 2 zonas; una que abarca el ámbito funcional de la arquitectura y otra vinculada a la calidad de servicio.

Los elementos que componen la arquitectura son:

Funciones

- Transporte: mecanismo utilizado para llevar las demandas de servicio desde un consumidor hacia un proveedor de servicio y las respuestas desde el proveedor hacia el consumidor.
- Protocolo de comunicación de servicios: es un mecanismo acordado a través del cual un proveedor de servicios y un consumidor de servicios comunican qué está siendo solicitado y qué está siendo respondido.
- Descripción de servicio: es un esquema acordado para describir qué es el servicio, cómo debe invocarse y qué datos requiere el servicio para invocarse con éxito.
- Servicio: describe un servicio actual que está disponible para utilizar.
- Procesos de Negocio: es una colección de servicios, invocados en una secuencia particular con un conjunto específico de reglas para satisfacer un requisito de negocio.
- Registro de Servicios: es un repositorio de descripciones de servicios y datos que pueden utilizar los proveedores de servicios para publicar sus servicios, así como los consumidores de servicios para descubrir o hallar servicios disponibles.

Calidad de Servicio

- Política: conjunto de condiciones o reglas bajo las cuales un proveedor de servicio hace el servicio disponible para consumidores.
- Seguridad: conjunto de reglas que pueden aplicarse para la identificación, autorización y control de acceso a consumidores de servicios.
- Transacciones: conjunto de atributos que podrían aplicarse a un grupo de servicios para entregar un resultado consistente.
- Administración: conjunto de atributos que podrían aplicarse para manejar los servicios proporcionados o consumidos (45).

La arquitectura SOA se basa en una forma natural de pensar los procesos, atomizando el conocimiento sin adicionar conceptos propiamente informáticos, hasta el momento de su implementación física. Una de las principales características de SOA es que resuelve los problemas de conectividad y permite independencia de las plataformas de desarrollo.

3.4. Modelo de Datos

El modelo de datos describe de forma gráfica como se encuentra estructurada la información dentro de la base de datos. En la figura 12 se observa la relación entre las tablas que conforman la base de datos.

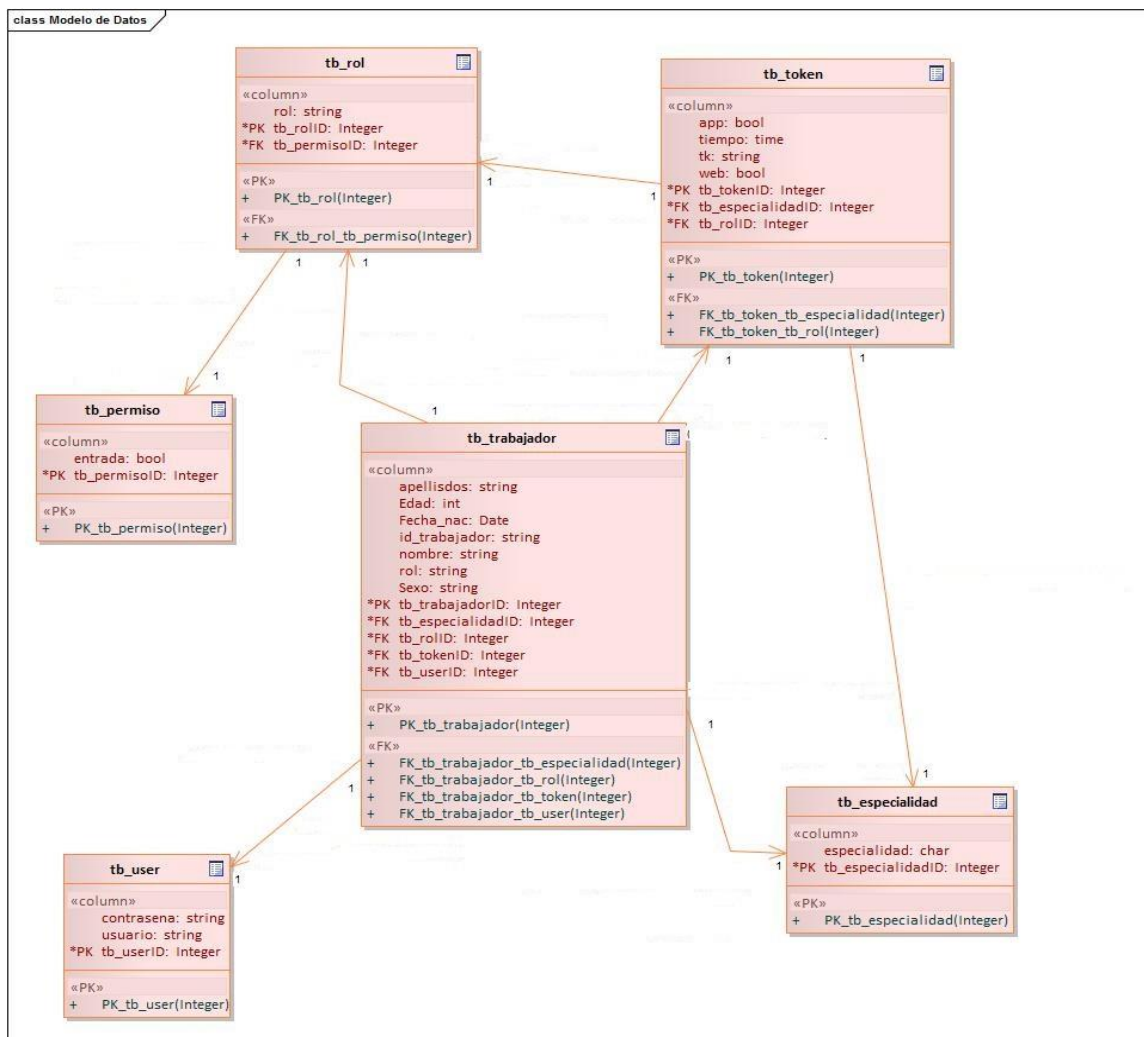


Figura 12. Modelo de datos (fuente: creada por los autores)

3.5. Descripción de los componentes del modelo de datos

Tabla. tb_rol

Contiene el rol que define al usuario que se encuentra registrado en el sistema, mediante su rol es que se conoce el acceso que tiene el usuario.

Tabla. tb_user

Contiene los atributos usuario y contraseña, estos datos son los que identifican al usuario cuando intenta entrar en un sistema.

Tabla. tb_trabajador

Contiene toda la información necesaria para registrar al usuario dentro del sistema, almacena datos como el nombre, apellidos, sexo, edad entre otros.

Tabla. tb_token

Almacena la información que será compartida con las aplicaciones para identificar a los usuarios dentro de ellas.

Tabla. tb_permiso

Contiene una variable booleana que indica el estado de los permisos de los usuarios para acceder a la ventana de administración del servicio web.

Tabla. tb_especialidad

Almacena el tipo de especialidad a la que puede pertenecer el usuario que se autentica, se refiere a cardiología, neurología ortopedia entre otras.

3.6. Web Services Description Language (WSDL)

Es un protocolo para la descripción de Servicios Web basado en un formato XML, indica cuáles son las interfaces que provee el Servicio web y los tipos de datos necesarios para su utilización. Define dónde está disponible el servicio y qué protocolo de comunicaciones utilizar para comunicarse con el mismo. Se observa en el [Anexo 4](#).

3.7. Diagrama de componentes

El diagrama de componentes modela los aspectos físicos de un sistema, la vista de implementación estática y los elementos físicos que residen en un nodo, tales como ejecutables, tablas, librerías, archivos y documentos (46). Para un mejor entendimiento del diagrama de componente se muestra en la Figura 13.

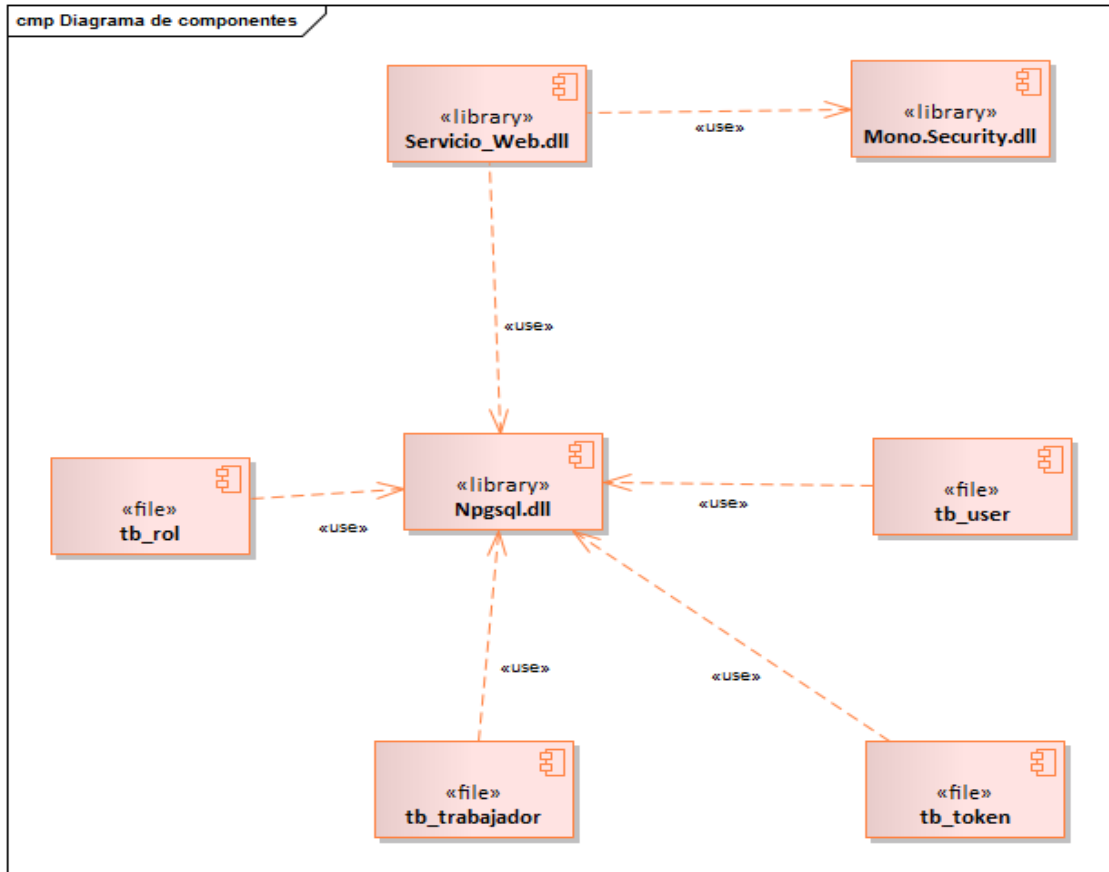


Figura 13. Diagrama de componentes (fuente: creada por los autores)

Se empleó la librería Npgsql.dll como enlace dinámico para enviar y recibir datos del servidor PostgreSQL; Mono.Security.dll para la seguridad y la captura de errores y Servicio_Web.dll que es donde se encuentra alojado el servicio.

3.8. Modelo de despliegue

El diagrama de despliegue provee un modelo detallado de la forma en la que los componentes se desplegarán a lo largo de la infraestructura del sistema. Detalla las capacidades de red, las especificaciones del servidor, los requisitos de hardware y otra información relacionada al despliegue del componente propuesto. En la Figura 14 se muestra el diagrama de despliegue del sistema.

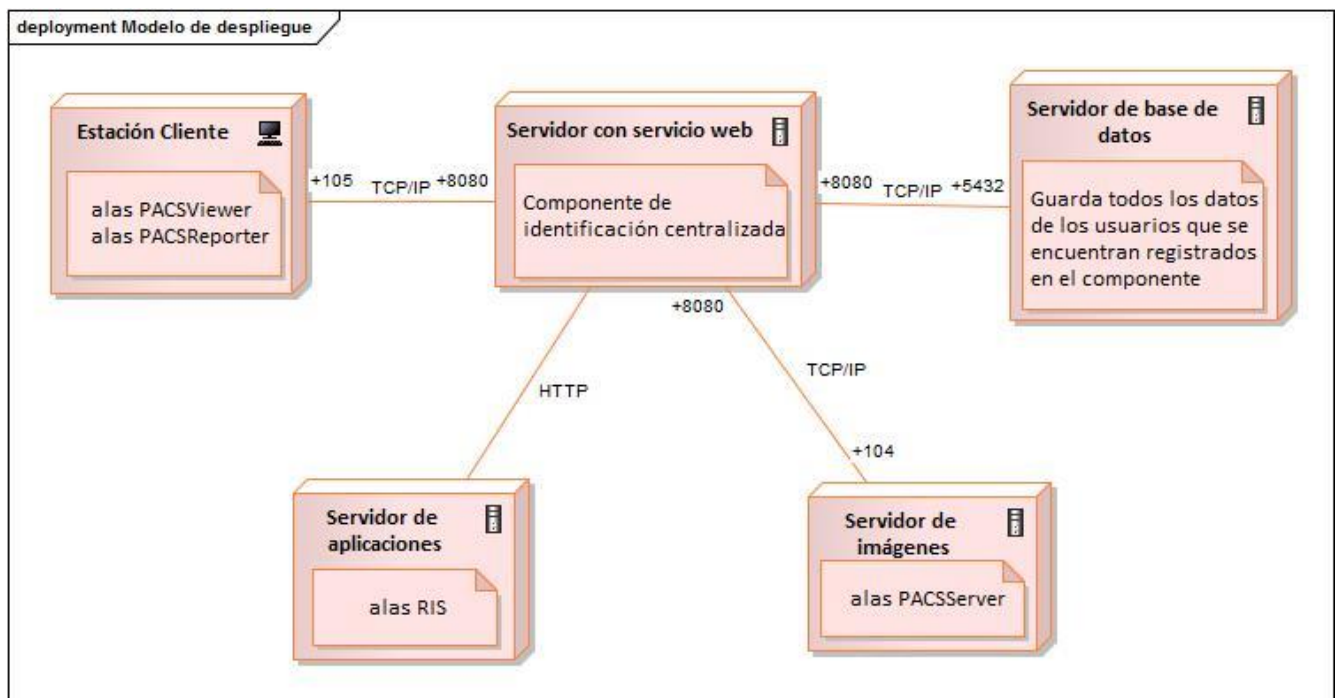


Figura 14. Diagrama de despliegue (fuente: creada por los autores)

Nodo Estación Cliente

Representa las estaciones de trabajo en donde van a estar instalados el alas PACSViewer y el alas PACSReporter y de donde el usuario va a acceder al sistema alas RIS mediante un navegador web. Desde aquí el usuario podrá gestionar las imágenes médicas, generar reportes y programar citas para consultas médicas.

Nodo Servidor con servicio web

Representa el servidor donde se va a encontrar instalado el componente que brindará los servicios a las aplicaciones que conforman la solución alas PACS-RIS. Teniendo en cuenta que es un servicio web el puerto

por el cual se comunica con las aplicaciones es configurable, por lo que en el diagrama se representa el puerto por defecto.

Nodo Servidor de base de datos

Este servidor va ser el encargado de guardar todos los datos de los usuarios que se encuentran registrados en el sistema. Estos datos se encuentran cifrados para una mayor protección de la información.

Nodo Servidor de imágenes

Representa el servidor de imágenes médicas alas PACSServer en el cual se almacenan todas las imágenes médicas con que se trabajan en la solución alas PACS-RIS.

Nodo Servidor de aplicaciones

En este nodo se encuentra instalado el sistema alas RIS. En el entorno real donde se encuentra desplegada la solución alas PACS-RIS, el componente de identificación, que en el diagrama se representa en el nodo servidor con servicio web, pudiera estar instalado en el servidor de aplicaciones debido a que cumple con todas las condiciones de rendimiento y capacidad.

3.9. Conclusiones

Los requisitos funcionales y no funcionales del sistema fueron especificados en forma de diseño, se elaboraron los diagramas de clases del diseño y los diagramas de secuencia de los casos de uso arquitectónicamente significativos; representando una primera aproximación de cómo se va a implementar el componente de identificación centralizada para la solución alas PACS-RIS. Se determinó que el patrón arquitectónico a utilizar durante la fase de desarrollo, será la Arquitectura Orientada a Servicios. Se concluyó que dicha arquitectura es la idónea para el desarrollo del componente, porque facilita el intercambio de información con total independencia de la plataforma hardware-software sobre la que se trabaja. Se describe de forma gráfica la estructura de información dentro de la base de datos, alcanzando una idea específica de los datos existentes y de la forma en que estos se relacionan. Con la representación del diagrama de componentes y de despliegue se evidencian los elementos que conforman el componente de identificación centralizada, la interacción entre ellos y se obtuvo la forma en que estos serán desplegados físicamente.

CONCLUSIONES

Una vez finalizado el proceso de investigación se puede arribar a las siguientes conclusiones:

- Después de realizado el estudio sobre los sistemas que realizan autenticación centralizada, se identificó que su adopción no es factible, pues no cumplen con las necesidades para eliminar el problema descrito, por lo que se concluyó que era necesario construir un componente para la identificación centralizada para la solución alas PACS-RIS.
- Se desarrolló un componente que posibilita la identificación centralizada de soluciones multi-entidad y se concluye que es factible su utilización en el entorno de funcionamiento real de la solución alas PACS-RIS
- La centralización del proceso de autenticación, posibilita que el usuario solo tenga que autenticarse una vez, dando cumplimiento al objetivo general de la investigación.

RECOMENDACIONES

Teniendo en cuenta el estudio realizado durante todo el proceso de desarrollo del trabajo, en aras de enriquecer la solución desarrollada, los autores recomiendan:

- ✓ Comprobar la compatibilidad del componente en un entorno GNU/LINUX.
- ✓ Incorporar el componente a la solución alas PACS-RIS para que sea probado en su entorno real de funcionamiento.
- ✓ Desarrollar las condiciones necesarias para que el componente pueda conectarse a servidores de dominio como LDAP.

REFERENCIAS BIBLIOGRÁFICAS

1. Ing. Edgardo Bonfils. Asociación Internacional de Clubes de Leones. *Asociación Internacional de Clubes de Leones*. [En línea] [Citado el: 8 de 4 de 2013.] <http://www.leonismoargentino.com.ar/SalDiagPorImagen.html>.
2. Grupo PAS. *Estándar y Protocolo de Imágenes Médicas DICOM*. s.l. : Universidad de Deusto, 2005.
3. Izaguirre, Leodan Vega. *Alas RIS Sistema de gestión de información* . La Habana : s.n., 2008.
4. González, Juan Pablo Alejo. IHE, el comienzo de la integración de la empresa de salud desde el Servicio de Radiología. [En línea] [Citado el: 30 de 4 de 2013.] http://www.conganat.org/SEIS/is/is45/IS45_103.pdf.
5. IHE-E. IHE España. *IHE España*. [En línea] [Citado el: 27 de 4 de 2013.] <http://www.ihe-e.org/>.
6. Rodriguez, Juan. subinet. *subinet*. [En línea] 23 de agosto de 2010. [Citado el: 28 de 4 de 2013.] <http://www.subinet.es/guias-y-tips/guias-tips-internet/%C2%BFque-es-el-control-de-acceso-en-sistemas-informaticos/>.
7. microsoft. msdn. *Autenticación*. [En línea] [Citado el: 28 de 4 de 2013.] <http://msdn.microsoft.com/es-es/library/syf5yeat.aspx>.
8. Velázquez, Karel Gómez. *Propuesta de procesos de autenticación, autorización y auditoría(AAA) para aplicaciones basadas en servicios web xml*. La Habana : s.n., 2010.
9. emarketservices. emarketservices. *emarketservices*. [En línea] [Citado el: 15 de 4 de 2013.] <http://www.emarketservices.es/icex/cma/contentTypes/common/records/mostrarDocumento/?doc=415901>.
10. Martínez M. A., Jiménez A.J.R, Medina B. V. Azpiroz L. J. Area de procesamiento digital de imágenes Biomédicas. *LOS SISTEMAS PACS*. [En línea] Itzamna. [Citado el: 15 de 2 de 2013.] <http://itzamna.uam.mx/pdsib/PROYECTOS/PACS/PACS.HTML>.
11. Leudis Hernandez Sánchez, Gerardo Ceruto Marrero. *Visor Ligero para la visualización y procesamiento básico de imágenes médicas*. Habana : s.n., 2010.

12. Nema. Digital Imaging and Communications in Medicine (DICOM). *Digital Imaging and Communications in Medicine (DICOM)*. [En línea] [Citado el: 10 de 2 de 2013.] <http://medical.nema.org/standard.html>.
13. Izaguirre, Leodan Vega. *Alas RIS Sistema de gestión de información*. La Habana : s.n., 2008.
14. Rodríguez López, Martha y Rodríguez García, Raymundo. Propuesta de aplicación de los perfiles de integración de IHE entre los sistemas alas PACS–alas RIS–alas HIS. La Habana : s.n., 2010.
15. IHE. IHE changing the way healthcare. *IHE changing the way healthcare*. [En línea] 2012. [Citado el: 25 de 3 de 2013.] <http://www.ihe.net/profiles/index.cfm>.
16. Iván M. Caballero, Jeimy J. Cano. *CONSIDERACIONES PARA IMPLEMENTAR UNA ARQUITECTURA SINGLE SIGN-ON*. Colombia : s.n., 2003.
17. ipsca. web.ipsca. *web.ipsca*. [En línea] [Citado el: 12 de 3 de 2013.] http://web.ipsca.com/en/Products_SSO_Single_Sign_On.
18. FreeBSD. FreeBSD. *Kerberos5*. [En línea] [Citado el: 8 de 1 de 2013.] <http://www.freebsd.org/doc/es/books/handbook/kerberos5.html>.
19. msdn. Autenticación Kerberos y SQL Server. *Autenticación Kerberos y SQL Server*. [En línea] [Citado el: 10 de 6 de 2013.] <http://msdn.microsoft.com/es-es/library/cc280744%28v=sql.105%29.aspx>.
20. Red Hat Linux. Red Hat Linux 4: Manual de referencia de Red Hat Linux. *Red Hat Linux 4: Manual de referencia de Red Hat Linux*. [En línea] [Citado el: 10 de 6 de 2013.] <http://www.gb.nrao.edu/pubcomputing/redhatELWS4/RH-DOCS/rhel-rg-es-4/ch-kerberos.html>.
21. Oracle. oracle.com. *oracle.com*. [En línea] 8 de 2011. [Citado el: 12 de 3 de 2013.] http://docs.oracle.com/cd/E24842_01/html/E23286/intro-25.html.
22. OpenID. crear crear. *crear crear*. [En línea] [Citado el: 3 de 5 de 2013.] <http://www.crearcrear.com/openid-identificacion-digital/>.
23. —. openid.es. *openid.es*. [En línea] [Citado el: 3 de 5 de 2013.] <http://openid.es/accion/help>.
24. Microsoft. Centro de seguridad y protección. *Microsoft*. [En línea] [Citado el: 3 de 5 de 2013.] <http://www.microsoft.com/es-es/security/online-privacy/phishing-symptoms.aspx>.

25. passwordbank. passwordbank. *passwordbank*. [En línea] [Citado el: 10 de 12 de 2012.] <http://www.passwordbank.com/es/passwordbank-esso/>.
26. gehealthcare. Centricity Imaging Solutions. *gehealthcare*. [En línea] [Citado el: 15 de 6 de 2013.] <http://www.gehealthcare.com/Centricity-Imaging-Solutions/assets/centricity-ris-pacs-iw-brochure.pdf>.
27. Imprivata. Imprivata. [En línea] 2012. [Citado el: 10 de 6 de 2013.] http://www.imprivata.com/sites/default/files/resource-files/0812_Radiology_Web.pdf.
28. —. Fingerprint biometrics increase data security. [En línea] [Citado el: 15 de 6 de 2013.] http://www.imprivata.com/fingerprint_biometrics.
29. Lázaro Antonio Marín Mártires, Luis Ramón Capriles Alvarado. *Sistema de autenticación y autorización centralizado*. Ciudad de la Habana : s.n., 2008.
30. SOFTEL. *Documento sobre la Arquitectura de Software para los componentes a emplear por el Sistema de Información para la Salud*. Habana : s.n., 2006.
31. Baryolo, Oiner Gómez. Serie científica. *Serie científica*. [En línea] 2011. [Citado el: 28 de 5 de 2013.] <http://publicaciones.uci.cu/index.php/SC/article/view/702>.
32. MasterMagazine. *MasterMagazine*. [En línea] [Citado el: 13 de febrero de 2013.] <http://www.mastermagazine.info/termino/7018.php>.
33. IBM. IBM. *Unified Modeling Language (UML)*. [En línea] 13 de 5 de 2012. [Citado el: 27 de 4 de 2013.] <http://www-01.ibm.com/software/rational/uml/>.
34. Microsoft. Microsoft Developer Network. *Microsoft Developer Network*. [En línea] [Citado el: 27 de 4 de 2013.] <http://msdn.microsoft.com/es-es/library/ms731082.aspx>.
35. Infolnova. WCF vs ASP.NET Web services. [En línea] [Citado el: 13 de 6 de 2013.] <http://infoinnova.net/2011/11/wcfvsaspnet/>.
36. Sparx System. *Sparx System*. [En línea] <http://www.sparxsystems.com.ar/products/ea.html>.
37. PostgreSQL. [En línea] 2 de 10 de 2010. http://www.postgresql.org/es/sobre_postgresql.
38. Microsoft. Microsoft. *Microsoft*. [En línea] [Citado el: 10 de 4 de 2013.] <http://www.microsoft.com/visualstudio/esn/products/visual-studio-team-foundation-server-2012>.

39. Allsoft. *Allsoft*. [En línea] [Citado el: 13 de febrero de 2013.] <http://www.allsoft.mx/recursos/EIModeloCMMI.pdf>.
40. Somerville, Ian. *Ingeniería del Software*. Madrid : Septima edición, 2006.
41. Universidad de Alemania. INFORMATION SYSTEMS GROUP. *INFORMATION SYSTEMS GROUP*. [En línea] [Citado el: 18 de 4 de 2013.] <http://indalog.ual.es/mtorres/LP/FundamentosDiseno.pdf>.
42. Grosso, Andrés. *Practicas de Software*. *Prácticas de Software*. [En línea] 21 de 3 de 2011. [Citado el: 12 de 4 de 2013.] <http://www.practicadesoftware.com.ar/2011/03/patrones-grasp/>.
43. Márquez, Alejandro Delgado García y Carlos Luis Castro. *Visor Web de imágenes médicas digitales y sistema de reportes radiológicos*. La Habana : s.n., 2012.
44. Llorente, Cesar de la Torre. *Guía de Arquitectura N-Capas orientadas al Dominio con .NET 4.0*. Madrid : Krasis Consulting, S. L., 2010.
45. César de la Torre, Roberto González. *Arquitectura SOA con tecnología Microsoft*. s.l. : Krasis Press, 2008.
46. microsoft. msdn. *Microsoft Developer Network*. [En línea] [Citado el: 23 de 5 de 2013.] <http://msdn.microsoft.com/es-es/library/dd409390.aspx>.

BIBLIOGRAFÍA

1. Ing. Edgardo Bonfils. Asociación Internacional de Clubes de Leones. *Asociación Internacional de Clubes de Leones*. [En línea] [Citado el: 8 de 4 de 2013.] <http://www.leonismoargentino.com.ar/SalDiagPorImagen.html>.
2. Grupo PAS. *Estándar y Protocolo de Imágenes Médicas DICOM*. s.l. : Universidad de Deusto, 2005.
3. Izaguirre, Leodan Vega. *Alas RIS Sistema de gestión de información* . La Habana : s.n., 2008.
4. González, Juan Pablo Alejo. IHE, el comienzo de la integración de la empresa de salud desde el Servicio de Radiología. [En línea] [Citado el: 30 de 4 de 2013.] http://www.conganat.org/SEIS/is/is45/IS45_103.pdf.
5. IHE-E. IHE España. *IHE España*. [En línea] [Citado el: 27 de 4 de 2013.] <http://www.ihe-e.org/>.
6. Rodriguez, Juan. subinet. *subinet*. [En línea] 23 de agosto de 2010. [Citado el: 28 de 4 de 2013.] <http://www.subinet.es/guias-y-tips/guias-tips-internet/%C2%BFque-es-el-control-de-acceso-en-sistemas-informaticos/>.
7. microsoft. msdn. *Autenticación*. [En línea] [Citado el: 28 de 4 de 2013.] <http://msdn.microsoft.com/es-es/library/syf5yeat.aspx>.
8. Velázquez, Karel Gómez. *Propuesta de procesos de autenticación, autorización y auditoría(AAA) para aplicaciones basadas en servicios web xml*. La Habana : s.n., 2010.
9. emarketservices. emarketservices. *emarketservices*. [En línea] [Citado el: 15 de 4 de 2013.] <http://www.emarketservices.es/icex/cma/contentTypes/common/records/mostrarDocumento/?doc=415901>.
10. Martínez M. A., Jiménez A.J.R, Medina B. V. Azpiroz L. J. Area de procesamiento digital de imágenes Biomédicas. *LOS SISTEMAS PACS*. [En línea] Itzamna. [Citado el: 15 de 2 de 2013.] <http://itzamna.uam.mx/pdsib/PROYECTOS/PACS/PACS.HTML>.
11. Leudis Hernandez Sánchez, Gerardo Ceruto Marrero. *Visor Ligero para la visualización y procesamiento básico de imágenes médicas*. Habana : s.n., 2010.

12. Nema. Digital Imaging and Communications in Medicine (DICOM). *Digital Imaging and Communications in Medicine (DICOM)*. [En línea] [Citado el: 10 de 2 de 2013.] <http://medical.nema.org/standard.html>.
13. Izaguirre, Leodan Vega. *Alas RIS Sistema de gestión de información*. La Habana : s.n., 2008.
14. Rodríguez López, Martha y Rodríguez García, Raymundo. Propuesta de aplicación de los perfiles de integración de IHE entre los sistemas alas PACS–alas RIS–alas HIS. La Habana : s.n., 2010.
15. IHE. IHE changing the way healthcare. *IHE changing the way healthcare*. [En línea] 2012. [Citado el: 25 de 3 de 2013.] <http://www.ihe.net/profiles/index.cfm>.
16. Iván M. Caballero, Jeimy J. Cano. *CONSIDERACIONES PARA IMPLEMENTAR UNA ARQUITECTURA SINGLE SIGN-ON*. Colombia : s.n., 2003.
17. ipsca. web.ipsca. *web.ipsca*. [En línea] [Citado el: 12 de 3 de 2013.] http://web.ipsca.com/en/Products_SSO_Single_Sign_On.
18. FreeBSD. FreeBSD. *Kerberos5*. [En línea] [Citado el: 8 de 1 de 2013.] <http://www.freebsd.org/doc/es/books/handbook/kerberos5.html>.
19. msdn. Autenticación Kerberos y SQL Server. *Autenticación Kerberos y SQL Server*. [En línea] [Citado el: 10 de 6 de 2013.] <http://msdn.microsoft.com/es-es/library/cc280744%28v=sql.105%29.aspx>.
20. Red Hat Linux. Red Hat Linux 4: Manual de referencia de Red Hat Linux. *Red Hat Linux 4: Manual de referencia de Red Hat Linux*. [En línea] [Citado el: 10 de 6 de 2013.] <http://www.gb.nrao.edu/pubcomputing/redhatELWS4/RH-DOCS/rhel-rg-es-4/ch-kerberos.html>.
21. Oracle. oracle.com. *oracle.com*. [En línea] 8 de 2011. [Citado el: 12 de 3 de 2013.] http://docs.oracle.com/cd/E24842_01/html/E23286/intro-25.html.
22. OpenID. crear crear. *crear crear*. [En línea] [Citado el: 3 de 5 de 2013.] <http://www.crearcrear.com/openid-identificacion-digital/>.
23. —. openid.es. *openid.es*. [En línea] [Citado el: 3 de 5 de 2013.] <http://openid.es/accion/help>.
24. Microsoft. Centro de seguridad y protección. *Microsoft*. [En línea] [Citado el: 3 de 5 de 2013.] <http://www.microsoft.com/es-es/security/online-privacy/phishing-symptoms.aspx>.

25. passwordbank. passwordbank. *passwordbank*. [En línea] [Citado el: 10 de 12 de 2012.] <http://www.passwordbank.com/es/passwordbank-esso/>.
26. gehealthcare. Centricity Imaging Solutions. *gehealthcare*. [En línea] [Citado el: 15 de 6 de 2013.] <http://www.gehealthcare.com/Centricity-Imaging-Solutions/assets/centricity-ris-pacs-iw-brochure.pdf>.
27. Imprivata. Imprivata. [En línea] 2012. [Citado el: 10 de 6 de 2013.] http://www.imprivata.com/sites/default/files/resource-files/0812_Radiology_Web.pdf.
28. —. Fingerprint biometrics increase data security. [En línea] [Citado el: 15 de 6 de 2013.] http://www.imprivata.com/fingerprint_biometrics.
29. Lázaro Antonio Marín Mártires, Luis Ramón Capriles Alvarado. *Sistema de autenticación y autorización centralizado*. Ciudad de la Habana : s.n., 2008.
30. SOFTEL. *Documento sobre la Arquitectura de Software para los componentes a emplear por el Sistema de Información para la Salud*. Habana : s.n., 2006.
31. Baryolo, Oiner Gómez. Serie científica. *Serie científica*. [En línea] 2011. [Citado el: 28 de 5 de 2013.] <http://publicaciones.uci.cu/index.php/SC/article/view/702>.
32. MasterMagazine. *MasterMagazine*. [En línea] [Citado el: 13 de febrero de 2013.] <http://www.mastermagazine.info/termino/7018.php>.
33. IBM. IBM. *Unified Modeling Language (UML)*. [En línea] 13 de 5 de 2012. [Citado el: 27 de 4 de 2013.] <http://www-01.ibm.com/software/rational/uml/>.
34. Microsoft. Microsoft Developer Network. *Microsoft Developer Network*. [En línea] [Citado el: 27 de 4 de 2013.] <http://msdn.microsoft.com/es-es/library/ms731082.aspx>.
35. Infolnova. WCF vs ASP.NET Web services. [En línea] [Citado el: 13 de 6 de 2013.] <http://infoinnova.net/2011/11/wcfvsaspnet/>.
36. Sparx System. *Sparx System*. [En línea] <http://www.sparxsystems.com.ar/products/ea.html>.
37. PostgreSQL. [En línea] 2 de 10 de 2010. http://www.postgresql.org/es/sobre_postgresql.
38. Microsoft. Microsoft. *Microsoft*. [En línea] [Citado el: 10 de 4 de 2013.] <http://www.microsoft.com/visualstudio/esn/products/visual-studio-team-foundation-server-2012>.

39. Allsoft. *Allsoft*. [En línea] [Citado el: 13 de febrero de 2013.] <http://www.allsoft.mx/recursos/EIModeloCMMI.pdf>.
40. Somerville, Ian. *Ingeniería del Software*. Madrid : Septima edición, 2006.
41. Universidad de Alemania. INFORMATION SYSTEMS GROUP. *INFORMATION SYSTEMS GROUP*. [En línea] [Citado el: 18 de 4 de 2013.] <http://indalog.ual.es/mtorres/LP/FundamentosDiseno.pdf>.
42. Grosso, Andrés. *Practicas de Software*. *Prácticas de Software*. [En línea] 21 de 3 de 2011. [Citado el: 12 de 4 de 2013.] <http://www.practicadesoftware.com.ar/2011/03/patrones-grasp/>.
43. Márquez, Alejandro Delgado García y Carlos Luis Castro. *Visor Web de imágenes médicas digitales y sistema de reportes radiológicos*. La Habana : s.n., 2012.
44. Llorente, Cesar de la Torre. *Guía de Arquitectura N-Capas orientadas al Dominio con .NET 4.0*. Madrid : Krasis Consulting, S. L., 2010.
45. César de la Torre, Roberto González. *Arquitectura SOA con tecnología Microsoft*. s.l. : Krasis Press, 2008.
46. microsoft. msdn. *Microsoft Developer Network*. [En línea] [Citado el: 23 de 5 de 2013.] <http://msdn.microsoft.com/es-es/library/dd409390.aspx>.
47. Segu-info. *Segu-info*. [En línea] [Citado el: 6 de 12 de 2012.] [//www.segu-info.com.ar/logica/seguridadlogica.htm](http://www.segu-info.com.ar/logica/seguridadlogica.htm).
48. Digital Imaging and Communications in Medicine (DICOM). *Digital Imaging and Communications in Medicine (DICOM)*. [En línea] <http://medical.nema.org/>.
49. Martínez, Ivette Carolina. itescam. *itescam*. [En línea] [Citado el: 28 de 2 de 2013.] www.itescam.edu.mx/principalsylabusfpdbrecursosr88846.PDF.
50. Schlumberger Excellence in Educational Development. *Schlumberger Excellence in Educational Development*. [En línea] [Citado el: 7 de 4 de 2013.] <http://www.planetseed.com/es/sciencearticle/historia-de-la-medicina>.
51. LIBROS de AUTORES CUBANOS. *LIBROS de AUTORES CUBANOS*. [En línea] [Citado el: 7 de 4 de 2013.] <http://gsdl.bvs.sld.cu/cgi-bin/library?e=d-00000-00---off-0imaginol--00-0----0-10-0---0---0direct->

10---4-----0-1l--11-es-50---20-about---00-0-1-00-0-0-11-1-0s-
00&a=d&cl=CL1&d=HASH961b392698e12ef2f53bac.9.2.

52. SOCIEDAD ESPAÑOLA DE INFORMÁTICA DE LA SALUD. *SOCIEDAD ESPAÑOLA DE INFORMÁTICA DE LA SALUD*. [En línea] [Citado el: 8 de 4 de 2013.] <http://www.seis.es/jsp/base.jsp?contenido=/jsp/home/home.jsp&id=00>.

53. Proveedores de servicios médicos. *CubanHealth*. [En línea] 2009. [Citado el: 13 de octubre de 2011.] <http://www.cubandhealth.com/proveedores.php>.

54. Fic. Digital Imaging and Communications in Medicine. [En línea] [Citado el: 1 de noviembre de 2010.] <http://www.fic.udc.es/files/asignaturas/58IB/DICOM-cas.pdf>.

55. Dahilys González López, Adrián Fernández Orozco. *Arquitectura de un sistema para la edición de informes de estudios imagenológicos*. La Habana : s.n., 2010.

56. Maikel Sánchez Dieguez, Antonio Enrique Vallés Gámez. *Framework CALIB. Procesos de lectura, procesamiento y visualización de imágenes médicas*. La Habana : s.n., 2010.

57. Adrian Font Hernández, Ariel Martínez Roque. *Aplicación para la Integración y Comunicación de módulos de alas PACS en una estación de Visualización: alasPACSManager*. La Habana : s.n., 2011.

58. Yadeny Aquino Jiménez, Laura M. Palomino Mariño. *Sistema de Seguridad Centralizada de Aplicaciones Web*. La Habana : s.n., 2007.

59. Carlos Guzman Díaz, Denys Bárbaro Vega Aguilar. *SISTEMA PARA EL ALMACENAMIENTO Y TRANSMISION DE IMAGENES MEDICAS VERSION 3.0*. Mayabeque : s.n., 2013.

60. Yudmila Ferrera Ramírez, Aramis Souлары Pelegrín. *Sistema de Autenticación, Autorización y Auditoría para los productos desarrollados en el Área Temática Sistemas de Apoyo a la Salud (SAAA-SAS) v1.1*. La Habana : s.n., 2009.

61. García, Adrian Naranjo. *Sistema de Autenticación y Control de Acceso para aplicaciones del Departamento de Soluciones para la Aduana*. La Habana : s.n., 2010.

62. Yusmisleidy Fernández Placeres, Angel Arias Baños. *Servicio de Autorización para el Sistema de Administración de Identidades*. La Habana : s.n., 2011.

-
63. David Crespo Pérez, Nelson Francisco Fernández Pérez. *Desarrollo del Módulo Configuración del Sistema de Información Hospitalaria alas HIS*. La Habana : s.n., 2010.
64. Yania Junco López, Ariel Felipe Camacho Díaz. *Desarrollo del módulo de columna para el sistema de Planificación Quirúrgica Ortopédica*. La Habana : s.n., 2012.
65. Yicel Rivera Suárez, Pedro Jorge Noguera López. *Desarrollo del módulo de extremidades para el sistema de Planificación Quirúrgica Ortopédica*. La Habana : s.n., 2012.
66. Yusleidis de la Caridad Cepero González, Arturo Yasmany González Yera. *Sistema para la edición de informes de estudios imagenológicos con gestión dinámica de interfaces de usuario*. La Habana : s.n., 2012.
67. General Electric. GE Healthcare. *GE Healthcare*. [En línea] [Citado el: 10 de 6 de 2013.] <http://www.gehealthcare.com/eues/iis/products/radiology/pacs/pacs.html>.

ANEXOS

Anexo 1. Descripción ampliada de los Casos de Uso del Sistema

CU. Gestionar Usuario

Objetivo	El objetivo principal de este caso de uso consiste en gestionar la información de los usuarios que intervienen en el componente
Actores	Administrador (Inicia)
Resumen	El administrador desea realizar cambios en los usuarios, ya sea insertar un nuevo usuario, modificar algún dato o eliminarlo completamente del sistema
Complejidad	Baja
Prioridad	Crítico
Referencias	RF 2, RF 2.1, RF 2.2, RF 2.3
Precondiciones	Debe de existir un usuario autenticado en la solución y tener el rol de administrador
Pos condiciones	Se insertó un nuevo usuario, se modificaron datos del usuario o se eliminó algún usuario
Flujo de eventos	
Flujo básico “Gestionar Usuario”	
<ol style="list-style-type: none"> 1. El Administrador selecciona en la página principal el menú Opciones 2. El Sistema muestra un submenú con las opciones 3. El Administrador selecciona la opción que desea realizar: <ul style="list-style-type: none"> - Si selecciona la opción “Insertar usuario”, ir a la Sección 1 “Insertar usuario” - Si selecciona la opción “Modificar usuario”, ir a la Sección 2 “Modificar usuario” 	

<ul style="list-style-type: none">- Si selecciona la opción “Eliminar usuario”, ir a la Sección 3 “Eliminar usuario” <p>4. Termina el caso de uso</p>
Flujos alternos
*a. En cualquier momento el sistema falla
<ol style="list-style-type: none">1. El Administrador reinicia el sistema, inicia la sección, y solicita la recuperación2. El Sistema reconstruye el fallo ocurrido3. El Administrador comienza de nuevo
Sección 1 “Insertar usuario”
<ol style="list-style-type: none">1. El Administrador selecciona la opción “Insertar usuario”2. El Sistema muestra una ventana con los parámetros necesarios para adicionar el nuevo usuario3. El Administrador introduce los datos del nuevo usuario y presiona el botón “Aceptar”4. El Sistema registra los datos introducidos y muestra un mensaje “Adicionado correctamente”
Flujos alternos
3a. Existen campos vacíos
<ol style="list-style-type: none">1. El Sistema muestra un mensaje indicando que “Existen campos vacíos”
3b. Entrada de datos no válidos
<ol style="list-style-type: none">1. El Sistema rechaza la entrada y muestra un mensaje indicando que “Los datos no son válidos”
3c. El administrador selecciona la opción “Cancelar”
<ol style="list-style-type: none">1. El Sistema carga la página principal
Sección 2 “Modificar datos del usuario”
<ol style="list-style-type: none">1. El Administrador selecciona la opción “Modificar datos del usuario”2. El Sistema muestra una ventana para buscar al usuario deseado3. El Administrador introduce el usuario que desea modificar y presiona el botón

<p>“Buscar”</p> <ol style="list-style-type: none"> El Sistema muestra en la misma ventana los datos del usuario buscado El Administrador modifica los datos del usuario y presiona el botón “Guardar” El Sistema actualiza los datos introducidos y muestra un mensaje “Modificado correctamente” 		
Flujos alternos		
3a. Existen campos vacíos		
<ol style="list-style-type: none"> El Sistema muestra un mensaje indicando que “Existen campos vacíos” 		
3b. Entrada de datos no válidos		
<ol style="list-style-type: none"> El Sistema rechaza la entrada y muestra un mensaje indicando que “Los datos no son válidos” 		
5a. El administrador selecciona la opción “Cancelar”		
<ol style="list-style-type: none"> El Sistema carga la página principal 		
Sección 3 “Eliminar usuario”		
<ol style="list-style-type: none"> El Administrador selecciona la opción “Eliminar usuario” El Sistema muestra una ventana para buscar el usuario deseado El Administrador introduce el usuario y presiona el botón “ Buscar” El Sistema muestra en la misma ventana el usuario buscado El Administrador lo selecciona y presiona el botón “Eliminar” El Sistema elimina el usuario y muestra un mensaje “Eliminado correctamente” 		
Flujos alternos		
3a. El campo está vacío		
<ol style="list-style-type: none"> El Sistema muestra un mensaje “Hay campos vacíos” 		
5a. El Administrador presiona el botón “Cancelar”		
<ol style="list-style-type: none"> El Sistema muestra la página principal 		
Relaciones	CU Incluidos	No aplicable

Componente de identificación centralizada para la solución alas PACS-RIS

	CU	No aplicable
	Extendidos	
Requisitos funcionales	no	No aplicable
Asuntos pendientes		No aplicable

CU. Crear Token

Objetivo	Crear un token de seguridad que contenga la información necesaria para otorgarles a los usuarios los permisos necesarios
Actores	Solución_PACS-RIS (Inicia)
Resumen	Cuando un usuario se autentica en una de las aplicaciones se validan esos datos en el componente, se obtienen los datos necesarios del usuario en la base de datos, se cifra la información y se crea el token, que posteriormente se envía a las aplicaciones
Complejidad	Alta
Prioridad	Crítico
Precondiciones	El usuario se ha autenticado en una de las aplicaciones
Pos condiciones	Se creó el token de seguridad
Flujo de eventos	
Flujo básico "Crear Token"	
<ol style="list-style-type: none"> 1. La Solución_PACS-RIS envía la información de la autenticación de los usuarios al componente 2. El Sistema realiza las siguientes acciones: <ul style="list-style-type: none"> - Valida la información recibida - Busca el rol del usuario en a base de datos y cifra la información - Crea el token con la información cifrada - Envía el token a la Solución_PACS-RIS 3. La Solución_PACS-RIS recibe el token y otorga los permisos al usuario 4. Termina el caso de uso 	

Componente de identificación centralizada para la solución alas PACS-RIS

Flujos alternos		
1a. Existen campos vacíos		
El Sistema muestra un mensaje “Existen campos vacíos”		
1b. La información no es válida		
El Sistema muestra un mensaje “Los datos son incorrectos”		
°Relaciones	CU Incluidos	No aplicable
	CU Extendidos	No aplicable
Requisitos no funcionales	No aplicable	
Asuntos pendientes	No aplicable	

CU. Cifrar datos

Objetivo	Cifrar los datos del usuario para proteger la información que se almacena
Actores	Solución_PACS-RIS (Inicia)
Resumen	Cuando un usuario se autentica en una de las aplicaciones envía un conjunto de datos al componente, luego de comprobar la valides de estos datos el componente cifra la información con el algoritmo de cifrado md5
Complejidad	Alta
Prioridad	Crítico
Precondiciones	El usuario se ha autenticado en una de las aplicaciones
Pos condiciones	Se cifro la información
Flujo de eventos	
Flujo básico “Crear Token”	

5. La Solución_PACS-RIS envía la información de la autenticación de los usuarios al componente 6. El Sistema cifra la información 7. Termina el caso de uso		
Flujos alternos		
No aplicable		
°Relaciones	CU Incluidos	No aplicable
	CU Extendidos	No aplicable
Requisitos no funcionales	No aplicable	
Asuntos pendientes	No aplicable	

CU. Gestionar rol

Objetivo	El objetivo principal de este caso de uso consiste en gestionar los roles que intervienen en el componente
Actores	Administrador (Inicia)
Resumen	Cuando un usuario se autentica en una de las aplicaciones envía un conjunto de datos al componente, luego de comprobar la valides de estos datos el componente cifra la información con el algoritmo de cifrado md5
Complejidad	Alta
Prioridad	Crítico
Precondiciones	El usuario se a autenticado como administrador
Pos condiciones	Se insertó un nuevo rol o se eliminó un rol
Flujo de eventos	
Flujo básico "Crear Token"	

<ol style="list-style-type: none"> 1. El administrador selecciona la acción que desea realizar: <ul style="list-style-type: none"> - Si selecciona la opción Insertar rol, ir a la Sección 1 “Insertar rol” - Si selecciona la opción Eliminar rol, ir a la Sección 2 “Eliminar rol” 2. Termina el caso de uso 		
Flujos alternos		
No aplicable		
Sección 1 “Insertar rol”		
<ol style="list-style-type: none"> 1. El Administrador selecciona la opción “Insertar rol” 2. El Sistema muestra una ventana con los parámetros necesarios para adicionar el nuevo rol 3. El Administrador introduce los datos del nuevo rol y presiona el botón “Aceptar” 4. El Sistema registra los datos introducidos 		
Sección 2 “Eliminar rol”		
<p>El Administrador selecciona la opción “Eliminar rol”</p> <ol style="list-style-type: none"> 7. El Sistema muestra una ventana para buscar el rol que desea eliminar 8. El Administrador introduce el rol y presiona el botón “ Buscar” 9. El Sistema muestra en la misma ventana el rol buscado 10. El Administrador lo selecciona y presiona el botón “Eliminar” <p>El Sistema elimina el usuario</p>		
°Relaciones	CU Incluidos	No aplicable
	CU Extendidos	No aplicable
Requisitos no funcionales	No aplicable	
Asuntos pendientes	No aplicable	

Anexo 2. Diagramas de clases

1. Diagrama de clases del caso de uso insertar usuario

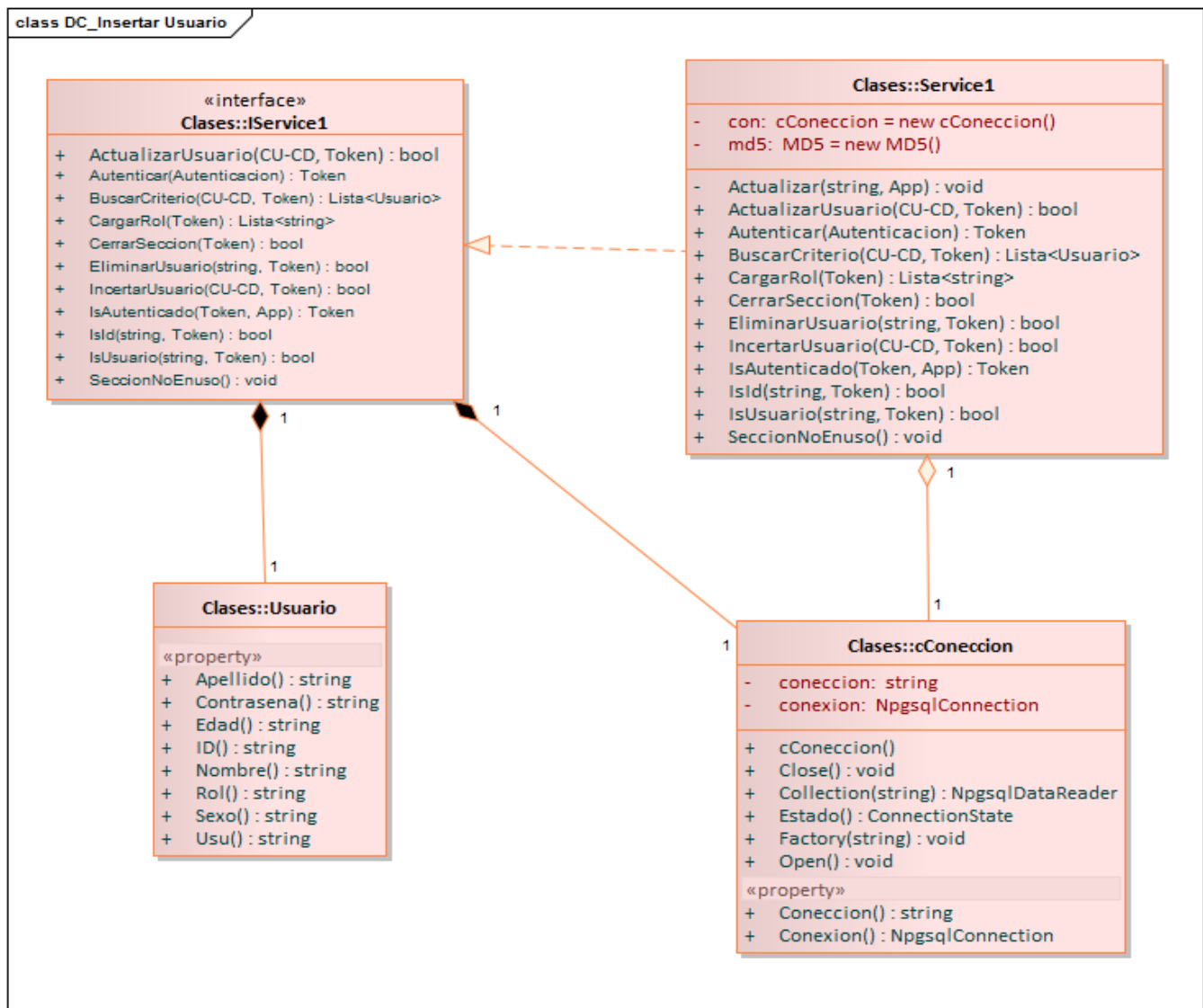


Figura 13. Diagrama de clases del caso de uso insertar usuario (fuente: creada por los autores)

2. Diagrama de clases del caso de uso eliminar usuario

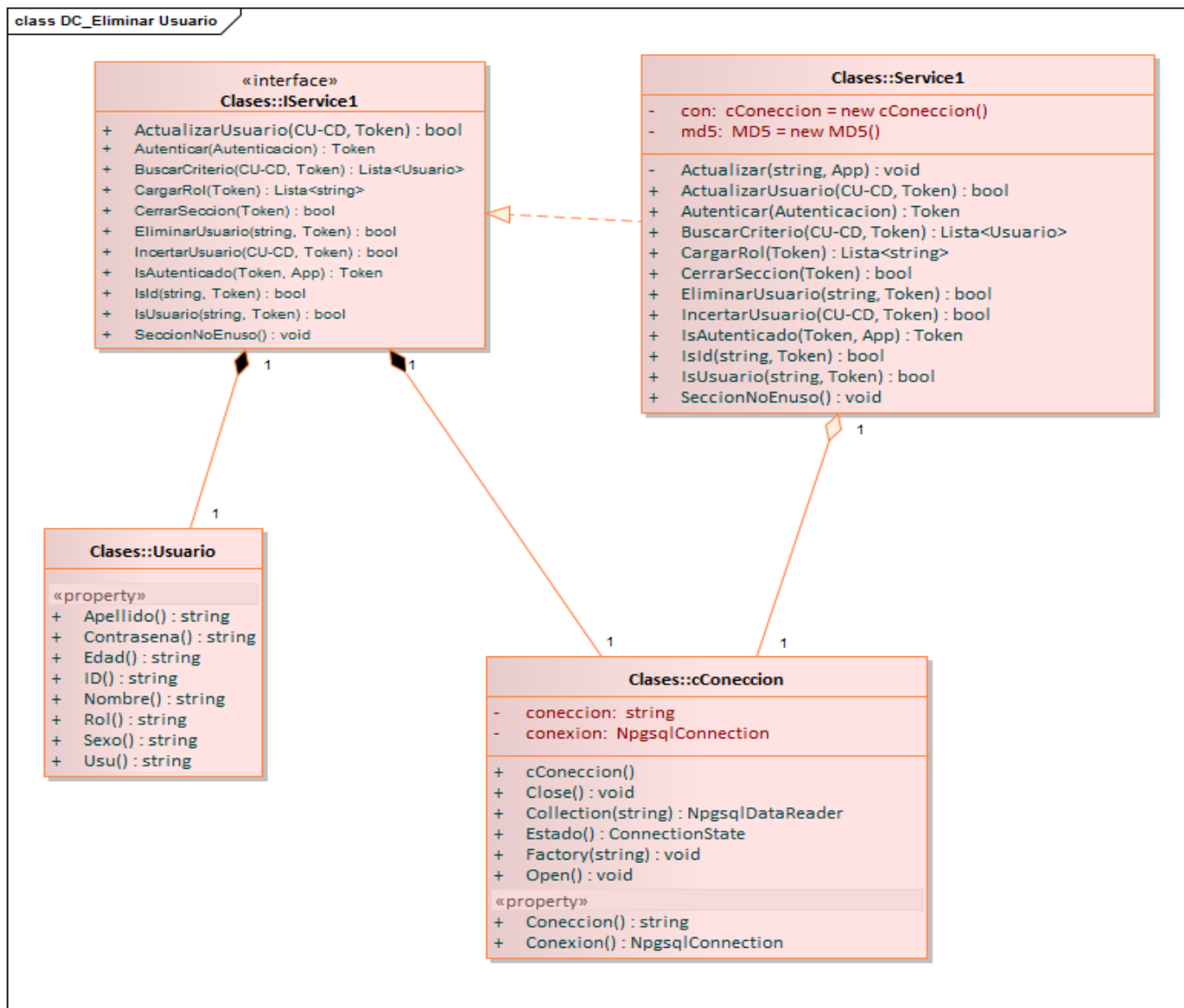


Figura 14. Diagrama de clases del caso de uso eliminar usuario (fuente: creada por los autores)

3. Diagrama de clases del caso de uso crear token

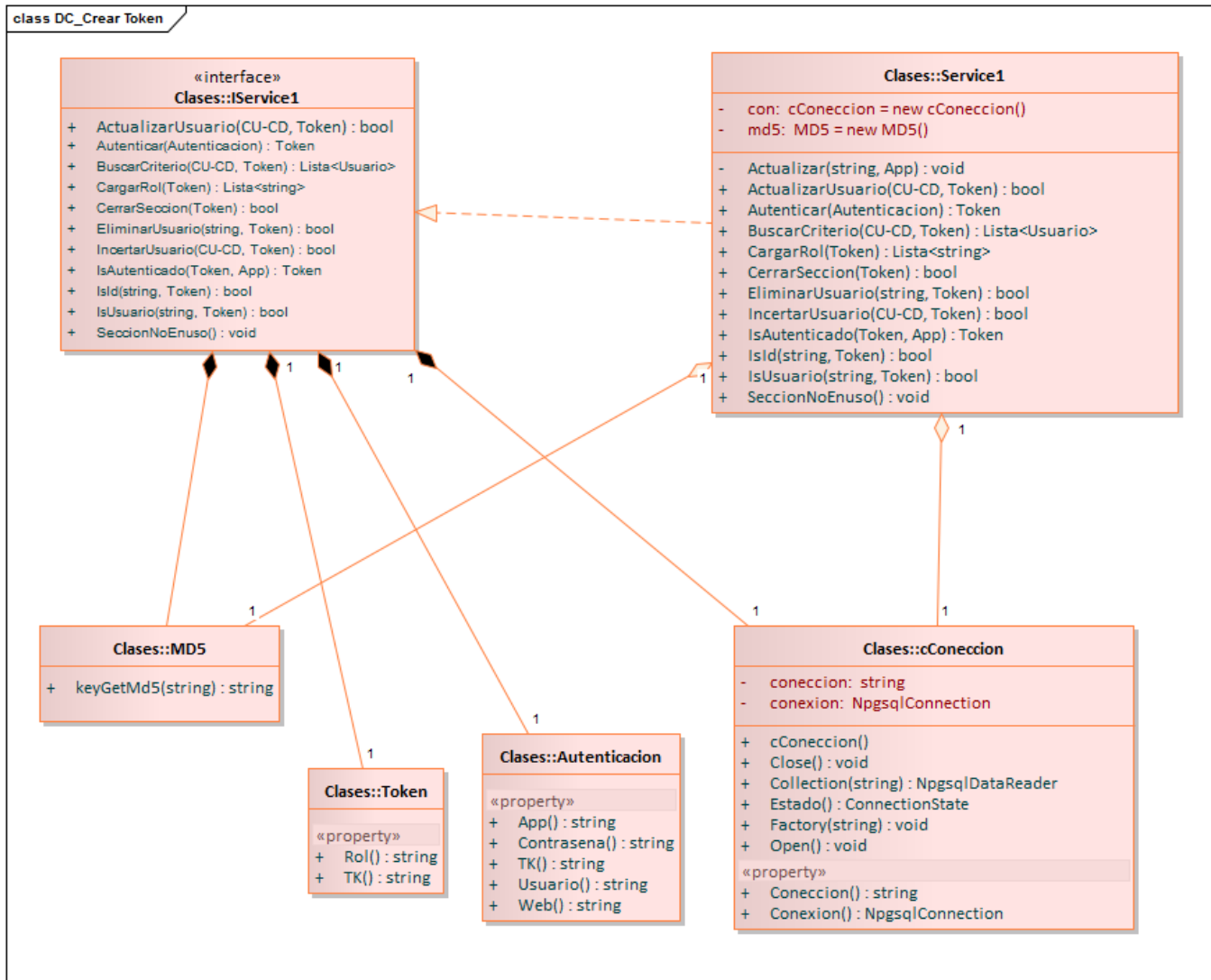


Figura 15. Diagrama de clases del caso de uso crear token (fuente: creada por los autores)

Anexo 3. Diagramas de Secuencia

CU. Crear token

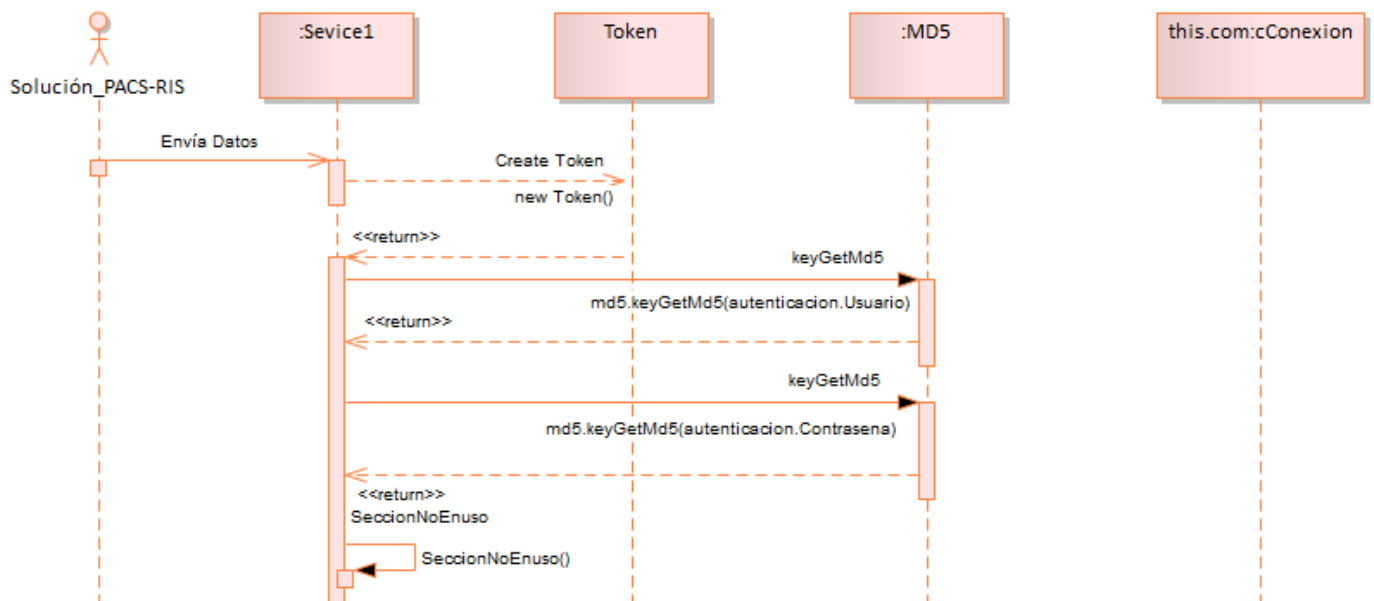


Figura 16. Diagrama de secuencia del caso de uso crear token (fuente: creada por los autores)

CU. Gestionar usuario

Insertar Usuario

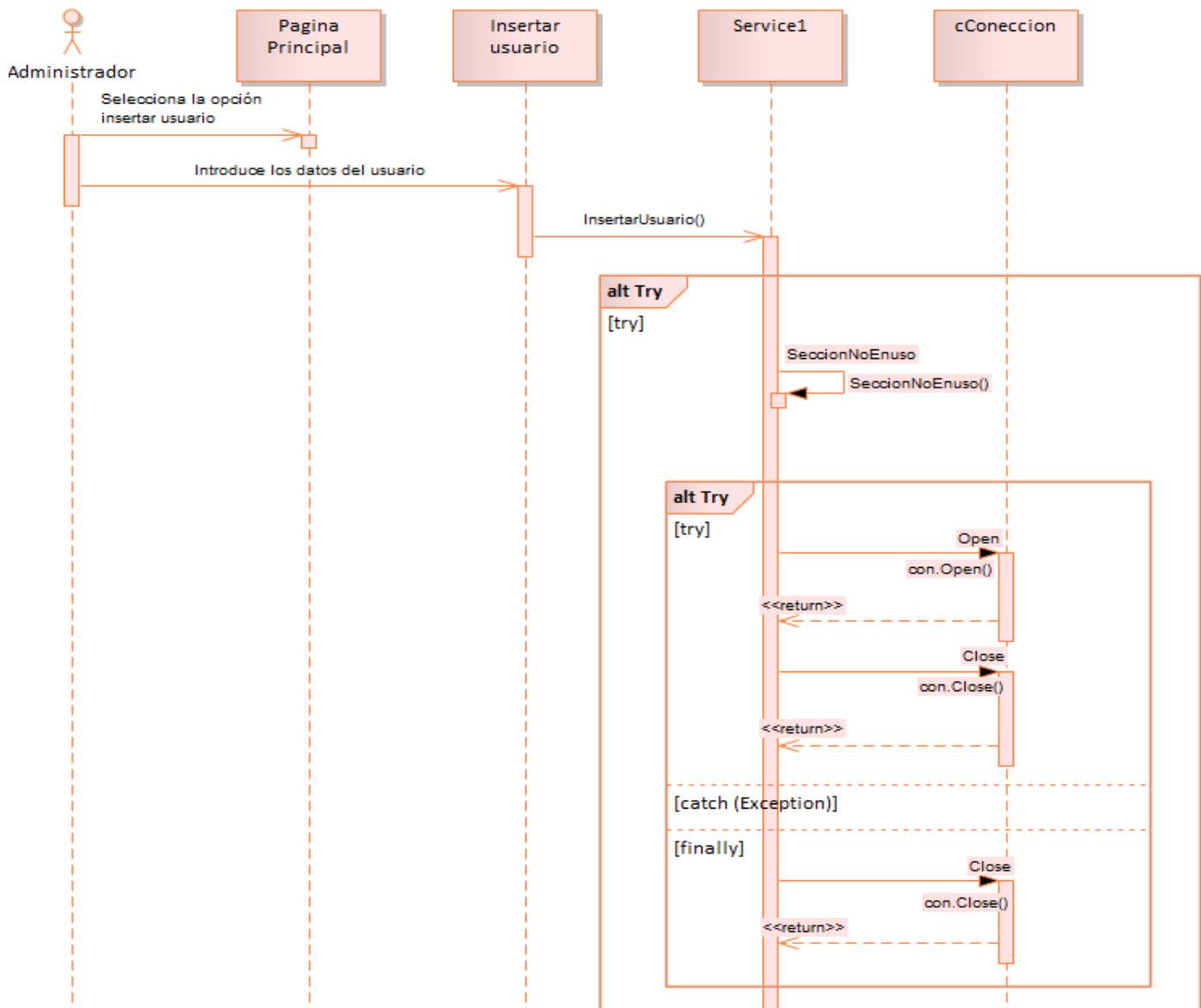


Figura 17. Diagrama de secuencia del escenario insertar usuario (fuente: creada por los autores)

Modificar usuario

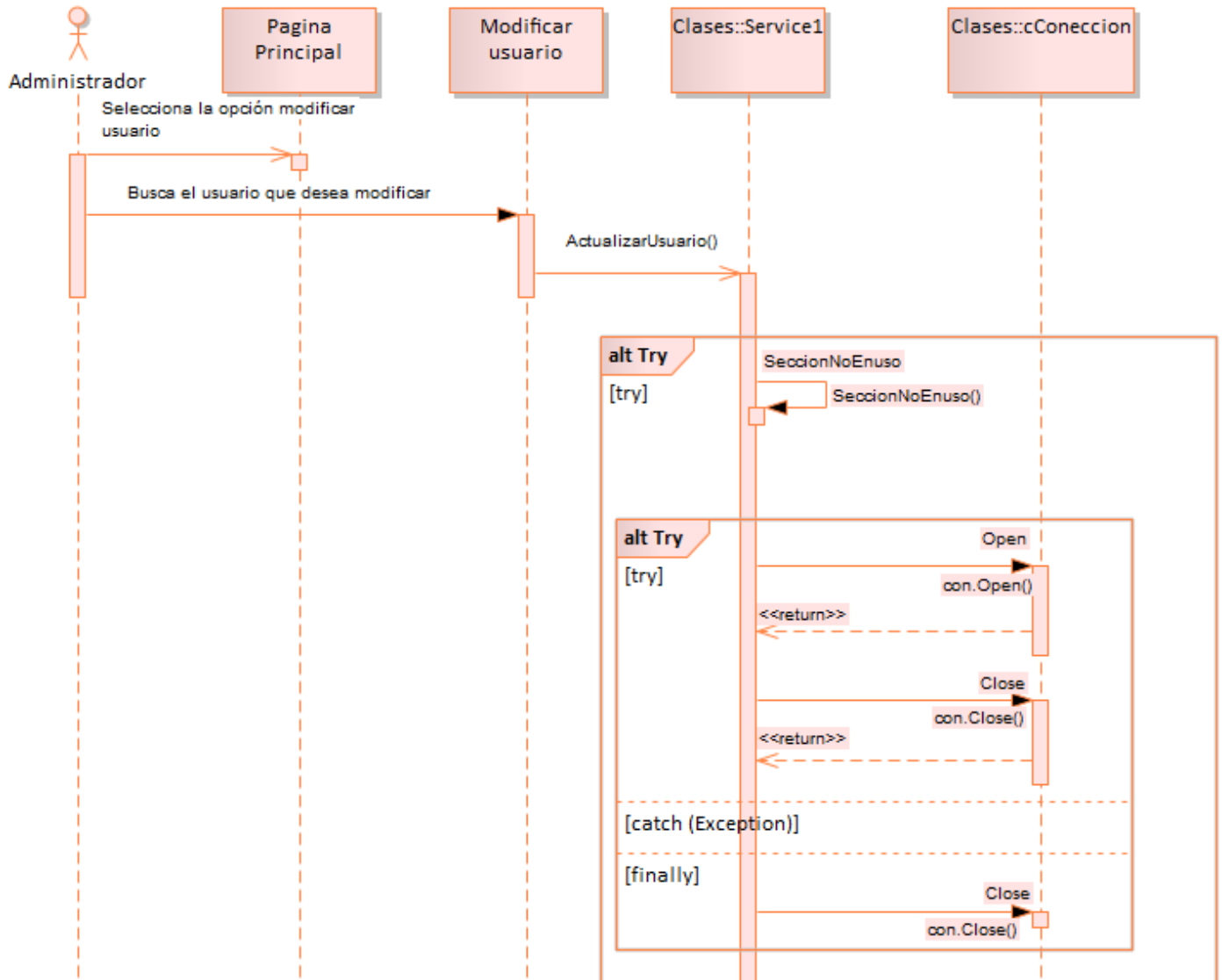


Figura 18. Diagrama de secuencia del escenario modificar usuario (fuente: creada por los autores)

Eliminar usuario

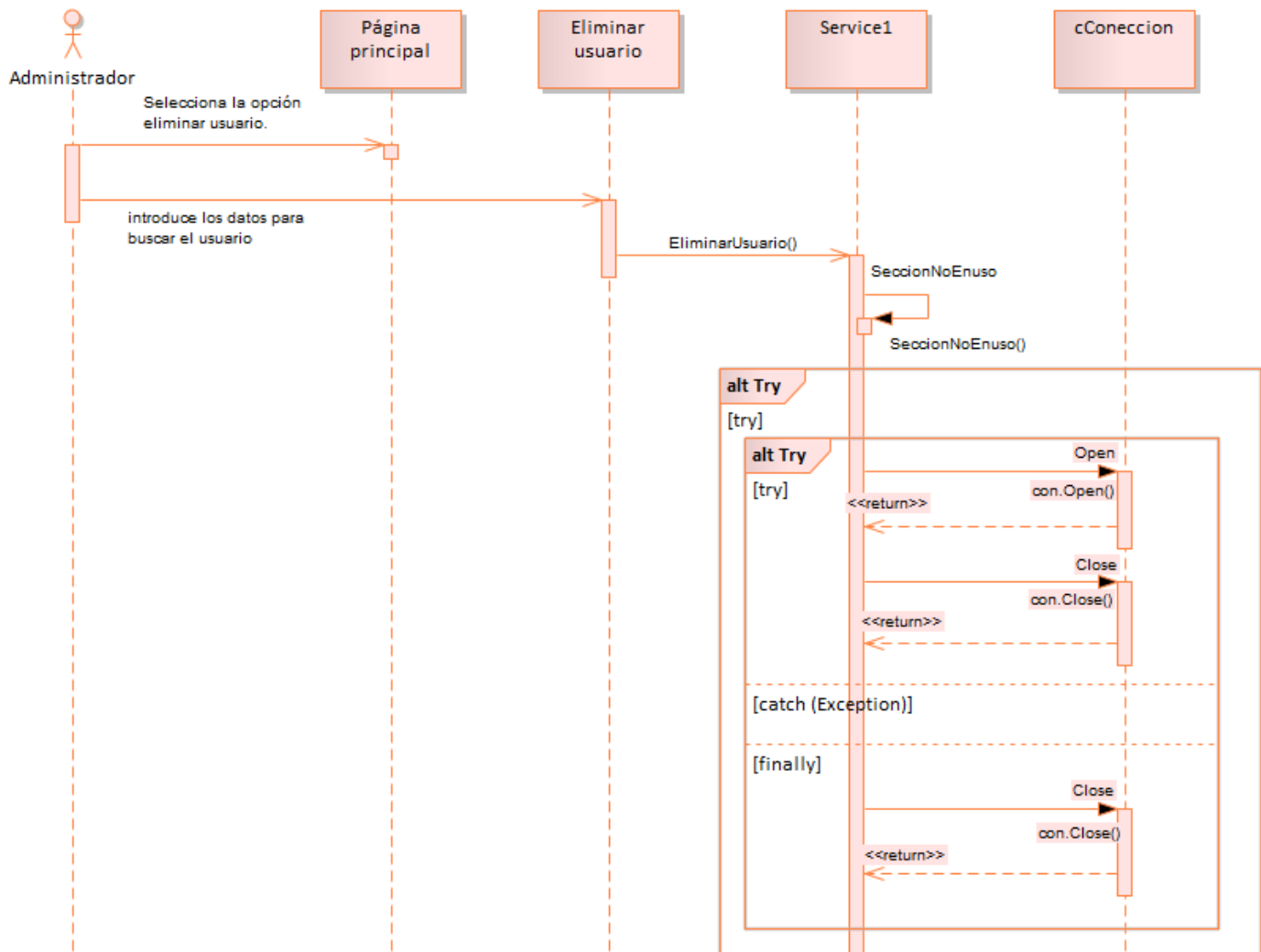


Figura 19. Diagrama de secuencia del escenario eliminar usuario (fuente: creada por los autores)

Anexo 4. WSDL del servicio web

```
- <wsdl:definitions name="Service1" targetNamespace="http://tempuri.org/">
  - <wsdl:types>
    - <xsd:schema targetNamespace="http://tempuri.org/Imports">
      <xsd:import schemaLocation="http://10.8.82.252/WCF_Tesis/Service1.svc?xsd=xsd0" namespace="http://tempuri.org"/>
      <xsd:import schemaLocation="http://10.8.82.252/WCF_Tesis/Service1.svc?xsd=xsd1" namespace="http://schemas.microsoft.com/2003/10/Serialization/">
      <xsd:import schemaLocation="http://10.8.82.252/WCF_Tesis/Service1.svc?xsd=xsd2" namespace="http://schemas.datacontract.org/2004/07/WCF_Tesis/">
      <xsd:import schemaLocation="http://10.8.82.252/WCF_Tesis/Service1.svc?xsd=xsd3" namespace="http://schemas.microsoft.com/2003/10/Serialization/Array;">
    </xsd:schema>
  </wsdl:types>
  - <wsdl:message name="IService1_Autenticar_InputMessage">
    <wsdl:part name="parameters" element="tns:Autenticar"/>
  </wsdl:message>
  - <wsdl:message name="IService1_Autenticar_OutputMessage">
    <wsdl:part name="parameters" element="tns:AutenticarResponse"/>
  </wsdl:message>
  - <wsdl:message name="IService1_IsAutenticado_InputMessage">
    <wsdl:part name="parameters" element="tns:IsAutenticado"/>
  </wsdl:message>
  - <wsdl:message name="IService1_IsAutenticado_OutputMessage">
    <wsdl:part name="parameters" element="tns:IsAutenticadoResponse"/>
  </wsdl:message>
  - <wsdl:message name="IService1_CerrarSeccion_InputMessage">
    <wsdl:part name="parameters" element="tns:CerrarSeccion"/>
  </wsdl:message>
  - <wsdl:message name="IService1_CerrarSeccion_OutputMessage">
    <wsdl:part name="parameters" element="tns:CerrarSeccionResponse"/>
  </wsdl:message>
  - <wsdl:message name="IService1_BuscarCriterio_InputMessage">
    <wsdl:part name="parameters" element="tns:BuscarCriterio"/>
  </wsdl:message>
  - <wsdl:message name="IService1_BuscarCriterio_OutputMessage">
    <wsdl:part name="parameters" element="tns:BuscarCriterioResponse"/>
  </wsdl:message>
  - <wsdl:message name="IService1_IncertarUsuario_InputMessage">
```

```
    <wsdl:part name="parameters" element="tns:IncertarUsuario"/>
  </wsdl:message>
- <wsdl:message name="IService1_IncertarUsuario_OutputMessage">
  <wsdl:part name="parameters" element="tns:IncertarUsuarioResponse"/>
</wsdl:message>
- <wsdl:message name="IService1_EliminarUsuario_InputMessage">
  <wsdl:part name="parameters" element="tns:EliminarUsuario"/>
</wsdl:message>
- <wsdl:message name="IService1_EliminarUsuario_OutputMessage">
  <wsdl:part name="parameters" element="tns:EliminarUsuarioResponse"/>
</wsdl:message>
- <wsdl:message name="IService1_ActualizarUsuario_InputMessage">
  <wsdl:part name="parameters" element="tns:ActualizarUsuario"/>
</wsdl:message>
- <wsdl:message name="IService1_ActualizarUsuario_OutputMessage">
  <wsdl:part name="parameters" element="tns:ActualizarUsuarioResponse"/>
</wsdl:message>
- <wsdl:message name="IService1_IsId_InputMessage">
  <wsdl:part name="parameters" element="tns:IsId"/>
</wsdl:message>
- <wsdl:message name="IService1_IsId_OutputMessage">
  <wsdl:part name="parameters" element="tns:IsIdResponse"/>
</wsdl:message>
- <wsdl:message name="IService1_IsUsuario_InputMessage">
  <wsdl:part name="parameters" element="tns:IsUsuario"/>
</wsdl:message>
- <wsdl:message name="IService1_IsUsuario_OutputMessage">
  <wsdl:part name="parameters" element="tns:IsUsuarioResponse"/>
</wsdl:message>
- <wsdl:message name="IService1_CargarRol_InputMessage">
  <wsdl:part name="parameters" element="tns:CargarRol"/>
</wsdl:message>
- <wsdl:message name="IService1_CargarRol_OutputMessage">
  <wsdl:part name="parameters" element="tns:CargarRolResponse"/>
</wsdl:message>
```

```
- <wsdl:message name="IService1_CargarEspecialidad_InputMessage">
  <wsdl:part name="parameters" element="tns:CargarEspecialidad"/>
</wsdl:message>
- <wsdl:message name="IService1_CargarEspecialidad_OutputMessage">
  <wsdl:part name="parameters" element="tns:CargarEspecialidadResponse"/>
</wsdl:message>
- <wsdl:message name="IService1_UsuariosConectados_InputMessage">
  <wsdl:part name="parameters" element="tns:UsuariosConectados"/>
</wsdl:message>
- <wsdl:message name="IService1_UsuariosConectados_OutputMessage">
  <wsdl:part name="parameters" element="tns:UsuariosConectadosResponse"/>
</wsdl:message>
- <wsdl:message name="IService1_CargarUsuarios_InputMessage">
  <wsdl:part name="parameters" element="tns:CargarUsuarios"/>
</wsdl:message>
- <wsdl:message name="IService1_CargarUsuarios_OutputMessage">
  <wsdl:part name="parameters" element="tns:CargarUsuariosResponse"/>
</wsdl:message>
- <wsdl:portType name="IService1">
  - <wsdl:operation name="Autenticar">
    <wsdl:input wsaw:Action="http://tempuri.org/IService1/Autenticar" message="tns:IService1_Autenticar_InputMessage"/>
    <wsdl:output wsaw:Action="http://tempuri.org/IService1/AutenticarResponse" message="tns:IService1_Autenticar_OutputMessage"/>
  </wsdl:operation>
  - <wsdl:operation name="IsAutenticado">
    <wsdl:input wsaw:Action="http://tempuri.org/IService1/IsAutenticado" message="tns:IService1_IsAutenticado_InputMessage"/>
    <wsdl:output wsaw:Action="http://tempuri.org/IService1/IsAutenticadoResponse" message="tns:IService1_IsAutenticado_OutputMessage"/>
  </wsdl:operation>
  - <wsdl:operation name="CerrarSeccion">
    <wsdl:input wsaw:Action="http://tempuri.org/IService1/CerrarSeccion" message="tns:IService1_CerrarSeccion_InputMessage"/>
    <wsdl:output wsaw:Action="http://tempuri.org/IService1/CerrarSeccionResponse" message="tns:IService1_CerrarSeccion_OutputMessage"/>
  </wsdl:operation>
  - <wsdl:operation name="BuscarCriterio">
    <wsdl:input wsaw:Action="http://tempuri.org/IService1/BuscarCriterio" message="tns:IService1_BuscarCriterio_InputMessage"/>
    <wsdl:output wsaw:Action="http://tempuri.org/IService1/BuscarCriterioResponse" message="tns:IService1_BuscarCriterio_OutputMessage"/>
  </wsdl:operation>
```

```
- <wsdl:operation name="EliminarUsuario">
  <wsdl:input wsaw:Action="http://tempuri.org/IService1/EliminarUsuario" message="tns:IService1_EliminarUsuario_InputMessage"/>
  <wsdl:output wsaw:Action="http://tempuri.org/IService1/EliminarUsuarioResponse" message="tns:IService1_EliminarUsuario_OutputMessage"/>
</wsdl:operation>
- <wsdl:operation name="ActualizarUsuario">
  <wsdl:input wsaw:Action="http://tempuri.org/IService1/ActualizarUsuario" message="tns:IService1_ActualizarUsuario_InputMessage"/>
  <wsdl:output wsaw:Action="http://tempuri.org/IService1/ActualizarUsuarioResponse" message="tns:IService1_ActualizarUsuario_OutputMessage"/>
</wsdl:operation>
- <wsdl:operation name="IsId">
  <wsdl:input wsaw:Action="http://tempuri.org/IService1/IsId" message="tns:IService1_IsId_InputMessage"/>
  <wsdl:output wsaw:Action="http://tempuri.org/IService1/IsIdResponse" message="tns:IService1_IsId_OutputMessage"/>
</wsdl:operation>
- <wsdl:operation name="IsUsuario">
  <wsdl:input wsaw:Action="http://tempuri.org/IService1/IsUsuario" message="tns:IService1_IsUsuario_InputMessage"/>
  <wsdl:output wsaw:Action="http://tempuri.org/IService1/IsUsuarioResponse" message="tns:IService1_IsUsuario_OutputMessage"/>
</wsdl:operation>
- <wsdl:operation name="CargarRol">
  <wsdl:input wsaw:Action="http://tempuri.org/IService1/CargarRol" message="tns:IService1_CargarRol_InputMessage"/>
  <wsdl:output wsaw:Action="http://tempuri.org/IService1/CargarRolResponse" message="tns:IService1_CargarRol_OutputMessage"/>
</wsdl:operation>
- <wsdl:operation name="CargarEspecialidad">
  <wsdl:input wsaw:Action="http://tempuri.org/IService1/CargarEspecialidad" message="tns:IService1_CargarEspecialidad_InputMessage"/>
  <wsdl:output wsaw:Action="http://tempuri.org/IService1/CargarEspecialidadResponse" message="tns:IService1_CargarEspecialidad_OutputMessage"/>
</wsdl:operation>
- <wsdl:operation name="UsuariosConectados">
  <wsdl:input wsaw:Action="http://tempuri.org/IService1/UsuariosConectados" message="tns:IService1_UsuariosConectados_InputMessage"/>
  <wsdl:output wsaw:Action="http://tempuri.org/IService1/UsuariosConectadosResponse" message="tns:IService1_UsuariosConectados_OutputMessage"/>
</wsdl:operation>
- <wsdl:operation name="CargarUsuarios">
  <wsdl:input wsaw:Action="http://tempuri.org/IService1/CargarUsuarios" message="tns:IService1_CargarUsuarios_InputMessage"/>
  <wsdl:output wsaw:Action="http://tempuri.org/IService1/CargarUsuariosResponse" message="tns:IService1_CargarUsuarios_OutputMessage"/>
</wsdl:operation>
</wsdl:portType>
- <wsdl:binding name="BasicHttpBinding_IService1" type="tns:IService1">
  <soap:binding transport="http://schemas.xmlsoap.org/soap/http"/>
```



```
- <wsdl:operation name="Autenticar">
  <soap:operation soapAction="http://tempuri.org/IService1/Autenticar" style="document"/>
  - <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  - <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
- <wsdl:operation name="IsAutenticado">
  <soap:operation soapAction="http://tempuri.org/IService1/IsAutenticado" style="document"/>
  - <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  - <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
- <wsdl:operation name="CerrarSeccion">
  <soap:operation soapAction="http://tempuri.org/IService1/CerrarSeccion" style="document"/>
  - <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  - <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
- <wsdl:operation name="BuscarCriterio">
  <soap:operation soapAction="http://tempuri.org/IService1/BuscarCriterio" style="document"/>
  - <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  - <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
```

```
</wsdl:operation>
- <wsdl:operation name="IncertarUsuario">
  <soap:operation soapAction="http://tempuri.org/IService1/IncertarUsuario" style="document"/>
  - <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  - <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
- <wsdl:operation name="EliminarUsuario">
  <soap:operation soapAction="http://tempuri.org/IService1/EliminarUsuario" style="document"/>
  - <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  - <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
- <wsdl:operation name="ActualizarUsuario">
  <soap:operation soapAction="http://tempuri.org/IService1/ActualizarUsuario" style="document"/>
  - <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  - <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
- <wsdl:operation name="IsId">
  <soap:operation soapAction="http://tempuri.org/IService1/IsId" style="document"/>
  - <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  - <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
```

```
    </wsdl:output>
  </wsdl:operation>
- <wsdl:operation name="IsUsuario">
  <soap:operation soapAction="http://tempuri.org/IService1/IsUsuario" style="document"/>
  - <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  - <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
- <wsdl:operation name="CargarRol">
  <soap:operation soapAction="http://tempuri.org/IService1/CargarRol" style="document"/>
  - <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  - <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
- <wsdl:operation name="CargarEspecialidad">
  <soap:operation soapAction="http://tempuri.org/IService1/CargarEspecialidad" style="document"/>
  - <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  - <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
- <wsdl:operation name="UsuariosConectados">
  <soap:operation soapAction="http://tempuri.org/IService1/UsuariosConectados" style="document"/>
  - <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  - <wsdl:output>
```

```
</wsdl:operation>
- <wsdl:operation name="CargarEspecialidad">
  <soap:operation soapAction="http://tempuri.org/IService1/CargarEspecialidad" style="document"/>
  - <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  - <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
- <wsdl:operation name="UsuariosConectados">
  <soap:operation soapAction="http://tempuri.org/IService1/UsuariosConectados" style="document"/>
  - <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  - <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
- <wsdl:operation name="CargarUsuarios">
  <soap:operation soapAction="http://tempuri.org/IService1/CargarUsuarios" style="document"/>
  - <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  - <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
</wsdl:binding>
- <wsdl:service name="Service1">
  - <wsdl:port name="BasicHttpBinding IService1" binding="tns:BasicHttpBinding IService1">
    <soap:address location="http://10.8.82.252/WCF_Tesis/Service1.svc"/>
  </wsdl:port>
</wsdl:service>
</wsdl:definitions>
```