



Pontificia Universidad Católica de Valparaíso

Facultad de Ingeniería

Escuela de Ingeniería Informática

Ingeniería de Ejecución en Informática

**SISTEMA DE GESTIÓN PARA LA OIRS Y DE
SOLICITUDES DE ATENCIÓN PARA EL SERVICIO
DE SALUD VIÑA DEL MAR-QUILLOTA**

Autores:

Luis Angelo Astete Donoso - Pablo Andrés Maldonado Blanche

Informe final del Proyecto para optar al Título profesional de

Ingeniero de Ejecución en Informática

Profesor guía:

Silvana Roncagliolo de la Horra

DICIEMBRE 2007

“Dedicado a mi Madre por su amor, comprensión y apoyo en cada etapa importante de mi vida”.

Luis Astete Donoso

“Dedicado a mis padres y familia, porque sin ellos hubiese sido imposible recorrer el camino para alcanzar esta anhelada meta”.

Pablo Maldonado Blanche

Agredecimientos

“Agradezco a Dios, mi Madre y familia por apoyarme durante toda la vida, y a la Universidad por los conocimientos y valores entregados”.

Luis Astete Donoso

Agredecimientos

“Agradezco a Dios, mis padres y familia por el amor, apoyo y confianza entregada durante toda mi vida”.

Pablo Maldonado Blanche

Glosario

- **Caja Blanca:** es un tipo de pruebas permiten examinar la estructura interna del programa.

- **Caja Negra:** es un tipo de pruebas que se llevan a cabo sobre la interfaz del software, y es completamente indiferente al comportamiento interno y la estructura del programa.
- **Centro de Responsabilidad:** Es un conjunto de servicios que tienen características comunes, por ejemplo el Centro de Responsabilidad Cardiología se puede componer de los servicios de Cirugía Cardiovascular, servicio de UTI y servicio de Hemodinámica.
- **Estamento:** representa un tipo de funcionario del recinto hospitalario, por ejemplo: enfermeras, médicos, auxiliares, paramédicos, administrativos, etc.
- **OIRS:** Oficina de Informaciones, Reclamos y Sugerencias.
- **Orientación a Objetos:** metodología de diseño de software que modela la características de objetos reales a través de la utilización de clases y objetos.
- **PHP (PHP Hypertext Preprocessor):** lenguaje "Open Source" interpretado de alto nivel, especialmente pensado para desarrollos Web y el cual puede ser embebido en páginas HTML.
- **PostgreSQL:** motor de base de datos "Open Source", liberado bajo la licencia BSD.
- **Proceso Unificado:** es un marco de desarrollo de software iterativo e incremental que puede ser adaptado a organizaciones o proyectos específicos.
- **QUIS (Questionnaire for User Interface Satisfaction):** cuestionario estándar que analiza el cumplimiento de los requerimientos de las interfaces del sistema.
- **RFS:** Reclamo, felicitación o sugerencia.
- **SSVQ:** Servicio de Salud Viña del Mar – Quillota.
- **SUMI (Software Usability Measurement Inventory):** cuestionario estándar que tiene como objetivo valorar la calidad de uso del sistema, a través de un inventario de medidas de usabilidad.

- **UML (Unified Modeling Language):** es un lenguaje gráfico utilizado para visualizar, especificar, construir y documentar un sistema de software.

Resumen

El presente informe constituye a la documentación asociada a los sistemas desarrollados para el Servicio de Salud Viña del Mar-Quillota, denominado formalmente como “Sistema de Gestión para la OIRS y de Solicitudes de Atención para el Servicio de Salud Viña del Mar-Quillota.”.

El documento comienza describiendo la organización y la situación actual, midiendo factibilidad y seleccionando metodologías, paradigmas y herramientas. Junto a esto se describen y especifican los requerimientos que permiten definir las funcionalidades más importantes del sistema.

Posteriormente se tiene la implementación misma de los sistemas, definiendo en mayor detalle la arquitectura y las interfaces.

Abstract

This report constitutes the documentation associated to the systems developed for Viña del Mar-Quillota Health Service, called formally as “Management System for OIRS and of Requests of Attention for Viña del Mar-Quillota Health Service”.

The document starts describing the organization and the actual situation, feasibility measuring and selecting methodologies, paradigms and tools. With they describe and specify the requirements which let define the system’s more important functionalities.

Later the same implementation of the system is had, defining in greater detail the architecture and the interfaces.

1. Introducción

En la actualidad es sabido por todos que el atributo más importante a la hora de determinar la posición de una empresa en el largo plazo, es la opinión de los clientes sobre los productos o servicios que estos adquieren. De esta manera, es lógico que para que estos se formen una opinión positiva de la empresa, ésta debe satisfacer con todas las necesidades y expectativas de los clientes. A esto se le denomina calidad de servicio.

La calidad, y más específicamente la calidad de servicio, se ha convertido hoy por hoy en un requisito indispensable para competir en las organizaciones industriales y comerciales de todo el mundo. Sin embargo, mejorar la calidad de servicio no es sólo una preocupación para las empresas privadas, sino también a nivel gubernamental. Esto se ve reflejado en el incremento de políticas que ayudan a entregar un mejor servicio a los ciudadanos y además a los mismos trabajadores de entidades públicas, con el fin de obtener mejoras en los procesos internos y de productividad de éstas, como también en la satisfacción de la ciudadanía al recibir un servicio ágil, confiable y eficiente. Es así como la calidad de servicio se convierte en un elemento estratégico fundamental.

Con este afán, el Servicio de Salud Viña del Mar – Quillota (SSVQ) lanzó propuestas para mejorar la calidad de servicio prestado a los pacientes y usuarios en general de los distintos servicios de la red de hospitales de la V región, más específicamente en cómo se acogen las inquietudes, reclamos, felicitaciones o sugerencias que estos presenten y en la respuesta y solución que se les da a estos, como también entre quienes trabajan en estos centros de salud en la realización de solicitudes de atención a los distintos departamentos de manutención.

Así nacen el **Sistema para la Oficina de Informaciones, Reclamos y Sugerencias (OIRS)** y el **Sistema de Solicitudes de Atención**, los cuales se detallarán en este informe, conociendo a fondo su problemática actual y cuales son las soluciones planteadas para su resolución.

2. Descripción del SSVQ

El Servicio de Salud Viña del Mar-Quillota (SSVQ) [1] tiene una extensión geográfica de 7.506 km² y una población asignada de 863.923 habitantes los cuales, en su mayoría, se encuentran concentrados en localidades urbanas.

Dentro de la jurisdicción del SSVQ existen 17 comunas correspondientes a las provincias de Petorca, Quillota y Valparaíso; esta última compartida con el Servicio de Salud Valparaíso – San Antonio.

El Servicio de Salud Viña del Mar-Quillota es uno de los tres Servicios de Salud ubicados en la Quinta Región. En la figura 1, se muestra el establecimiento del SSVQ.



Figura 1: Servicio de Salud Viña del Mar – Quillota.

Uno de los principales problemas estructurales que afecta la atención de salud en el SSVQ es el déficit de recursos humanos. La tasa actual de personal en relación a la población es de 4 por mil habitantes, siendo que en el Servicio de Salud Valparaíso - San Antonio es de 7,7 funcionarios por cada mil habitantes y en el Servicio de Salud Aconcagua es de 6,2 por cada mil habitantes.

Dentro de su red asistencial se encuentran los siguientes hospitales que dependen de su gestión. Estos serán clasificados de acuerdo a su “tamaño” y la dependencia de ellos en cuanto al soporte que les brinda el SSVQ:

Hospitales grandes:

Estos hospitales cuentan con sus propios departamentos de informática y manutención, por lo que cuando necesitan de algún tipo de soporte este no es solicitado al SSVQ, sino que sus propios departamentos se encargan de ese rol. Estos hospitales son:

- Hospital Dr. Gustavo Fricke, de Viña del Mar.
- Hospital San Martín de Quillota.
- Hospital de Quilpué.

Hospitales pequeños:

Estos hospitales, a diferencia de los “grandes”, no cuentan con los departamentos de informática y manutención, y dependen exclusivamente del soporte que les pueda brindar el SSVQ. Estos hospitales son:

- Hospital Juana Ross, de Peñablanca.
- Hospital Adriana Cousiño, de Quintero.
- Hospital Santo Tomás, de Limache.
- Hospital San Agustín, de La Ligua.
- Hospital Dr. Mario Sánchez, de La Calera.
- Hospital Dr. Víctor Moll, de Cabildo.
- Hospital de Petorca.
- Hospital Geriátrico Paz de la Tarde, de Limache.

2.1. Organización del SSVQ

El Servicio de Salud para cumplir con su Misión y sus Políticas ha modificado la estructura de la organización existente a partir de 1994. Lo anterior, implica sectorizar el territorio del Servicio en cuatro áreas, coincidentes con la división político-administrativa, constituyéndose en el “Departamento de Planificación de Apoyo a la Gestión”. En cada uno de ellos se cuenta con un equipo multidisciplinario encargado de coordinar la acción de los diferentes niveles de atención y de la implementación de los Programas de Salud avanzando, en esta forma, hacia la efectiva integración de la red asistencial.

Se ha constituido además, un “Consejo Técnico Sectorial de Salud” por sector geográfico, en el que participan todos los actores de salud tanto del Servicio como del área municipal, con el fin de tomar las decisiones en relación a los problemas de la salud de las personas y del ambiente de cada área territorial, de manera coordinada y participativa.

Se han formado también los “Consejos Técnicos Intersectoriales” para obtener la adecuada coordinación en el tratamiento de los problemas que requieren un enfoque multisectorial.

2.2. Misión

El SSVQ tiene como misión gestionar y liderar la articulación y desarrollo de la Red Asistencial que constituye, y las macroredes que integra, para responder en forma equitativa, oportuna, eficiente, eficaz y de calidad a la demanda de los usuarios, a través de acciones integrales de salud.

2.3. Visión

El SSVQ tiene como visión ser un Gestor de Redes Asistenciales de excelencia que asegure una atención de salud integrada con equipos humanos multidisciplinarios, motivados, comprometidos y competentes velando por los principios de equidad, solidaridad, participación y calidad para lograr los objetivos sanitarios locales, regionales y nacionales.

3. Objetivos

3.1. Objetivo General

Dada la naturaleza del proyecto, el cual comprende el desarrollo de 2 sistemas, se ha individualizado un objetivo general para cada sistema.

- Desarrollar un sistema para la OIRS (Oficina de Información, Reclamos y Sugerencias) que le permita gestionar y llevar un registro de las inquietudes realizadas por los pacientes de los recintos hospitalarios, con el objeto de brindarles una mejor atención.
- Desarrollar un sistema para los departamentos de informática y manutención del SSVQ que les facilite la recepción de las solicitudes realizadas por los funcionarios internos, permitiéndoles una mejor gestión de las actividades llevadas a cabo.

3.2. Objetivos Específicos

Los objetivos específicos tienen como finalidad contribuir al cumplimiento de los objetivos generales y con ello contribuir al desarrollo del proyecto. Estos son:

- Permitir a la OIRS dar una mejor atención a los usuarios y pacientes de los recintos hospitalarios.
- Permitir a los departamentos de informática y manutención del SSVQ dar un soporte a los hospitales pequeños que no cuentan con estos en sus establecimientos.
- Apoyar la toma de decisiones de la Dirección del SSVQ y de los recintos hospitalarios, entregándoles informes estadísticos que reflejen de manera manifiesta el funcionamiento general de cada uno de estos.
- Entregar un medio ágil y de fácil acceso a los usuarios, que les permita contactarse rápidamente y mantenerse informados del estado de sus diligencias.

4. Antecedentes Preliminares

4.1. Situación Actual

Hoy en día, las OIRS de los distintos hospitales pertenecientes al SSVQ no cuentan con un sistema que les permita llevar un registro y control de las RFS, todo es hecho manualmente mediante libros ubicados en cada una de las unidades de los centros hospitalarios y otro en la misma OIRS. Esto implica un tedioso proceso de recopilación de la información y de posterior respuesta al usuario. A su vez, la dirección del SSVQ y de los recintos hospitalarios no pueden tomar decisiones en torno a las diferentes inquietudes que puedan tener los usuarios, puesto que a los datos no se les realiza un análisis profundo que les permita ver, de forma clara, cuales son las unidades donde se presentan mayores falencias en lo que se refiere a atención a los usuarios o, en su defecto, las virtudes que éstas puedan tener.

En cada hospital del SSVQ existe una OIRS, la cual se encarga de acoger las RFS (Reclamo, Felicitación o Sugerencia) que han sido dejadas por escrito en cada uno de los servicios clínicos con los que cuenta el hospital. Actualmente en cada servicio existen libros donde los usuarios o pacientes del hospital dejan por escrito la RFS. Luego, al final del día, las hojas son retiradas de los libros y llevadas a la OIRS, donde son transcritas a una planilla Excel, con el nombre de la persona, la fecha y la unidad a la cual se refiere la RFS e indicando además el tipo de ésta. Una vez que es registrada, existe un periodo de tiempo de 20 días hábiles donde se le debe dar una respuesta a cada una de ellas. Para ello las RFS son derivadas a los respectivos servicios clínicos donde éstas fueron hechas, donde el jefe del servicio tiene la misión de responder a la RFS hecha por el usuario en el tiempo indicado, para que posteriormente sea entregada a éste a través de la Dirección del hospital. A su vez, la dirección del SSVQ y de los hospitales no cuentan con la información necesaria para saber si las inquietudes de los pacientes están siendo solucionadas o tomadas en cuenta, siendo ésta necesaria para tomar las decisiones adecuadas cuando corresponda, y

los pacientes por su lado tampoco pueden llevar un seguimiento del estado de su RFS (en los casos de reclamos y sugerencias). La figura 2 muestra el funcionamiento actual de la OIRS.

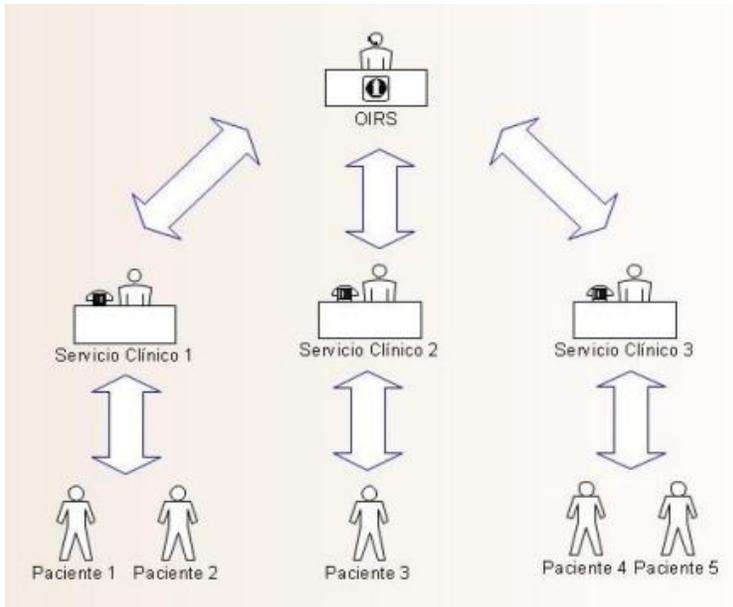


Figura 2: Funcionamiento OIRS.

Por otro lado, no existe una comunicación formal entre los hospitales pequeños y los departamentos de informática y manutención del SSVQ. Actualmente, ésta se realiza telefónicamente, entre el solicitante y el jefe encargado de la unidad de manutención y/o informática, pero sin existir un sistema que permita llevar un registro de estas solicitudes, ni tampoco de los gastos asociados a los materiales proporcionados, ni el personal asignado a la asistencia técnica.

El SSVQ cuenta con un departamento de Informática y Manutención que se encargan de dar soporte a los distintos hospitales pequeños que no cuentan con estos departamentos en sus instalaciones. Los hospitales grandes, por su parte, tienen sus propios departamentos, por lo que no dependen necesariamente del SSVQ en esta materia. Actualmente existe una aplicación Access monousuario muy limitada en lo que se refiere a la información que es almacenada en ella, la cual no ofrece una interacción con los distintos servicios clínicos, y sólo se encuentra implementada en el Hospital Gustavo Fricke. El problema radica en que esta aplicación sólo se encuentra en el departamento de Informática y Manutención,

ocasionando que la persona que tenga un problema en su lugar de trabajo deba comunicarse telefónicamente con uno de estos departamentos y explicárselo. Luego, de acuerdo a la información proporcionada por el funcionario, ésta es ingresada en el sistema para determinar las acciones a seguir, como las personas que serán asignadas a la tarea por parte del jefe del departamento y los materiales a ser ocupados en ésta. Todo este proceso sería mucho más ágil si en cada uno de los servicios los usuarios pudieran acceder a una aplicación donde ellos mismos pudieran ingresar la solicitud, la que posteriormente fuese vista por el departamento de Informática o Manutención. Algo similar ocurre con los hospitales pequeños, donde existe un funcionario que se encarga de coordinar los tipos de peticiones y hacérselos llegar al correspondiente departamento en el SSVQ. Además, no existe registro de los activos reparados, de manera de determinar las veces que un mueble o un equipo informático han sido reparados con anterioridad y el tipo de falla que presentase, con el fin de poder establecer si es más conveniente o no la compra de un nuevo equipo o mueble. En la Figura 3 se puede observar la función que cumple el SSVQ con respecto a los hospitales pequeños.

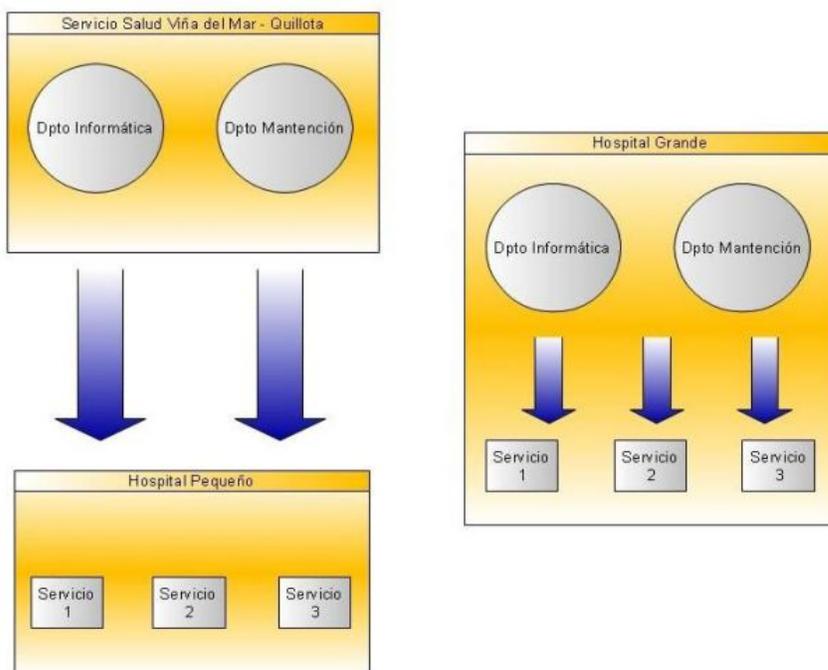


Figura 3: Función que cumple el SSVQ.

5. Soluciones Propuestas

Luego de conocer la situación actual, tanto de la OIRS como de los departamentos de Informática y Manutención del SSVQ, se desarrollarán 2 sistemas Web, los cuales permitirán, entre otras cosas, el acceso remoto a los usuarios ubicados en los distintos recintos hospitalarios pertenecientes a la red del SSVQ. De esta manera, se mantendrá el desarrollo orientado a ambientes Web, basado en lenguaje PHP y PostgreSQL como motor de base de datos, establecido por el SSVQ, ofreciendo interfaces claras y amigables que faciliten la navegación por cada sistema. Estos tendrán las siguientes características:

a) Sistema OIRS

- **Agilizar la recepción de RFS y el seguimiento de éstas mediante portal de acceso libre:** Los usuarios que deseen constatar un RFS podrán hacerlo ingresando al sitio Web del SSVQ desde cualquier computador con acceso a Internet. Asimismo, podrán realizar un seguimiento a la acogida de su RFS y ver el plazo máximo de respuesta luego de haberla enviado.
- **Controlar los tiempos de respuesta:** El sistema será capaz de mostrar aquellas RFS que se encuentren en el límite del tiempo establecido para dar una respuesta y/o las que eventualmente haya vencido su plazo, con el objeto de darle una respuesta urgente.
- **Generar informes estadísticos a la Dirección del SSVQ/Hospital:** Con esto, se podrá evaluar la atención prestada en cada recinto hospitalario rápida y comprensiblemente, con el fin de que la Dirección pueda tomar decisiones de carácter administrativo.
- **Administración de usuarios:** El administrador podrá agregar, modificar o eliminar funcionarios del sistema.

b) Sistema Solicitudes de Atención

- **Permitir la prestación de servicios a todas las unidades hospitalarias:** Las unidades sólo se preocuparán de realizar la solicitud, ver el estado y el cumplimiento de ésta.
- **Registrar y generar informes sobre el historial de reparaciones:** Al conocer el historial de reparaciones, los directivos podrán evaluar la situación de un activo determinado y, de esta manera, determinar la conveniencia entre seguir reparándolo o adquirir uno nuevo.
- **Registrar los gastos por reparaciones asociados al uso de materiales:** Los materiales usados en las reparaciones deben ser registrados por parte del encargado, para así poder establecer el costo económico del trabajo realizado y el total del gasto mensual.
- **Generar informes:** Con esto el Jefe del departamento podrá obtener información que le permita una mejor toma de decisiones.
- **Administración de usuarios:** El Jefe del departamento podrá agregar, modificar o eliminar funcionarios del sistema. En el caso del Jefe de mantenimiento podrá realizar estas funciones sólo con los trabajadores de su departamento.

6. Análisis de Factibilidad

En el análisis de factibilidad se estudian los siguientes tipos:

- Factibilidad Técnica
- Factibilidad Legal.
- Factibilidad Económica.
- Factibilidad Operacional.

Cabe mencionar que el análisis realizado se basó con información entregada por el jefe del departamento de informática del SSVQ.

6.1. Factibilidad Técnica

En este apartado se estudian los recursos tecnológicos necesarios para la implementación del sistema. La factibilidad técnica tiene por objeto determinar si el problema tiene una posible solución, respecto a los recursos computacionales y tecnológicos que se encuentran disponibles o pueden ser adquiridos en una organización.

6.1.1. Necesidad y Disponibilidad de Recursos

Los recursos necesarios y disponibles para la realización de este proyecto se consideran de acuerdo a tres tipos:

- Recursos de Software.
- Recursos de Hardware.
- Recursos Humanos.

6.1.2. Recursos de Software

Los recursos de software con los que se dispone y que serán considerados a utilizar en la realización de este proyecto y su posterior puesta en marcha, se presentan en la Tabla 1.

Tabla 1: Recursos de Software.

Análisis y Diseño	<ul style="list-style-type: none">▪ Microsoft Office Visio 2003▪ Microsoft Office Word 2003▪ Microsoft Office Project 2003▪ Power Designer
Implementación	<ul style="list-style-type: none">▪ Servidor Web Apache▪ PHP 4.0▪ PostgreSQL 8.0▪ CentOS
Pruebas	<ul style="list-style-type: none">▪ Windows XP Professional▪ PostgreSQL

6.1.3. Recursos de Hardware

Para el desarrollo del proyecto se cuenta con dos computadores personales, equipados con el hardware apropiado para utilizar las herramientas necesarias para el desarrollo de los sistemas. Además se dispone de los computadores pertenecientes a la Escuela de Ingeniería Informática, que también cuentan con los recursos necesarios para el desarrollo.

La implementación del sistema se llevará a cabo en un servidor ubicado en las dependencias del departamento de informática del SSVQ, que posee las siguientes características:

- Procesador Pentium IV 2400 Ghz.
- 2GB RAM.
- Disco Duro 160 GB.

6.1.4. Recursos Humanos

Los desarrolladores del sistema son alumnos de último año de la carrera de Ingeniería de Ejecución en Informática, con experiencia en las metodologías y herramientas a utilizar. Esto, más lo aprendido a lo largo de la carrera y una investigación permanente colaborarán con el correcto desarrollo, funcionamiento del sistema y puesta en marcha, considerando la instalación de las aplicaciones y la capacitación a los usuarios.

6.2. Factibilidad Legal

En este punto se analizó que el proyecto se enmarcara dentro de los decretos y leyes establecidas, ya sea en lo que se refiere a los sistemas informáticos, como a la utilización de información que pueda resultar confidencial.

Las aplicaciones serán desarrolladas en PHP y PostgreSQL como gestor de base de datos. Ambos son de código abierto, por lo que no representan problema en el aspecto legal. También se utilizarán herramientas de Microsoft que se encuentran en la Escuela de Ingeniería Informática, lugar donde se desarrollará el proyecto, las cuales cuentan con todas las licencias respectivas. Para la puesta en marcha se utilizará CentOS, el cual es un sistema operativo de código abierto, por lo que su uso no representa impedimento legal alguno.

6.3. Factibilidad Económica

Para el análisis de la factibilidad económica fueron considerados 3 puntos: desarrollo, elaboración y puesta en marcha.

No existen costos asociados al desarrollo del proyecto, es decir, desarrollo de las aplicaciones, elaboración de documentos, suministros (Internet, electricidad, etc.), pago de personal, etc., puesto que estos serán llevados a cabo por los mismos alumnos en instalaciones de la Escuela de Ingeniería en Informática.

Con respecto a la elaboración del proyecto, tampoco tiene costos asociados, debido a que, tanto el hardware como el software que se utilizará para la elaboración del proyecto ya ha sido adquirido por la Escuela de Ingeniería en Informática.

Por último, la puesta en marcha de este proyecto tiene, al igual que los puntos anteriores, costo \$ 0, ya que el SSVQ ya cuenta con los equipos necesarios para la implantación y posterior puesta en marcha de los sistemas, como además de las licencias necesarias para su funcionamiento legal.

6.4. Factibilidad Operacional

El sistema ayudará a agilizar y facilitar las labores, la comunicación y la administración dentro del SSVQ, como también de los usuarios que se involucran con éste. Los usuarios dentro del establecimiento tienen conocimientos básicos e intermedios en computación y, además, experiencia en el uso de sistemas informáticos. Los demás usuarios que utilicen el sistema no tendrán inconvenientes en hacerlo, ya que el sistema contará con interfaces amigables, de fácil uso y con soporte de ayuda online.

Conclusión sobre el análisis de factibilidad

Luego de comprobar la factibilidad en cada etapa analizada, se puede concluir que el proyecto es operacional, técnica, económica y legalmente factible y, por lo tanto, se puede iniciar el desarrollo de éste.

7. Análisis de Riesgo

Las tablas 1 a la 5, describen los riesgos identificados más importantes, divididos según la categoría a la cual correspondan (proyecto, producto, persona, tecnología o herramienta).

Tabla 2: Riesgos de proyecto.

Clave	Riesgo de proyecto	Probabilidad	Efecto
R1	Mala planificación Los plazos determinados para el cumplimiento de los hitos no son realistas o están fuera de alcance.	Media	Muy Grave
R2	Tamaño del sistema mal estimado El sistema es demasiado grande para su elaboración dentro de los plazos determinados.	Media	Muy Grave
R3	Pérdida de material de desarrollo La información utilizada para el desarrollo del sistema se ha dañado o perdido, siendo irrecuperable.	Baja	Grave
R4	Número de desarrolladores insuficiente La capacidad de recursos humanos es insuficiente para cumplir con los plazos determinados para la elaboración del sistema.	Baja	Grave

Tabla 3: Riesgos de producto.

Clave	Riesgo de producto	Probabilidad	Efecto
R5	Poca aceptación del sistema El sistema no es acogido por los usuarios debido a interfaces poco amigables.	Baja	Grave
R6	Cambio de requerimientos El cliente agrega o modifica los requerimientos acordados en un principio.	Media	Grave
R7	Problemas de comunicación con el cliente La comunicación con el cliente disminuye, lo que provoca incomprensión por parte de los desarrolladores.	Baja	Muy Grave
R8	Sistema difícil de utilizar El sistema no cumple con las condiciones aptas para la fácil utilización por parte de usuarios inexpertos o con pocos conocimientos.	Baja	Grave

Tabla 4: Riesgos de persona.

Clave	Riesgo de persona	Probabilidad	Efecto
R9	Desarrolladores incapaces de cumplir con los requerimientos Los desarrolladores no conocen o son incapaces	Baja	Grave

	de dar solución a un problema.		
R10	Mala comunicación y organización entre los desarrolladores Esto impide que se avance rápida y eficientemente en el desarrollo del sistema.	Baja	Media
R11	Poca motivación de los desarrolladores La desmotivación a trabajar, investigar y a realizar otras actividades, puede provocar un quiebre en el grupo	Baja	Muy Grave

Tabla 5: Riesgos de tecnología.

Clave	Riesgo de tecnología	Probabilidad	Efecto
R12	Problemas con recursos de hardware para desarrollo Los recursos utilizados para desarrollar el sistema tienen poca o nula disponibilidad.	Baja	Medio

Tabla 6: Riesgos de herramienta.

Clave	Riesgo de herramienta	Probabilidad	Efecto
R13	Herramientas de difícil manejo Los desarrolladores no pueden trabajar	Media	Grave

	tranquilamente por la dificultad que presentan las herramientas utilizadas.		
--	---	--	--

7.1. Plan de Contingencia y Mitigación

Las tablas 7 y 8 presentan el plan de contingencia y mitigación respectivamente, que definen las maniobras de reacción ante alguno de los riesgos identificados con anterioridad o la forma de reducir el efecto de éstos en el proyecto.

Tabla 7: Plan de Contingencia.

Clave	Plan de contingencia
R1	Reajustar las fechas de plazo y aumentar el tiempo de trabajo para adaptarlas al nuevo plan de desarrollo.
R2	Evaluar la importancia de los módulos y determinar cuáles pueden eliminarse sin dejar de cumplir con los requerimientos del cliente.
R3	Mantener respaldos de toda la información y documentación generada en unidades de memoria transportables, como también en correo electrónico.
R4	Aumentar las horas de trabajo con objeto de suplir las necesidades presentadas por la falta de desarrolladores.
R12	Disponer de equipos alternativos que cuenten con los requerimientos mínimos de trabajo y con las herramientas adecuadas.

Tabla 8: Plan de Mitigación.

Clave	Plan de mitigación
-------	--------------------

R5	A semejar la apariencia del sistema a los que el cliente ya utiliza, con el fin de hacerlo más amigable para los usuarios.
R6	Mantener al máximo la flexibilidad del sistema para así poder adaptarse a eventuales nuevos requerimientos.
R7	Recopilar la información más importante del negocio desde un principio y así mantener una cierta autonomía por parte de los desarrolladores.
R8	Ayudar a los usuarios entregándoles una capacitación clara y consistente del sistema, permitiéndoles conocer a fondo las utilidades que puedan obtener de él.
R9	Realizar procesos investigativos con la finalidad de aprender los métodos y las herramientas a utilizar dependiendo de los problemas que se presenten.
R10	Establecer horarios de trabajo, de manera de mejorar la comunicación del grupo. Además, realizar una asignación de tareas a cumplir para cada desarrollador.
R13	Utilizar la experiencia en las herramientas de trabajo, además de investigar sobre nuevas utilidades que se puedan obtener de estas.

8. Paradigma, Metodología y Herramientas para la realización del Proyecto

Para llevar a cabo un proyecto de software es necesario la utilización de un paradigma de desarrollo que establezca el modo de trabajar y guíe el desarrollo del mismo. Es por ello que es vital la evaluación de diversas alternativas de manera de escoger la que se adapte mejor a la naturaleza del proyecto en cuestión.

8.1. Evaluación de un Paradigma

Con el fin de escoger un paradigma que se adecuara mejor a la naturaleza del proyecto, se considerarán las siguientes alternativas de manera de seleccionar una que se enmarque lo mejor posible al desarrollo del proyecto.

Dentro de los paradigmas [2] a evaluar se consideraron los siguientes:

- Modelo Lineal Secuencial.
- Construcción de Prototipos.
- Modelo Espiral.
- Proceso Unificado.

Modelo Lineal Secuencial

Conocido también como “Ciclo de vida clásico” o “Modelo en Cascada”, este modelo representa un desarrollo secuencial, donde no se puede comenzar a trabajar en la siguiente etapa sin que se haya terminado la actual. En cada actividad se van generando entradas y nueva documentación que es utilizada por la siguiente actividad. Las etapas que comprende esta modelo son: Análisis, Diseño, Codificación, Prueba y Mantenimiento. Lo anterior puede ser observado en la figura 4.

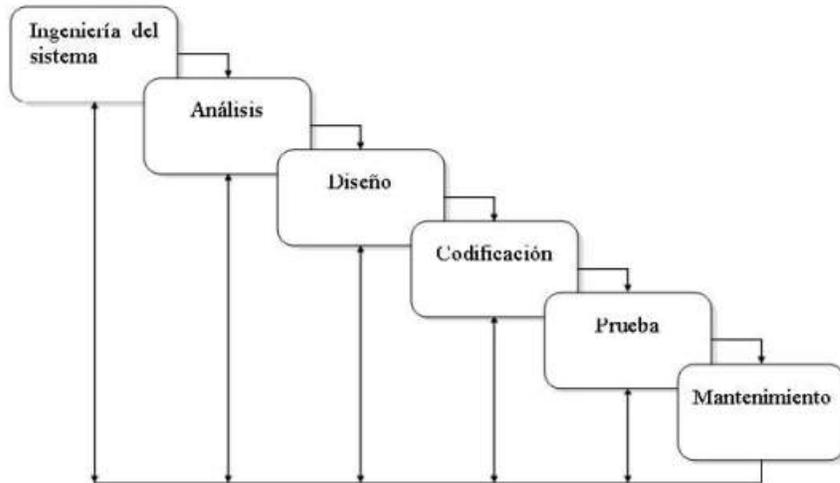


Figura 4: Modelo Lineal Secuencial.

Ventajas

- Es el paradigma más antiguo y ampliamente utilizado en la ingeniería del software, existiendo bastante información asociada a éste.
- Si el cliente está claro en lo que quiere y los requerimientos son captados de muy buena manera, el resultado obtenido puede ser muy bueno.
- Ya que es un modelo secuencial donde una etapa conduce a la otra, resulta sencillo planificar el trabajo en torno a cada una de las fases de desarrollo.

Desventajas

- Los proyectos muy pocas veces siguen la secuencialidad propuesta por el modelo.
- Resulta difícil que el cliente exponga todos los requisitos en un principio.
- El cliente debe esperar hasta el final para ver los resultados del proyecto.
- El no detectar los errores en las primeras etapas (análisis y diseño), puede resultar muy costoso.
- Dado que el cliente sólo ve los resultados al final, el mantenimiento está dado por hacer parches debido a cambios por parte del cliente o del entorno del mismo.

Construcción de Prototipos

Este paradigma comienza con el establecimiento de los requerimientos del sistema, se definen los objetivos y requisitos en relación a las áreas de mayor prioridad e importancia para el sistema.

Luego el objetivo se centra en el desarrollo de un prototipo inicial, el cual es mostrado al cliente con el fin de aclarar las dudas con respecto a lo que realmente se quiere.

Este paradigma ayuda mucho cuando no se tiene una total claridad de los requerimientos del sistema por parte del cliente. En la figura 5 se muestra este paradigma, el cual parte en la recolección y refinación de requerimientos, avanzando en el sentido de las agujas del reloj, hasta llegar al mismo punto.

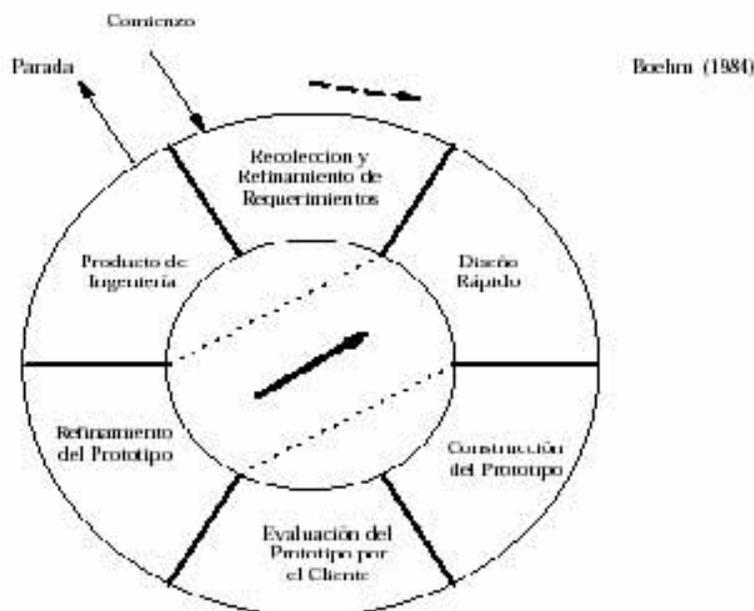


Figura 5: Construcción de Prototipos.

Ventajas

- Puede servir como un medio de captación de requerimientos, como también para descubrir la existencia de errores.
- Permite asegurar que el producto de software se acerca a lo que el cliente quiere.

- El cliente no tiene que esperar tanto tiempo para ver resultados.
- Si el prototipo está construido en base a un buen conjunto de requerimientos, sólidos y que satisface lo que el cliente desea, puede ser utilizado como un prototipo de trabajo sobre el cual se va a desarrollar la versión final del software.

Desventajas

- El cliente puede pensar que el prototipo representa una versión funcional del software que desea.
- El desarrollador, a menudo, hace compromisos de implementación para hacer que el prototipo funcione rápidamente, por lo que puede utilizar un sistema operativo o lenguajes de programación que son conocidos y que están disponibles, pero que no son los más adecuados.
- El cliente no logra entender que el sistema tiene que ser construido nuevamente para mantener altos niveles de calidad.

Modelo Espiral

Es un modelo Evolutivo que mezcla la construcción de prototipos con los aspectos controlados y sistemáticos del modelo lineal secuencial.

Este modelo se basa en la construcción de software en versiones incrementales, donde las primeras iteraciones se podría tener un modelo en papel o prototipo, y en las últimas iteraciones se tienen versiones del software cada vez más completas. En la figura 6 se puede apreciar las etapas de este modelo.

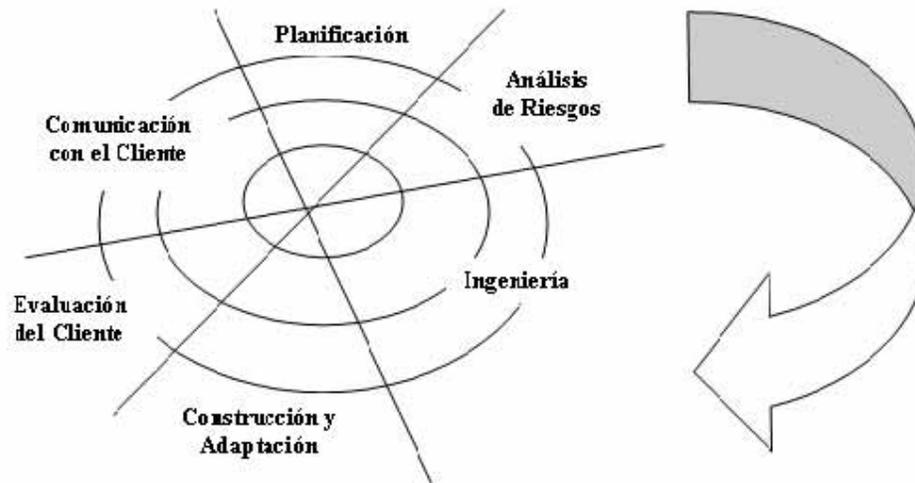


Figura 6: Modelo Espiral.

Este modelo se divide en un número de actividades de marco de trabajo. Por lo general, existen entre tres y seis actividades. Estas son las siguientes:

- Comunicación con el cliente: corresponde a las tareas requeridas para establecer comunicación entre el desarrollador y el cliente.
- Planificación: se refiere a las tareas necesarias para definir los recursos, el tiempo y otro tipo de información relacionada con el proyecto.
- Análisis de riesgo: corresponde a las tareas requeridas para evaluar riesgos técnicos y de gestión.
- Ingeniería: se refiere a las tareas requeridas para construir una o más representaciones de la aplicación.
- Construcción y acción: corresponde a las tareas requeridas para construir, probar, instalar y proporcionar soporte al usuario (por ejemplo documentación y práctica).
- Evaluación del cliente: se refiere a las tareas que se necesitan para obtener la reacción del cliente según la evaluación de las representaciones del software creadas durante la etapa de ingeniería e implementada durante la etapa de instalación.

Ventajas

- Cuando el proyecto es pequeño, el número de tareas junto con la documentación asociada tienen poca formalidad.
- Permite al desarrollador y cliente comprender y reaccionar mejor ante riesgos en cada uno de los niveles evolutivos, utilizando la construcción de prototipos como mecanismo de reducción de riesgos.

Desventajas

- Puede resultar complicado convencer a grandes clientes de que este enfoque es el más adecuado.
- Requiere de una gran habilidad para la evaluación de los riesgos, lo cual puede incidir a futuro en problemas en caso de no llevar a cabo bien este paso.

Proceso Unificado

El Proceso Unificado es un proceso de desarrollo de software que se configura y adapta a proyectos de distinto tamaño y complejidad.

Este paradigma permite guiar a los equipos de desarrollo en el manejo de las iteraciones incrementales y las duraciones de estas, que conforme pasa el tiempo van dando forma al sistema en su totalidad, manteniendo un balance entre los requerimientos del negocio, el tiempo de lanzamiento al mercado y los riesgos del proyecto. Además, es soportado por herramientas que automatizan entre otras cosas, el modelamiento visual, la administración de cambios y las pruebas.

Características

- Generalmente iterativo, para proyectos pequeños puede usarse un modelo lineal.
- Desarrollo centrado en la arquitectura, facilitándose el desarrollo en paralelo, la reutilización y el mantenimiento.
- Está dirigido por los casos de uso.

- Configurable y adaptable a diferentes proyectos.
- Pone énfasis en el control de la calidad y la gestión de los riesgos.

Fases del Proceso Unificado

Una fase en un paradigma de desarrollo consiste en un intervalo de tiempo entre dos hitos importantes del proceso durante el cual se cumple un conjunto de objetivos bien definidos, se completan artefactos y se toman decisiones sobre si pasar a la siguiente fase. El Proceso Unificado define cuatro fases, como se observa en la figura 7.

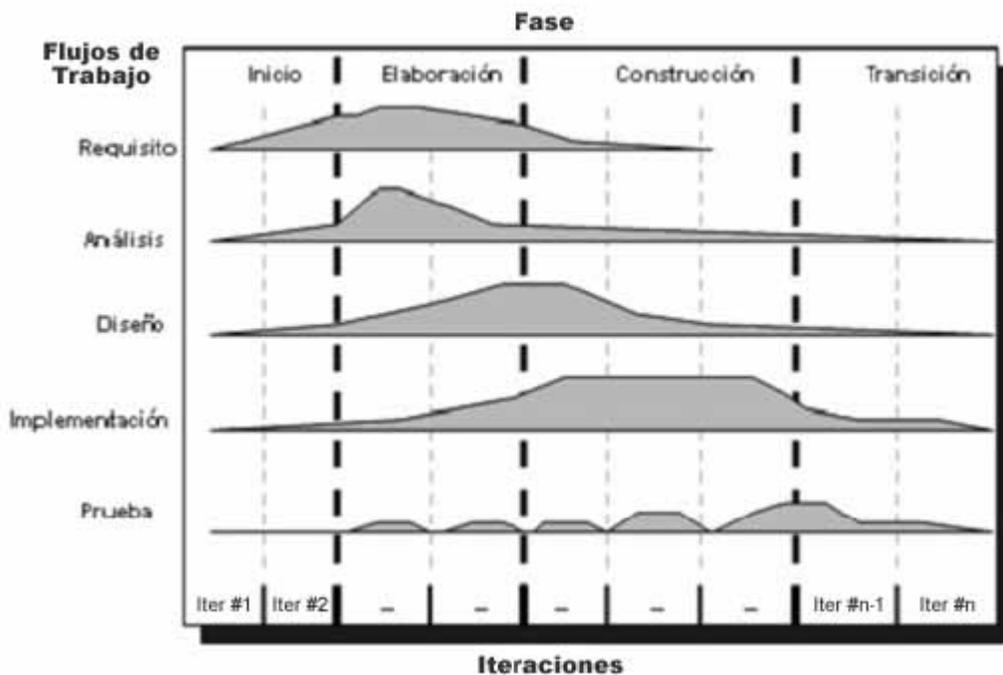


Figura 7: Fases del Proceso Unificado.

Inicio

- Se establece la planificación del proyecto y los costos asociados.
- Identificación de los riesgos.
- A partir de una idea inicial se establece una visión aproximada del producto final y un análisis del negocio.

Elaboración

- Establecimiento de un plan y arquitectura correcta.
- Mitigación de cuestiones de alto riesgo.
- Especificación en detalle de los casos de uso importantes.
- Se diseña la arquitectura conceptual, lógica y física del sistema.

Construcción

- Desarrollo del sistema.
- La arquitectura especificada en la Elaboración es complementada añadiendo el producto de software.

Transición

- En esta fase el sistema es sometido a pruebas, con el fin de encontrar posibles errores.
- El sistema es puesto en producción.

8.2. Selección del Paradigma

El paradigma que se utilizará en el desarrollo del proyecto será el Proceso Unificado de Desarrollo, ya que dada la naturaleza del proyecto, el cual contempla el desarrollo de dos sistemas, permite manejar de una mejor forma las iteraciones, realizar actividades en paralelo, punto muy importante para avanzar de igual manera en ambos sistemas, especialmente en lo que se refiere al análisis y diseño. Además ayuda a establecer responsabilidades en las tareas asignadas individualmente y verlas a su vez como un todo.

Otro punto tomado en cuenta, es que permite manejar de una mejor manera los riesgos que puedan afectar el desarrollo del software, dado su naturaleza iterativa e incremental, y dirigido por los casos de uso.

8.3. Metodología a utilizar

De acuerdo al paradigma seleccionado, la metodología a utilizarse en el desarrollo del proyecto será la Orientación a Objetos [3]. A continuación se dará una explicación de esta metodología y las ventajas que posee.

Orientación a Objetos

Antes de describir esta metodología, es necesario tomar en cuenta dos conceptos que resultan fundamentales para su comprensión: objeto y clase.

Un objeto es la abstracción de algo dentro del contexto del problema, que refleja las capacidades del sistema para mantener información sobre él, interactuar con otros objetos, encapsular los valores de los atributos y de sus operaciones exclusivas. Un objeto corresponde a una “instancia” de una clase.

Una clase describe un conjunto de objetos que poseen atributos y operaciones comunes.

Esta metodología está enfocada en la utilización de objetos para la representación de “cualquier cosa” real o abstracta, permitiendo almacenar en ellos los datos y operaciones que fueron definidos en cada clase de la cual fue instanciado. Los objetos se crean con la finalidad de prestar servicios, los cuales responden a solicitudes que les son hechas, generando operaciones que ejecutan métodos que retornan una respuesta.

Las clases pueden estar compuestas de otras clases, llamadas subclases. Estas heredan todas las características de su clase padre, y a su vez pueden definir atributos y/o operaciones propios que la identifiquen. La Orientación a Objetos (OO) asegura lo anterior, permitiendo que cada subclase pueda heredar atributos y operaciones de sus predecesores.

Para llevar a cabo el Desarrollo Orientado a Objetos se construyen dos tipos de modelos:

- Análisis de la Estructura de Objetos: conduce a la representación de los tipos de objetos que le son propios al sistema sometido a tratamiento informático y las relaciones que mantienen mediante diagramas de relación entre objetos.

- Análisis de Comportamiento de Objetos: permite dar cuenta de lo que le ocurre a los objetos con el pasar del tiempo, chequeando los distintos estados por los que este pueda haber pasado.

En comparación con el Desarrollo Estructurado, la OO difiere de éste porque:

- Primero construye un modelo de objetos en vez de un modelo funcional.
- Integra objetos, atributos y operaciones, en vez de separarlos entre el modelo de datos y el modelo funcional.

Ventajas

- Fomenta la reutilización y extensión del código.
- Permite crear sistemas más complejos.
- Relacionar el sistema al mundo real.
- Facilita la creación de programas visuales.
- Construcción de prototipos.
- Agiliza el desarrollo de software.
- Facilita el trabajo en equipo.
- Facilita el mantenimiento del software.

8.4. Herramientas a utilizar en el desarrollo del proyecto

Herramienta de Modelado: UML (Unified Modeling Language)

La herramienta para modelar a utilizar en el análisis y diseño del proyecto va a ser UML [4], ya que ésta permite el modelado orientado a objetos, logrando así poder especificar, visualizar, construir y documentar artefactos del futuro sistema de software. Además,

permite entender, diseñar, configurar, mantener y controlar la información sobre los sistemas que se vayan a construir.

UML capta la información sobre la estructura estática y el comportamiento dinámico de un sistema. Un sistema se modela como una colección de objetos discretos que interactúan para realizar un trabajo que finalmente beneficia a un usuario externo.

UML no es un lenguaje de programación, sino más bien un lenguaje de modelado visual que permite una abstracción del sistema y sus componentes.

UML permite expresar un modelo utilizando una notación de modelado con reglas sintácticas, semánticas y prácticas.

En UML un sistema puede ser representado desde cinco vistas diferentes, en que cada vista lo define de una perspectiva diferente:

- Vistas de Usuario: representa el sistema desde las perspectivas de los usuarios. El caso de uso es el enfoque elegido para modelar la vista.
- Vista Estructural: los datos y la funcionalidad se muestran desde dentro del sistema, es decir, modela la estructura estática (clases, objetos y relaciones).
- Vista del Comportamiento: esta parte del modelo del análisis representa los aspectos dinámicos o de comportamiento del sistema. También muestra las interacciones o colaboraciones entre los diversos elementos estructurales descritos en las vistas anteriores.
- Vista de Implementación: los aspectos estructurales y de comportamiento se representan aquí tal y como van a ser implementados.
- Vista del Entorno: aspectos estructurales y de comportamiento en el que el sistema a implementar se representa.

En general, el modelo de análisis de UML se centra en las vistas del usuario y estructural. Estas proporcionan una visión interna del uso de las situaciones para el sistema (facilitan guías para el modelado del comportamiento), y establecen fundamentos para la

implementación y vistas del modelo ambiental, identificando y describiendo elementos estructurales estáticos del sistema. El diseño de UML se dirige más a las vistas del comportamiento y del entorno. El diseño de sistema se centra en la arquitectura de software y definición de subsistemas. El diseño describe objetos, hasta un nivel en el cual puedan ser implementados en un lenguaje de programación.

Lenguaje de Programación: PHP

PHP, acrónimo de "PHP: Hypertext Preprocessor", es un lenguaje "Open Source" interpretado de alto nivel, especialmente pensado para desarrollos Web y el cual puede ser embebido en páginas HTML. La mayoría de su sintaxis es similar a C, Java y Perl y es fácil de aprender. La meta de este lenguaje es permitir escribir a los creadores de páginas Web, páginas dinámicas de una manera rápida y fácil, aunque se pueda hacer mucho más con PHP, su funcionamiento puede ser apreciado en la figura 8.



Figura 8: Funcionamiento de PHP.

Pero a diferencia de Java o JavaScript que se ejecutan en el navegador, PHP se ejecuta en el servidor, por eso permite acceder a los recursos que tenga el servidor como por ejemplo podría ser una base de datos. El programa PHP es ejecutado en el servidor y el resultado enviado al navegador. El resultado es normalmente una página HTML pero igualmente podría ser una página WML.

Al ser PHP un lenguaje que se ejecuta en el servidor no es necesario que su navegador lo soporte, es independiente del navegador, pero sin embargo para que sus páginas PHP funcionen, el servidor donde están alojadas debe soportar PHP.

PHP puede hacer cualquier cosa que se pueda hacer con un script CGI, como procesar la información de formularios, generar páginas con contenidos dinámicos, o mandar y recibir cookies. Y esto no es todo, se puede hacer mucho más.

PHP puede ser utilizado en cualquiera de los principales sistemas operativos del mercado, incluyendo Linux, muchas variantes Unix (incluido HP-UX, Solaris y OpenBSD), Microsoft Windows, Mac OS X, RISC OS y probablemente alguno más. PHP soporta la mayoría de servidores Web de hoy en día, incluyendo Apache, Microsoft Internet Information Server, Personal Web Server, Netscape y iPlanet, O'Reilly Website Pro server, Caudium, Xitami, OmniHTTPd y muchos otros. PHP tiene módulos disponibles para la mayoría de los servidores, para aquellos otros que soporten el estándar CGI, PHP puede usarse como procesador CGI.

Base de Datos: PostgreSQL

PostgreSQL es la base de datos a código abierto más avanzada del mundo, superando incluso a bases de datos comerciales. PostgreSQL permite a las organizaciones obtener el máximo valor de su información garantizando a las personas el acceso a la información correcta en el momento y lugar adecuados.

PostgreSQL es un sistema de gran alcance para los datos gerenciales sobre los usos misión-críticos, transacción-intensivos de la empresa. Las compañías, en industrias altamente competitivas y que cambian rápidamente, eligen PostgreSQL por su flexibilidad, rendimiento, y por el uso eficiente de los recursos.

Las características realzadas de la seguridad incluyendo el control de acceso a los clientes y las comunicaciones cifradas protegen datos valiosos mientras que aseguran el acceso permanente a los empleados y a los clientes.

9. Desarrollo

9.1. Descripción general del proyecto

El proyecto en curso comprende el desarrollo de 2 sistemas independientes entre sí, los cuales serán implementados en un ambiente Web, con el objeto de que los usuarios que se encuentran en los distintos recintos hospitalarios de la red asistencial del SSVQ puedan acceder a estas aplicaciones en forma remota, logrando así una mayor conectividad entre estos recintos y el SSVQ.

9.1.1. Características de los usuarios

Tomando en cuenta a ambos sistemas y tratándose de aplicaciones Web, se pueden identificar 2 tipos de usuarios:

- Usuario común: este tipo de usuario interactuará con el Sistema OIRS, y lo podrá hacer desde cualquier computador con acceso a Internet, pudiendo acceder a las funciones disponibles para este tipo de usuario en el sistema. Dado que el nivel computacional de estos usuarios no es el mismo, se utilizarán estándares planteados por el SSVQ y que ya han sido implementados en otras aplicaciones creadas para este organismo.
- Usuario interno o funcionario del hospital: este tipo de usuario podrá acceder a la gran parte de las funcionalidades que tendrán ambos sistemas, las cuales dependerán del área de desempeño del funcionario. En el caso del Sistema OIRS están los siguientes tipos de usuarios:
 - Auxiliar del Servicio.
 - Jefe del Servicio.
 - Director Hospital o SSVQ.

- Jefe de la OIRS (que será el administrador del sistema).

Por otro lado, se tienen a los usuarios del Sistema Solicitudes de Atención:

- Empleado del Servicio.
- Trabajador, el cual será el que realice cada uno de los trabajos que implique las solicitudes hechas por el empleado del servicio.
- Director Hospital o SSVQ.
- Jefe Departamento (que será el administrador del sistema).

Estos usuarios están más familiarizados con el uso de aplicaciones a nivel interno, y como se mencionó anteriormente, se mantendrán estándares de aplicaciones con las que ya cuenta el SSVQ.

9.1.2. Requerimientos funcionales

A continuación se presentan los requerimientos de cada sistema:

Sistema OIRS

- Permitir el ingreso directo de las RFS al sistema por parte de los usuarios.
- Controlar el vencimiento de los plazos establecidos para responder a cada RFS.
- Permitir a los usuarios ver el estado de la RFS que ingresó al sistema.
- Generación de informes estadísticos con la cantidad de RFS por cada unidad del hospital, los cumplimientos de éstas, RFS frecuentes, entre otros informes.

Sistema de Solicitudes de Atención

- Permitir al solicitante ingresar su solicitud al sistema, como también ver el estado en que se encuentre ésta.
- Facilitar al jefe del departamento la labor de asignación de personal para una tarea.

- Registrar todo los materiales utilizados en una tarea realizada, sus costos asociados, como también las observaciones que pueda considerar el personal que realizó dicho trabajo.
- Permitir al departamento de informática seguir el historial de un equipo computacional, con el objeto de determinar las fallas que ha tenido, el tipo, y frente a esto poder establecer si es necesario una nueva reparación o la compra del insumo o equipo computacional.
- Generar informes que permitan determinar los costos en que se ha estado incurriendo, fallas frecuentes, personal asignado y horas trabajadas, entre otros.

9.1.3. Requerimientos no funcionales

Los requerimientos no funcionales se centran en las características que de una u otra forma pudieran limitar la funcionalidad del sistema. En relación a lo anterior se han tomado en cuenta los siguientes puntos:

- Usabilidad: debe permitir al usuario manejarse fácilmente dentro del sistema, ya que se ignora el conocimiento computacional que puedan tener los usuarios externos al SSVQ.
- Confiabilidad: el sistema debe proporcionar al usuario la confianza de que la información que está siendo ingresada en éste es almacenada correctamente para que posteriormente pueda ser usada en la forma que se espera.
- Seguridad: el sistema debe permitir el ingreso controlado por medio de nombres de usuario y contraseñas. Además, los usuarios que sólo tengan permisos de administrador podrán tener acceso a funciones de mantenimiento y control del sistema.
- Limitaciones del diseño: el diseño de las interfaces del sistema debe ajustarse a los parámetros establecidos en anteriores aplicaciones elaborados para el SSVQ, manteniendo la distribución de los componentes, los colores y todo lo necesario para hacer más fácil la adaptación a éste.

9.1.4. Métricas de evaluación

Para la evaluación del cumplimiento de estos requerimientos se han establecido métricas que permitirán determinar numéricamente cuán cumplido está cada requerimiento.

Para medir cuán usables son los sistemas se realizarán cuestionarios tipo “test”, en los cuales los usuarios de los sistemas deberán responder preguntas acerca del sistema, escogiendo alguna alternativa dentro de un rango específico. El cumplimiento de las limitaciones de diseño se medirá a través de un cuestionario que sea capaz de evaluar los puntos considerados por QUIS (Questionnaire for User Interface Satisfaction) [5], un cuestionario estándar que analiza el cumplimiento de los requerimientos de las interfaces del sistema. El otro cuestionario a realizar también es estándar, denominado SUMI (Software Usability Measurement Inventory) [6], el cual tiene como objetivo valorar la calidad de uso del sistema, a través de un inventario de medidas de usabilidad.

La confiabilidad del sistema se medirá por medio de pruebas que se realizarán al finalizar el sistema donde se analizará en detalle el correcto funcionamiento del sistema y, a partir de esto, poder cuantificar la confiabilidad de cada módulo y del sistema en general.

Finalmente se estudiará el punto más importante y complejo de los requerimientos no funcionales, la seguridad. Para ello se realizarán mediciones a través de métricas que evalúen los riesgos del sistema, sus vulnerabilidades, la respuesta a incidentes, el control de acceso, etc.

9.2. Actividades realizadas por iteración

Cada uno de los módulos del sistema será desarrollado dentro de una iteración, la cual estará compuesta de las siguientes etapas:

a) Especificación de los casos de uso más relevantes

En esta etapa se hará la especificación de los casos de uso contemplados en el módulo, con el objeto que puedan ser entendidos en su totalidad.

b) Análisis

De acuerdo a las especificaciones de los casos de uso realizadas, se realizará un análisis de éstas con el objeto de poder visualizar las entidades que participan en el módulo. Una vez identificadas, se podrá generar el Modelo del Dominio correspondiente, que identifique las entidades asociadas al negocio. Después de ser identificadas las entidades, se definirán las distintas operaciones que debe tener cada una, con el propósito de generar los Contratos de operaciones del módulo.

Los Contratos de operaciones son creados con la finalidad de que puedan reflejar los efectos que tendrán las operaciones sobre los objetos del modelo del dominio, describiendo los diversos cambios causados por la operación. Para identificar estos contratos se deberá analizar cada especificación de caso de uso junto con el diagrama de secuencia asociado a éste.

Para realizar la especificación de los contratos se utilizará la siguiente tabla:

Tabla 9: Plantilla especificación de contratos.

Operación	Nombre de la operación y sus parámetros.
Precondiciones	Suposiciones relevantes sobre el estado del sistema o de los objetos del Modelo del Dominio antes de la ejecución de la operación. No se comprobará en la lógica de esta operación, se asume que son verdad y son suposiciones no triviales.
Descripción	Se describe en lenguaje natural el impacto que implica la realización del proceso dentro del Modelo del Dominio, además se establece el cómo se realizará.
Poscondiciones	Describen los cambios en el estado de los objetos del Modelo del Dominio. Estos cambios comprenden:

	<ul style="list-style-type: none">- La creación y eliminación de instancias.- Formación o rotura de asociaciones.- Modificaciones en los atributos.
--	---

c) Diseño

Una vez terminada la etapa de análisis, los diagramas obtenidos, modelo del dominio y los contratos de operaciones, permitirán realizar el Modelo del Diseño del módulo. Este modelo corresponde a un Diagrama de clases, el que a diferencia del modelo del dominio, posee un nivel de especificación mayor, el cual incluye los atributos y las operaciones que tendrá cada objeto.

9.2.1. Integración de los módulos

Una vez finalizadas cada una de las iteraciones, los modelos de análisis y diseño obtenidos serán integrados con el objeto de generar el Modelo de Análisis y Diseño integrado.

De la misma manera, los modelos de datos resultantes en cada iteración serán integrados, con el objeto de generar el Modelo de la base de datos que tendrá el sistema.

10. Análisis y Diseño del Proyecto

La utilización de artefactos UML para la especificación y representación de un proyecto resulta fundamental para el desarrollo de este y poder comprender el alcance que tendrá, así como lograr la identificación de cada uno de los actores del sistema en cuestión y las futuras funcionalidades, de modo que puedan ser comprendidas por cualquier persona, independiente de sus conocimientos, al estar en lenguaje de modelado visual de fácil comprensión.

A continuación se detallarán cada una de las iteraciones planteadas en el punto 6.4, donde se partirá en primer lugar por el Sistema OIRS, para continuar con el de Solicitudes de Atención.

10.1. Sistema OIRS

Diagrama general de Casos de uso

Figura 9: Casos de uso Paciente.

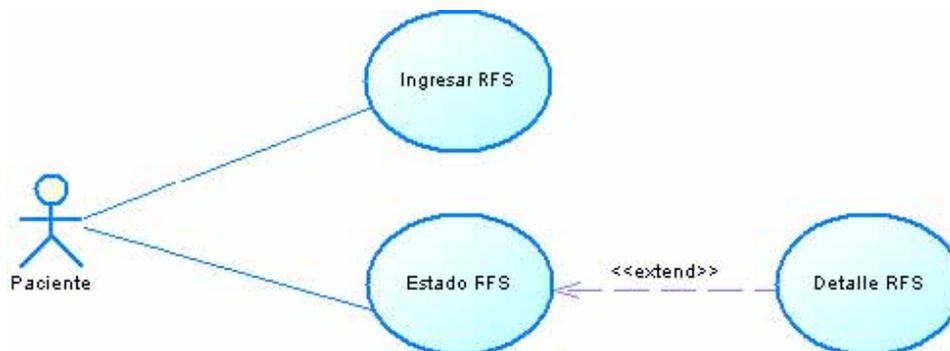


Figura 10: Casos de uso Director.

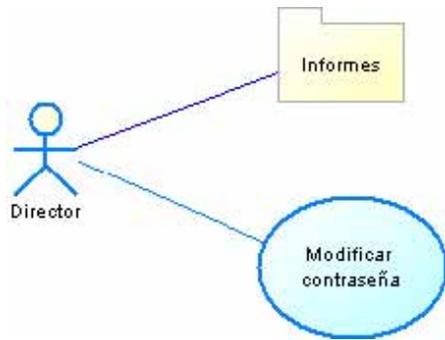


Figura 11: Casos de Uso Jefe Servicio.

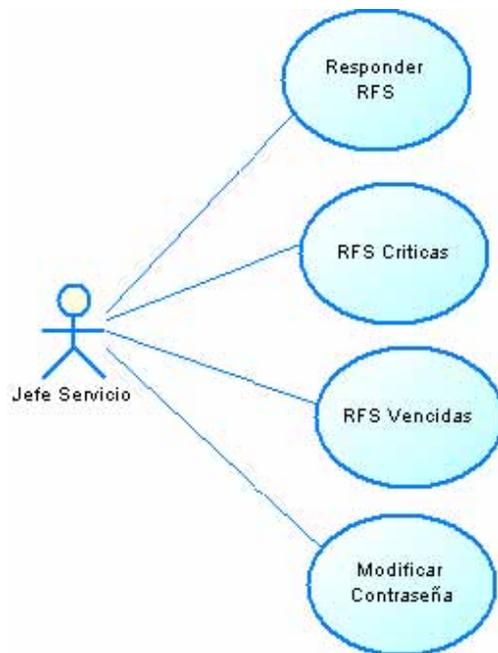
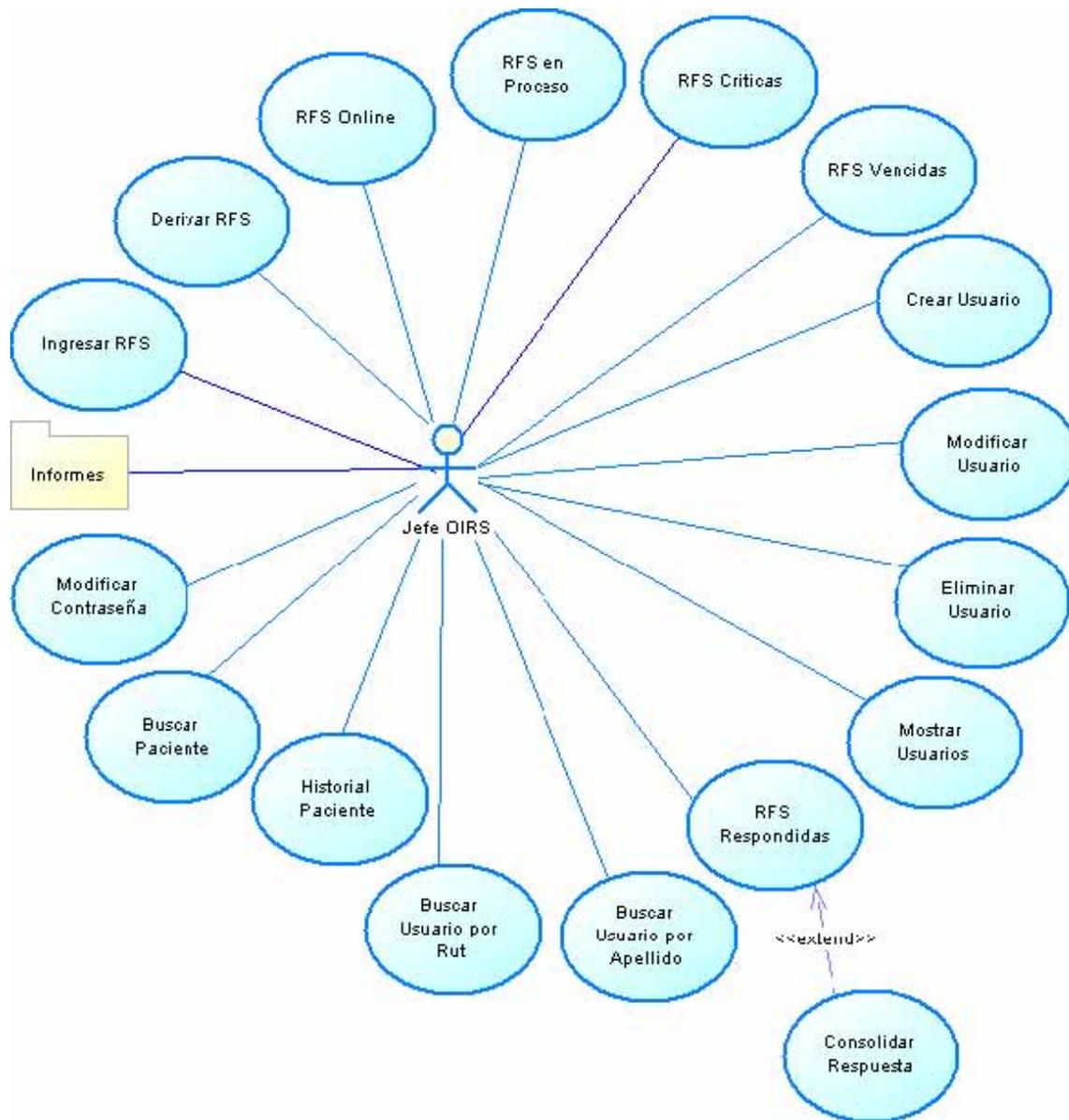


Figura 12: Casos de uso Jefe OIRS.



10.1.1. Iteración 1: Módulo RFS

Especificación Caso de Uso Login

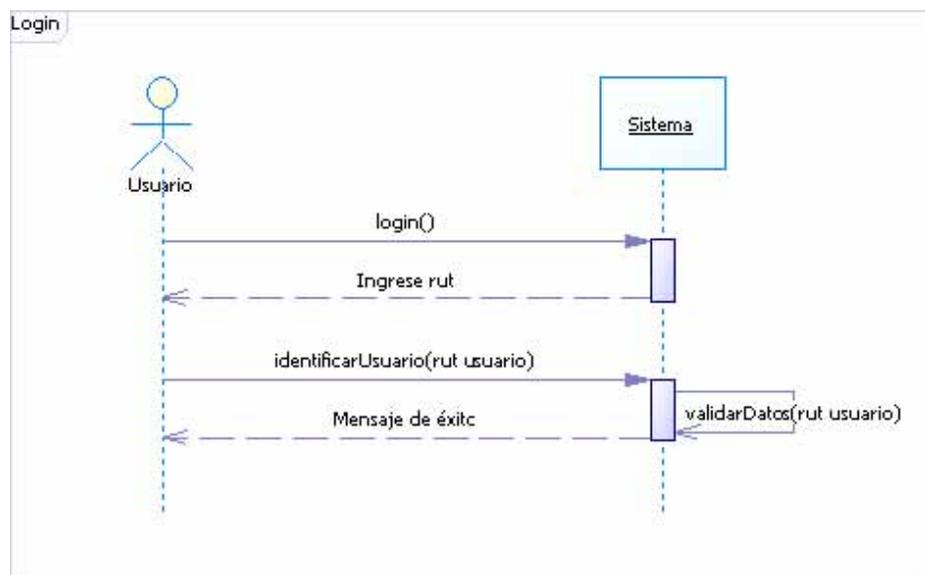
Tabla 10: Especificación Caso de Uso Login.

Nombre	Login

Actores	Usuario (Director, Jefe OIRS, Jefe Servicio)	
Descripción	Este caso de uso permite la identificación de un usuario en el sistema.	
Precondición	El Usuario existe en el sistema.	
Escenario Principal	Paso	Acción
	1	El Usuario ingresa al sistema.
	2	El sistema solicita el ingreso del Rut y contraseña del usuario.
	3	El Usuario ingresa los datos solicitados.
	4	El sistema lo identifica como un usuario valido y le entrega los privilegios del perfil correspondiente.
Poscondición	No hay.	
Excepciones	Paso	Acción
	3a	Si el Rut ingresado es inválido, el sistema mostrará un mensaje solicitando que escriba uno que sí lo sea.
	3b	Si la contraseña es inválida, el sistema mostrará un mensaje solicitando que la reingrese.
	4a	Si el usuario no existe, el sistema mostrará un mensaje indicándolo.
Frecuencia	Alta.	

Diagrama de Secuencia

Figura 13: Diagrama de Secuencia Login.



Especificación Caso de Uso Ingresar RFS

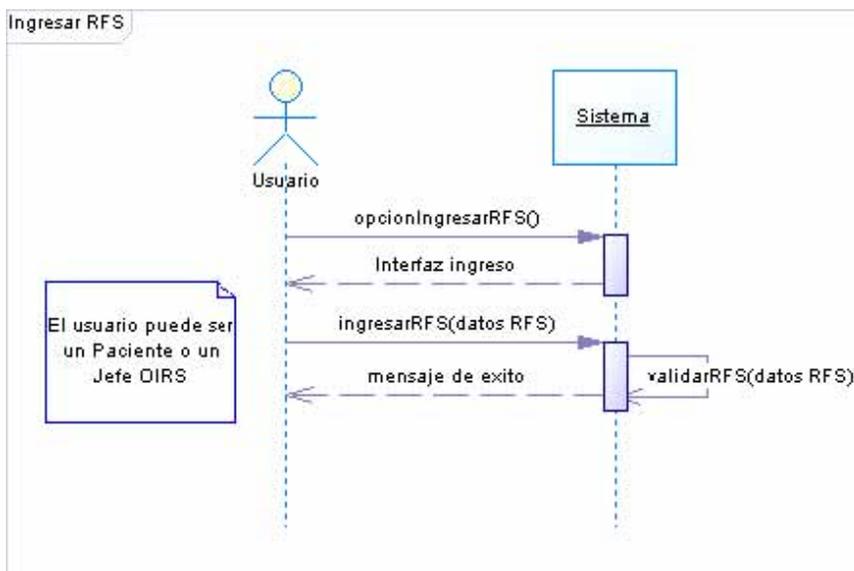
Tabla 11: Especificación Caso de Uso Ingresar RFS.

Nombre	Ingresar RFS	
Actores	Paciente, Jefe OIRS.	
Descripción	En este caso de uso se describe la acción de poder ingresar un reclamo, una felicitación o sugerencia.	
Precondición		
Escenario	Paso	Acción
Principal	1	El usuario selecciona la opción Ingresar RFS.
	2	El sistema muestra la interfaz para el ingreso de la información.

	3	El usuario ingresa la información de la nueva RFS.
	4	El sistema valida la información ingresada por el usuario.
Poscondición	El sistema valida la información guardándola en el sistema.	
Excepciones	Paso	Acción
	3a	El usuario presiona volver, cancelando la operación.
	4a	Si los datos no son correctos, el sistema muestra un mensaje de error, volviendo al paso 2.
Frecuencia	Alta	

Diagrama de secuencia

Figura 14: Diagrama de secuencia Ingresar RFS.



Especificación Caso de Uso Responder RFS

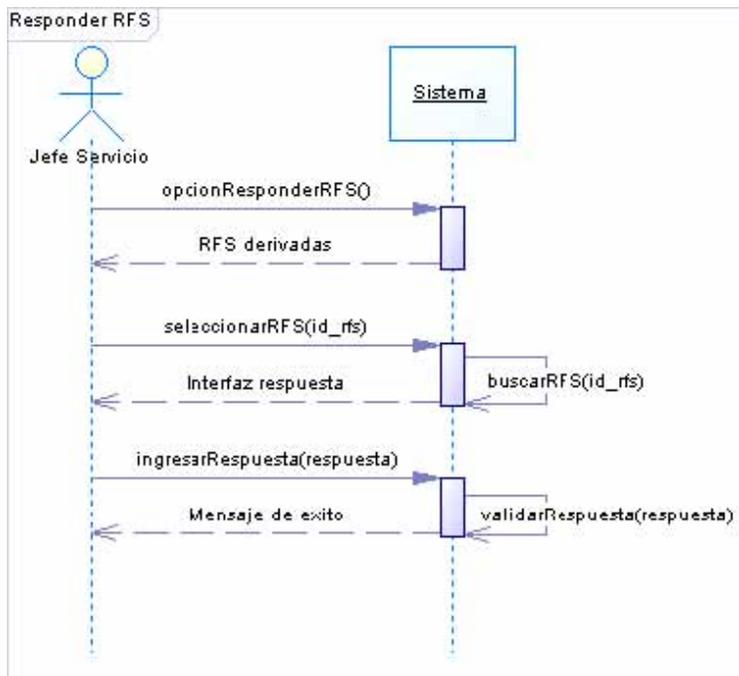
Tabla 12: Especificación Caso de Uso Responder RFS.

Nombre	Responder RFS	
Actores	Jefe del servicio.	
Descripción	En este caso de uso se describe la acción en que el jefe del servicio clínico tiene que responder la RFS derivada a su servicio.	
Precondición	El jefe del servicio tiene que haberse identificado en el sistema.	
Escenario	Paso	Acción
Principal	1	El jefe del servicio selecciona la opción Responder RFS.
	2	El sistema muestra todas las RFS que han sido derivadas al servicio.
	3	El jefe del servicio selecciona una RFS.
	4	El sistema muestra interfaz con el detalle de la RFS, donde el jefe del servicio procede a responderla.
	5	El sistema valida el proceso, mostrando un mensaje de confirmación, finalizando el caso de uso.
Poscondición	El sistema almacena la respuesta, quedando disponible para que el jefe de la OIRS pueda consolidarla.	

Excepciones	Paso	Acción
	2 ^a	No hay RFS derivadas al servicio.
	3 ^a	El jefe del servicio presiona volver, cancelando la operación.
	3b	Si el jefe del servicio selecciona una RFS reciente, habiendo una más antigua, el sistema informa de ello, con el objeto de darle prioridad a esta última.
	4 ^a	El jefe del servicio presiona volver, volviendo al paso anterior.
	5 ^a	El jefe de servicio puede volver al paso 3 para dar respuesta a otras RFS.
Frecuencia	Alta	

Diagrama de secuencia

Figura 15: Diagrama de secuencia Responder RFS.



Especificación Caso de Uso Estado RFS

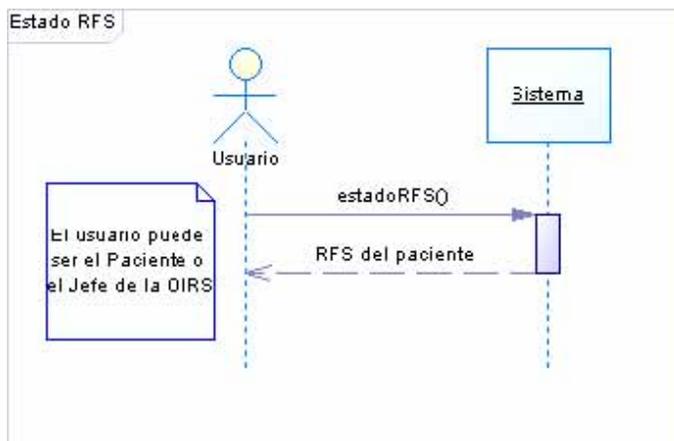
Tabla 13: Especificación Caso de Uso Estado RFS.

Nombre	Estado RFS	
Actores	Paciente, Jefe OIRS.	
Descripción	En este caso de uso se describe la acción de ver si una RFS ha sido tomada en cuenta o se le ha dado una respuesta.	
Precondición	El usuario tiene que haberse identificado en el sistema.	
Escenario Principal	Paso	Acción
	1	El usuario selecciona la opción Estado RFS.
	2	El sistema muestra el estado de todas las RFS que tenga el paciente.

Poscondición		
Excepciones	Paso	Acción
	2ª	El paciente no tiene RFS ingresadas.
Frecuencia	Media	

Diagrama de secuencia

Figura 16: Diagrama de secuencia Estado RFS.



Especificación Caso de Uso RFS Vencidas

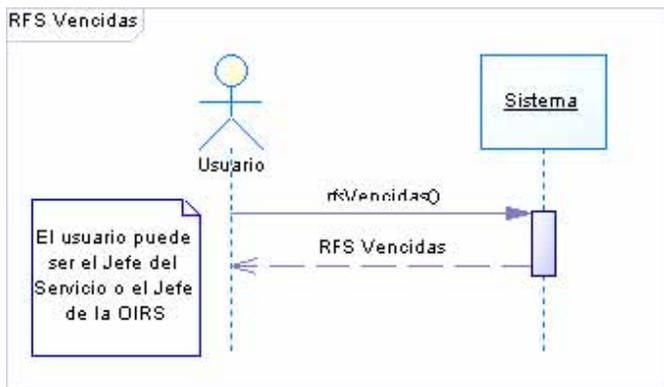
Tabla 14: Especificación Caso de Uso RFS Vencidas.

Nombre	RFS Vencidas
Actores	Jefe Servicio, Jefe OIRS.
Descripción	En este caso de uso se describe la acción de ver aquellas RFS que se encuentren fuera del periodo de tiempo establecido para darles una respuesta.

Precondición	El usuario tiene que haberse identificado en el sistema.	
Escenario Principal	Paso	Acción
	1	El usuario selecciona la opción RFS Vencidas.
	2	El sistema muestra todas aquellas RFS que se encuentran en esta condición.
Poscondición		
Excepciones	Paso	Acción
	2ª	No hay RFS fuera del periodo establecido.
Frecuencia	Baja	

Diagrama de secuencia

Figura 17: Diagrama de secuencia RFS Vencidas.



Especificación Caso de Uso RFS Críticas

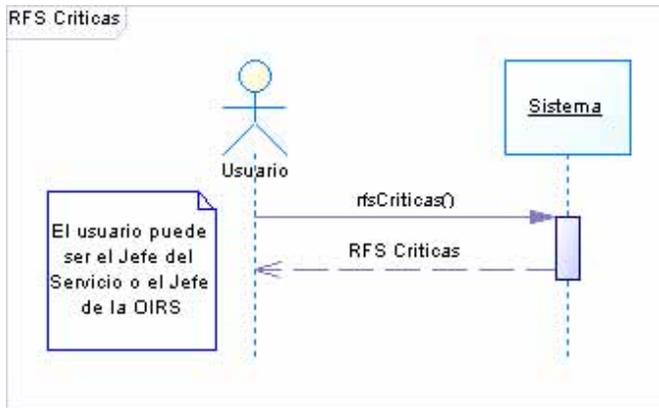
Tabla 15: Especificación Caso de Uso RFS Críticas.

--	--

Nombre	RFS Criticas	
Actores	Jefe Servicio, Jefe OIRS.	
Descripción	En este caso de uso se describe la acción de ver aquellas RFS que se encuentren próximas a terminar el periodo de tiempo en el cual se les tiene que dar una respuesta.	
Precondición	El usuario tiene que haberse identificado en el sistema.	
Escenario Principal	Paso	Acción
	1	El usuario selecciona la opción RFS Críticas.
	2	El sistema muestra todas aquellas RFS que se encuentran en esta condición.
Poscondición		
Excepciones	Paso	Acción
	2ª	No hay RFS en esta situación.
Frecuencia	Baja	

Diagrama de secuencia

Figura 18: Diagrama de secuencia RFS Criticas.



Especificación Caso de Uso Derivar RFS

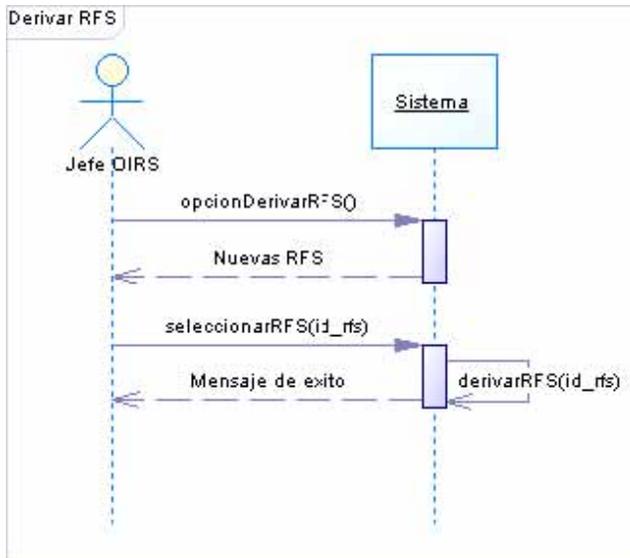
Tabla 16: Especificación Caso de Uso Derivar RFS.

Nombre	Derivar RFS	
Actores	Jefe OIRS.	
Descripción	En este caso de uso se describe la acción de derivar las RFS a los diferentes servicios clínicos con el objeto que se le de una respuesta.	
Precondición	El jefe de la OIRS tiene que haberse identificado en el sistema.	
Escenario Principal	Paso	Acción
	1	El jefe de la OIRS selecciona la opción Derivar RFS.
	2	El sistema muestra las nuevas RFS que han sido ingresadas.
	3	El jefe de la OIRS selecciona la RFS y envía la solicitud.

	4	El sistema muestra mensaje de confirmación, finalizando el caso de uso.
Poscondición		
Excepciones	Paso	Acción
	2ª	No hay nuevas RFS en el sistema.
	4ª	El jefe de la OIRS puede volver a derivar otras RFS, para lo cual tiene que volver al paso 3.
Frecuencia	Alta	

Diagrama de secuencia

Figura 19: Diagrama de secuencia Derivar RFS.



Especificación Caso de Uso Detalle RFS

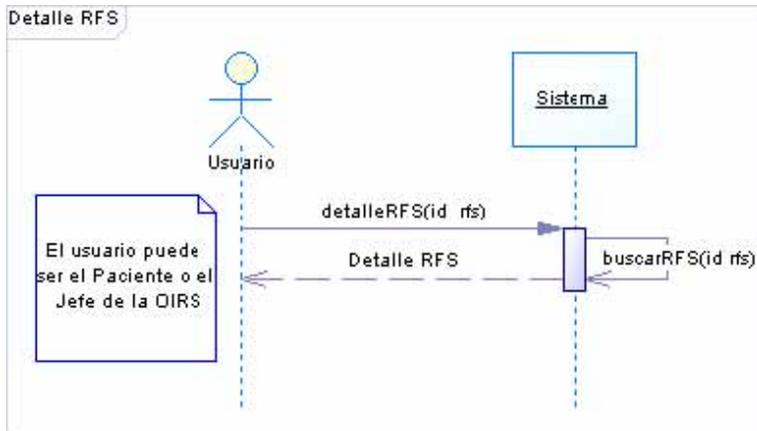
Tabla 17: Especificación Caso de Uso Detalle RFS.

--	--

Nombre	Detalle RFS.	
Actores	Paciente, Jefe OIRS.	
Descripción	En este caso de uso se describe la acción de ver el detalle de una RFS asociada a una paciente.	
Precondición	El usuario tiene que haberse identificado en el sistema.	
Escenario	Paso	Acción
Principal	1	El usuario selecciona la opción Ver detalle.
	2	El sistema muestra la RFS con toda la información asociada a ella.
Poscondición		
Excepciones	Paso	Acción
	2ª	El usuario presiona volver, cancelando la operación.
Frecuencia	Media	

Diagrama de secuencia

Figura 20: Diagrama de secuencia Detalle RFS.



Especificación Caso de Uso Consolidar Respuesta

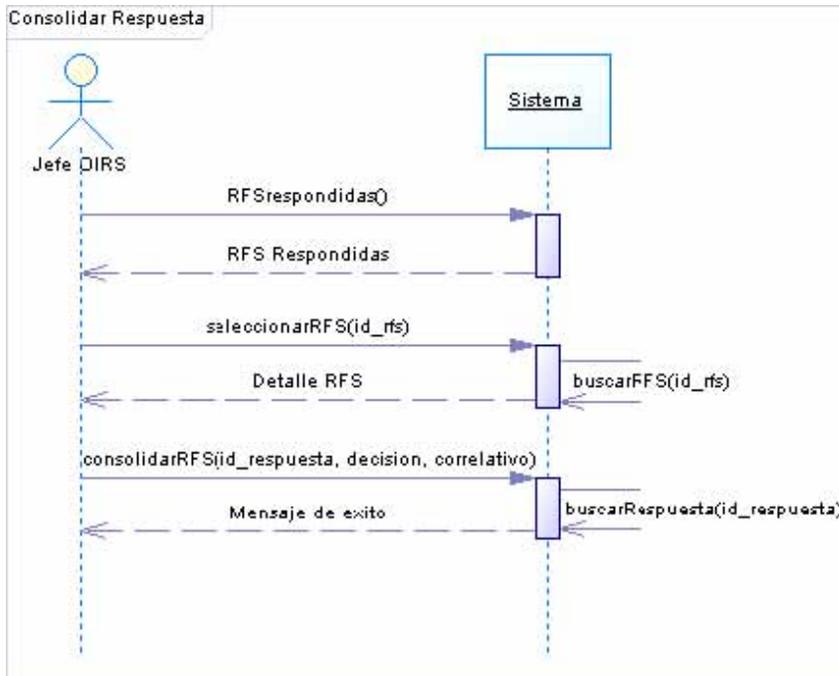
Tabla 18: Especificación Caso de Uso Consolidar Respuesta.

Nombre	Consolidar Respuesta.								
Actores	Jefe OIRS.								
Descripción	En este caso de uso se describe la acción en la cual el jefe de la OIRS verifica la respuesta dada por el jefe del servicio al paciente.								
Precondición	El jefe de la OIRS tiene que haberse identificado en el sistema.								
Escenario Principal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El jefe de la OIRS selecciona la opción RFS Respondidas.</td> </tr> <tr> <td>2</td> <td>El sistema muestra interfaz con las RFS respondidas.</td> </tr> <tr> <td>3</td> <td>El jefe de la OIRS selecciona una RFS.</td> </tr> </tbody> </table>	Paso	Acción	1	El jefe de la OIRS selecciona la opción RFS Respondidas.	2	El sistema muestra interfaz con las RFS respondidas.	3	El jefe de la OIRS selecciona una RFS.
Paso	Acción								
1	El jefe de la OIRS selecciona la opción RFS Respondidas.								
2	El sistema muestra interfaz con las RFS respondidas.								
3	El jefe de la OIRS selecciona una RFS.								

	4	El sistema muestra RFS en detalle con la respuesta
	5	El jefe de la OIRS ingresa una decisión, con el objeto de consolidar la respuesta.
	6	El jefe de la OIRS presiona guardar.
	7	El sistema muestra mensaje de confirmación, finalizando el caso de uso.
Poscondición	El sistema almacena la decisión, quedando disponible para que pueda ser vista por el paciente que realizó la RFS.	
Excepciones	Paso	Acción
	5 ^a	El jefe de la OIRS presiona volver, retornando al paso 3.
Frecuencia	Alta	

Diagrama de secuencia

Figura 21: Diagrama de secuencia Consolidar Respuesta.



Especificación Caso de Uso RFS Online

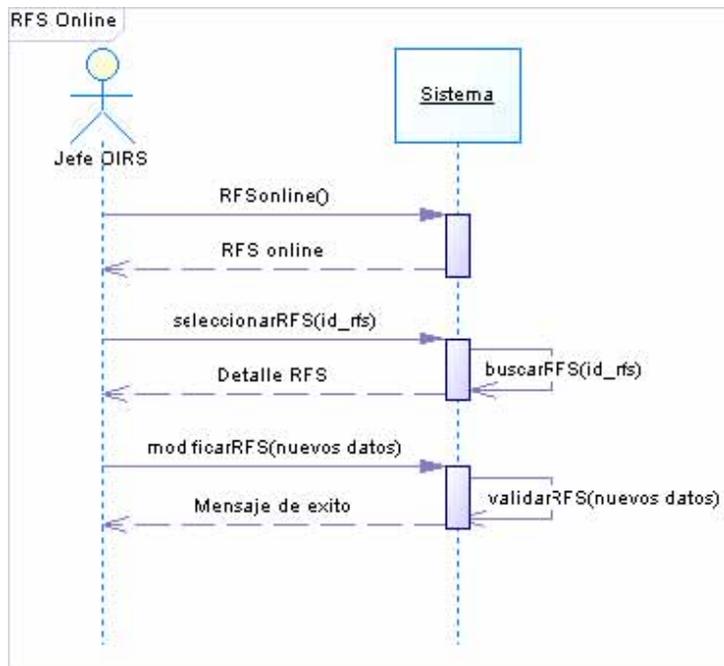
Tabla 19: Especificación Caso de Uso RFS Online.

Nombre	RFS Online.				
Actores	Jefe OIRS.				
Descripción	En este caso de uso se describe la acción en la cual el jefe de la OIRS modifica el contenido de una RFS hecha por un paciente a través del sistema en línea, ya sea por contar con información poco clara o que falte el respeto a alguna persona perteneciente al recinto hospitalario.				
Precondición	El jefe de la OIRS tiene que haberse identificado en el sistema.				
Escenario	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> </tr> </tbody> </table>	Paso	Acción		
Paso	Acción				

Principal	1	El jefe de la OIRS selecciona la opción RFS Online.
	2	El sistema muestra todas las RFS ingresadas por pacientes vía online.
	3	El jefe de la OIRS selecciona una RFS.
	4	El sistema muestra RFS en detalle.
	5	El jefe de la OIRS realiza las modificaciones que correspondan y presiona guardar.
	6	El sistema valida la información ingresada, mostrando un mensaje de éxito, finalizando el caso de uso.
Poscondición	La RFS es guardada en el sistema.	
Excepciones	Paso	Acción
	2 ^a	No se han ingresados RFS vía online.
	5 ^a	El jefe de la OIRS presiona volver, retornando al paso 3.
Frecuencia	Media	

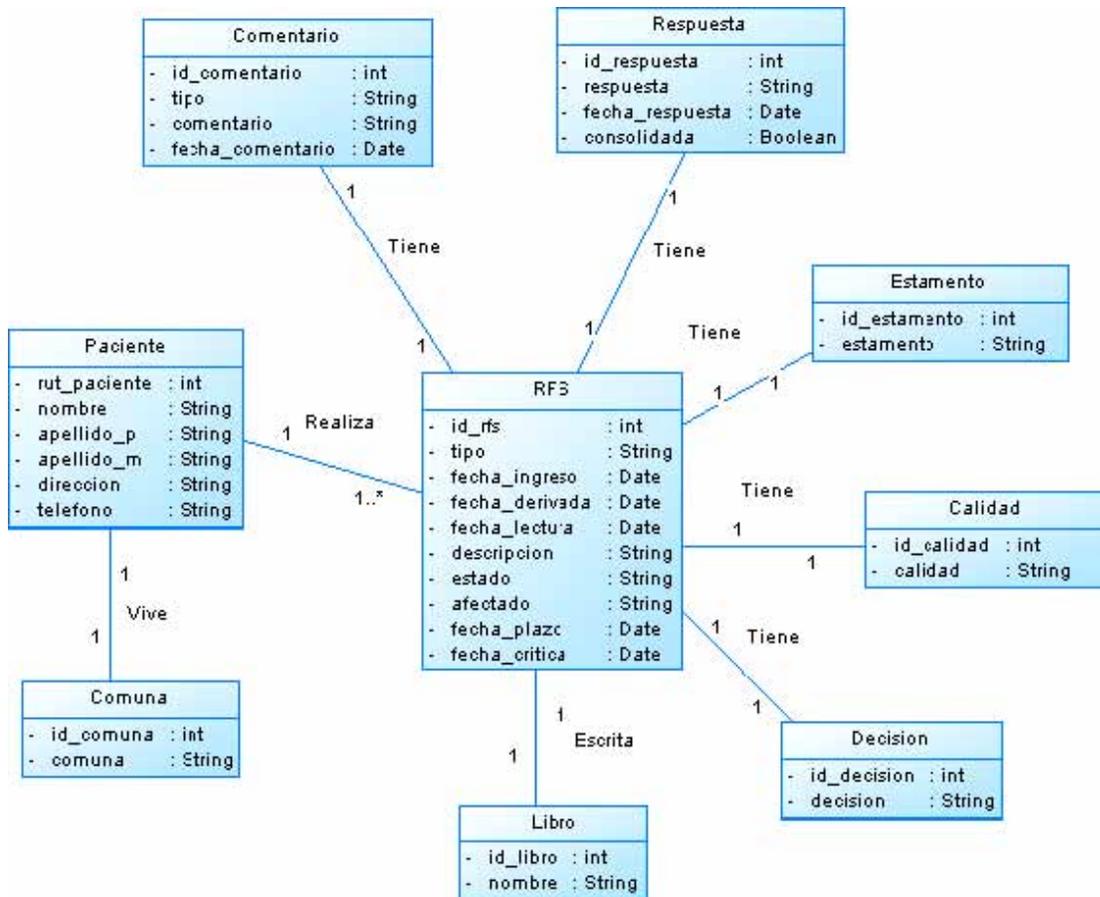
Diagrama de secuencia

Figura 22: Diagrama de secuencia RFS Online.



Modelo del Dominio del módulo

Figura 23: Modelo del dominio Módulo RFS.



Contratos de operaciones

Tabla 20: Contrato de operaciones ingresarRFS.

Operación	ingresarRFS(datos rfs)
Precondiciones	No hay.
Descripción	El usuario ingresa una nueva RFS al sistema.
Poscondiciones	<ul style="list-style-type: none"> - Se creó una instancia de RFS. - Si es primera vez que un paciente ingresa una RFS al sistema, se crea una instancia de Paciente.

	- Se creó una asociación entre Paciente y RFS.
--	--

Tabla 21: Contrato de operaciones validarRFS.

Operación	validarRFS(datos rfs)
Precondiciones	No hay.
Descripción	Se realiza la validación de la información de la RFS ingresada por el usuario.
Poscondiciones	Se modificaron los atributos de la instancia RFS que se creó: id_rfs, tipo, fecha_ingreso, descripción, calidad, libro.

Tabla 22: Contrato de operaciones seleccionarRFS.

Operación	seleccionarRFS(id_rfs)
Precondiciones	El usuario tiene que estar registrado en el sistema.
Descripción	El usuario hace clic en una de la RFS mostradas en pantalla, con el objeto de realizar alguna operación.
Poscondiciones	No hay.

Tabla 23: Contrato de operaciones buscarRFS.

--	--

Operación	buscarRFS(id_rfs)
Precondiciones	El usuario tiene que estar registrado en el sistema.
Descripción	Se realiza una búsqueda de la RFS a través de su identificador.
Poscondiciones	No hay.

Tabla 24: Contrato de operaciones ingresarRespuesta.

Operación	ingresarRespuesta(respuesta)
Precondiciones	<ul style="list-style-type: none"> - El Jefe del Servicio tiene que estar registrado en el sistema. - Tienen que haber RFS derivadas al servicio.
Descripción	Se ingresa la respuesta asociada a la RFS que fue derivada al servicio.
Poscondiciones	<ul style="list-style-type: none"> - Se creó una nueva instancia de Respuesta. - Se creó una asociación entre RFS y Respuesta.

Tabla 25: Contrato de operaciones validarRespuesta.

Operación	validarRespuesta(respuesta)
Precondiciones	No hay.
Descripción	Se realiza la validación de la respuesta ingresada por el usuario.

Poscondiciones	Se modificaron los atributos de la instancia Respuesta que se creó: id_respuesta, respuesta y fecha_respuesta.
-----------------------	---

Tabla 26: Contrato de operaciones derivarRFS.

Operación	derivarRFS(id_rfs)
Precondiciones	<ul style="list-style-type: none"> - El Jefe de la OIRS tiene que estar registrado en el sistema. - Tienen que haber RFS ingresadas en el sistema.
Descripción	El Jefe de la OIRS selecciona una RFS con el objeto de que el Jefe del Servicio le de una respuesta.
Poscondiciones	<ul style="list-style-type: none"> - Se creó una nueva instancia de Servicio. - Se creó una asociación entre RFS y Servicio.

Tabla 27: Contrato de operaciones verDetalle.

Operación	verDetalle(id_rfs)
Precondiciones	<ul style="list-style-type: none"> - El usuario tiene que estar registrado en el sistema. - Tienen que existir RFS en el sistema.
Descripción	Se muestra el detalle asociado a una RFS.
Poscondiciones	No hay.

Tabla 28: Contrato de operaciones consolidarRFS.

Operación	consolidarRFS(id_respuesta, consolidada)
Precondiciones	<ul style="list-style-type: none"> - El usuario tiene que estar registrado en el sistema. - Tienen que haber sido ingresada la respuesta previamente.
Descripción	Se consolida la respuesta ingresada por el Jefe del Servicio.
Poscondiciones	Se modificó el atributo consolidada de la instancia de Respuesta.

Tabla 29: Contrato de operaciones buscarRespuesta.

Operación	buscarRespuesta(id_respuesta)
Precondiciones	<ul style="list-style-type: none"> - El usuario tiene que estar registrado en el sistema. - Tienen que existir respuestas en el sistema asociadas a RFS.
Descripción	Se realiza una búsqueda de la respuesta.
Poscondiciones	No hay.

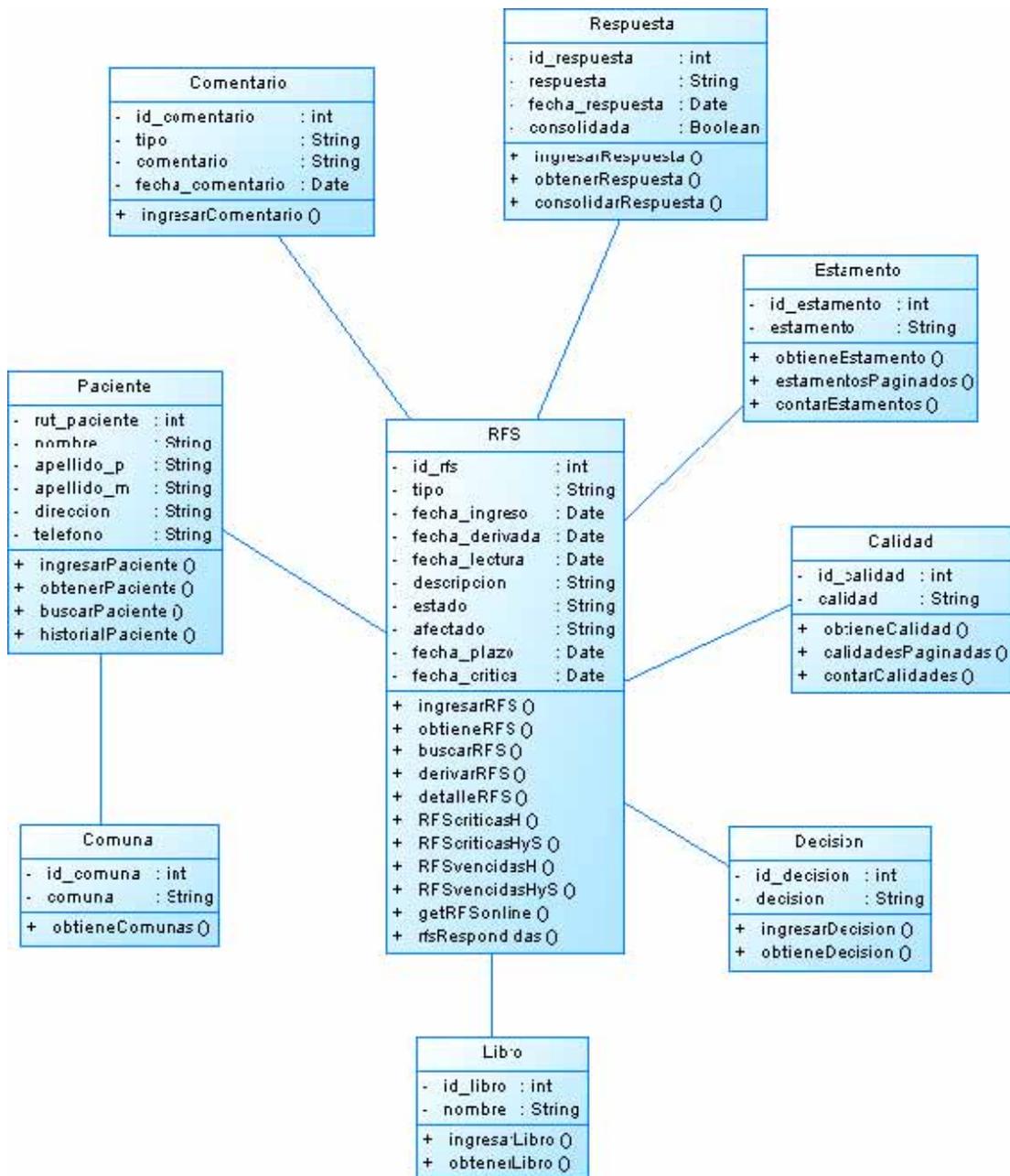
Tabla 30: Contrato de operaciones RFSOnline.

Operación	RFSOnline(comentario, calidad, estamento)
Precondiciones	- El usuario tiene que estar registrado en el sistema.

	- Tienen que existir la RFS en el sistema.
Descripción	Se modifican los campos de la RFS ingresada.
Poscondiciones	Se modifican los atributos de la instancia de la RFS creada: tipo, descripción, calidad, libro.

Modelo de Clases del módulo

Figura 24: Modelo de clases Módulo RFS.



10.1.2. Iteración 2: Módulo Administración

Especificación Caso de Uso Mostrar Usuarios

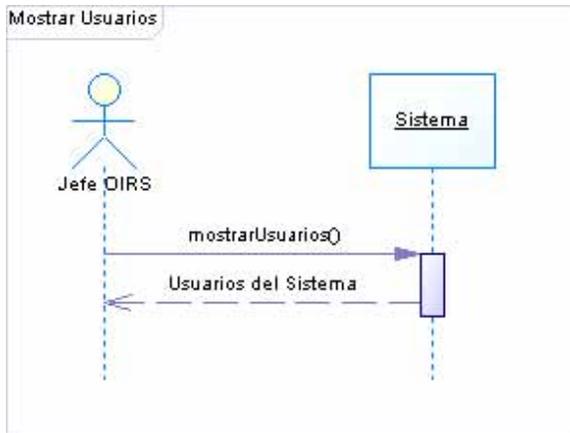
Tabla 31: Especificación Caso de Uso Mostrar Usuarios.

--	--

Nombre	Mostrar Usuarios.	
Actores	Jefe OIRS.	
Descripción	En este caso de uso se describe la acción de ver todos los usuarios que tiene el sistema.	
Precondición	El jefe de la OIRS tiene que haberse identificado en el sistema.	
Escenario Principal	Paso	Acción
	1	El jefe de la OIRS selecciona Mostrar Usuarios.
	2	El sistema muestra todos los usuarios que se encuentran registrados.
Poscondición		
Excepciones	Paso	Acción
	2 ^a	No existen usuarios en el sistema.
Frecuencia	Baja	

Diagrama de secuencia

Figura 25: Diagrama de secuencia Mostrar Usuarios.



Especificación Caso de Uso Historial Paciente

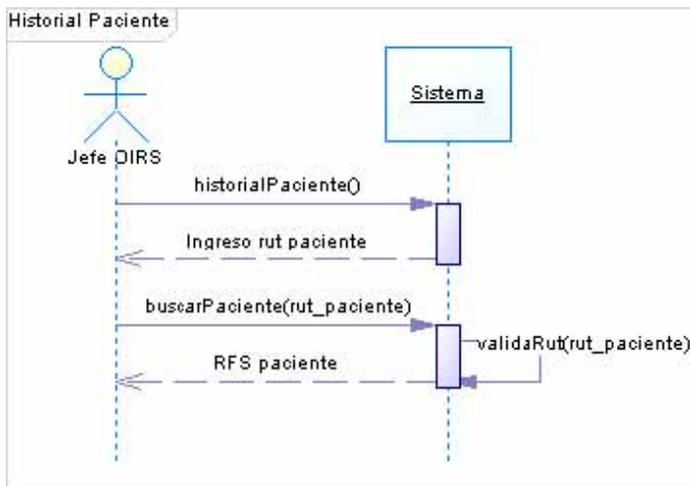
Tabla 32: Especificación Caso de Uso Historial Paciente.

Nombre	Historial Paciente.	
Actores	Jefe OIRS.	
Descripción	En este caso de uso se describe la acción de ver todas las RFS hechas por un paciente.	
Precondición	El jefe de la OIRS tiene que haberse identificado en el sistema.	
Escenario	Paso	Acción
Principal	1	El jefe de la OIRS selecciona Historial Paciente.
	2	El sistema muestra interfaz donde el usuario tiene que ingresar el rut del paciente.
	3	El jefe de la OIRS ingresa el rut del paciente.
	4	El sistema muestra las RFS asociadas al paciente.

Poscondición		
Excepciones	Paso	Acción
	4 ^a	El usuario no ha ingresado RFS, no existiendo en la base de datos del sistema.
Frecuencia	Baja	

Diagrama de secuencia

Figura 26: Diagrama de secuencia Historial Paciente.



Especificación Caso de Uso Crear usuario

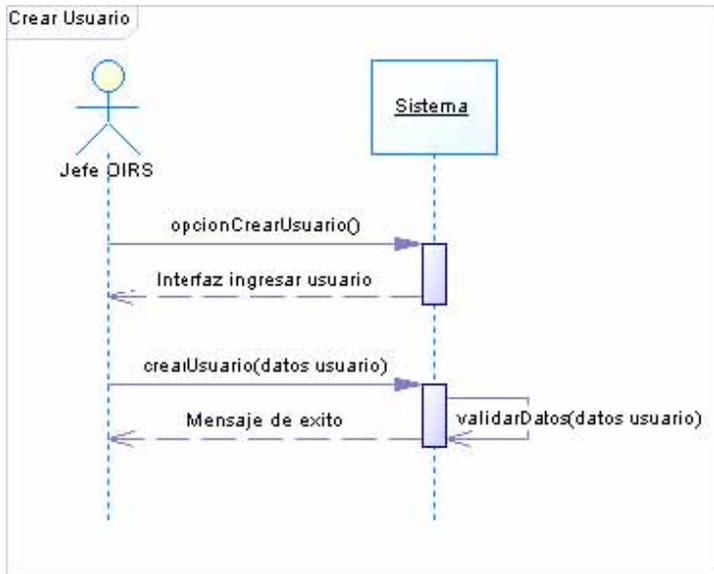
Tabla 33: Especificación Caso de Uso Crear usuario.

Nombre	Crear usuario.
Actores	Jefe OIRS.
Descripción	En este caso de uso se describe la acción en la cual el jefe de la OIRS crea un nuevo usuario para acceder al sistema.

Precondición	El jefe de la OIRS tiene que haberse identificado en el sistema.	
Escenario	Paso	Acción
Principal	1	El jefe de la OIRS selecciona la opción Crear Usuario.
	2	El sistema muestra interfaz con los campos a ser ingresados en el sistema.
	3	El jefe de la OIRS ingresa la información solicitada por el sistema.
	4	El sistema valida los datos ingresados, mostrando un mensaje de éxito, finalizando el caso de uso.
Poscondición	El sistema cuenta un nuevo usuario.	
Excepciones	Paso	Acción
	3 ^a	El jefe de la OIRS presiona volver, cancelando la operación.
	4 ^a	Si los datos no son correctos, el sistema muestra mensaje de error, volviendo al paso 3.
Frecuencia	Baja.	

Diagrama de secuencia

Figura 27: Diagrama de secuencia Crear Usuario.



Especificación Caso de Uso Modificar usuario

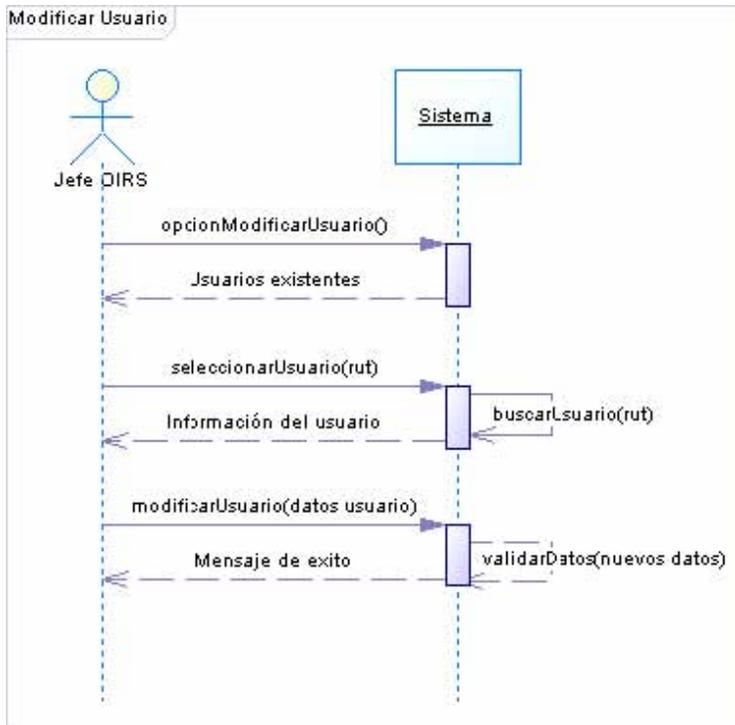
Tabla 34: Especificación Caso de Uso Modificar usuario.

Nombre	Modificar usuario.						
Actores	Jefe OIRS.						
Descripción	En este caso de uso se describe la acción en la cual el jefe de la OIRS modifica la información relacionada a un usuario.						
Precondición	El jefe de la OIRS tiene que haberse identificado en el sistema.						
Escenario	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El jefe de la OIRS selecciona la opción Modificar Usuario.</td> </tr> <tr> <td>2</td> <td>El sistema muestra interfaz con los usuarios</td> </tr> </tbody> </table>	Paso	Acción	1	El jefe de la OIRS selecciona la opción Modificar Usuario.	2	El sistema muestra interfaz con los usuarios
Paso	Acción						
1	El jefe de la OIRS selecciona la opción Modificar Usuario.						
2	El sistema muestra interfaz con los usuarios						
Principal							

		existentes en el sistema.
	3	El jefe de la OIRS selecciona el usuario a ser modificado.
	4	El sistema muestra la información asociada al usuario.
	5	El jefe de la OIRS ingresa la nueva información.
	6	El sistema valida la información ingresada, mostrando un mensaje de éxito, finalizando el caso de uso.
Poscondición	La información asociada al usuario ha sido actualizada en la base de datos.	
Excepciones	Paso	Acción
	3 ^a	El jefe de la OIRS presiona volver, cancelando la operación.
	5 ^a	El jefe de la OIRS presiona volver, volviendo al paso anterior.
	6 ^a	Si los datos no son correctos, el sistema muestra mensaje de error, volviendo al paso 5.
Frecuencia	Baja.	

Diagrama de secuencia

Figura 28: Diagrama de secuencia Modificar Usuario.



Especificación Caso de Uso Eliminar usuario

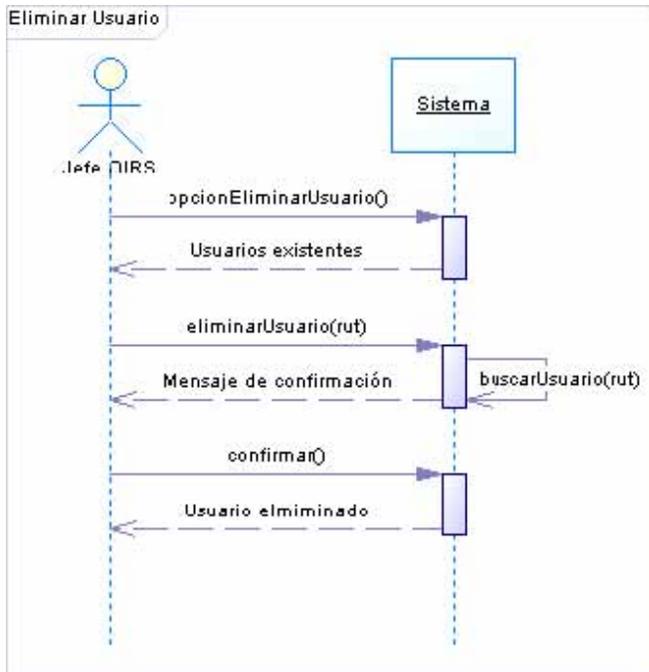
Tabla 35: Especificación Caso de Uso Eliminar usuario.

Nombre	Eliminar usuario.	
Actores	Jefe OIRS.	
Descripción	En este caso de uso se describe la acción en la cual el jefe de la OIRS elimina un usuario del sistema.	
Precondición	El jefe de la OIRS tiene que haberse identificado en el sistema.	
Escenario	Paso	Acción
Principal	1	El jefe de la OIRS selecciona la opción Eliminar Usuario.

	2	El sistema muestra interfaz con los usuarios existentes en el sistema.
	3	El jefe de la OIRS selecciona el usuario que desea eliminar.
	4	El sistema muestra mensaje de confirmación.
	5	El jefe de la OIRS confirma la operación.
	6	El sistema elimina el usuario del sistema, finalizando el caso de uso.
Poscondición	El usuario ha sido eliminado de la base de datos del sistema.	
Excepciones	Paso	Acción
	3 ^a	El jefe de la OIRS presiona volver, cancelado la operación.
	5 ^a	El jefe de la OIRS presiona cancelar, volviendo al paso 2
Frecuencia	Baja.	

Diagrama de secuencia

Figura 29: Diagrama de secuencia Eliminar Usuario.



Especificación Caso de Uso Modificar Contraseña

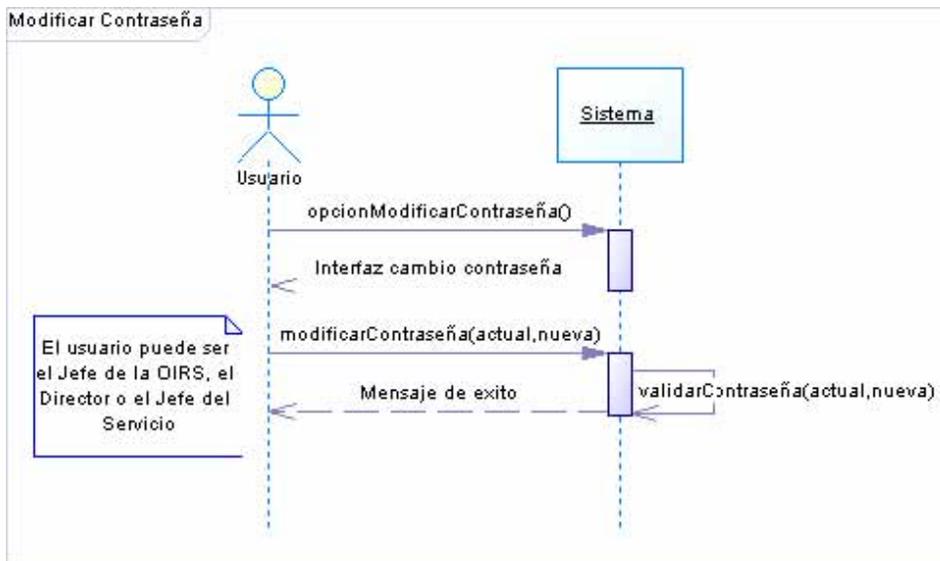
Tabla 36: Especificación Caso de Uso Modificar Contraseña.

Nombre	Modificar Contraseña.	
Actores	Jefe OIRS, Director, Jefe Servicio.	
Descripción	En este caso de uso se describe la acción de modificar la contraseña de acceso al sistema.	
Precondición	El usuario tiene que haberse identificado en el sistema.	
Escenario	Paso	Acción
Principal	1	El usuario selecciona la opción Modificar contraseña.
	2	El sistema muestra interfaz de cambio de

		contraseña.
	3	El usuario ingresa la contraseña actual y la nueva.
	4	El sistema valida la información, modificando la contraseña.
Poscondición	El sistema actualiza los datos del usuario.	
Excepciones	Paso	Acción
	2ª	El usuario cancela la operación, volviendo al inicio del sistema.
	4ª	La contraseña ingresada como actual es incorrecta.
Frecuencia	Baja	

Diagrama de secuencia

Figura 30: Diagrama de secuencia Modificar Contraseña.



Modelo del Dominio del módulo

Figura 31: Modelo del dominio Módulo Administración.



Contratos de operaciones

Tabla 37: Contrato de operaciones buscarPaciente.

Operación	buscarPaciente(rut)
Precondiciones	<ul style="list-style-type: none"> - El Jefe de la OIRS tiene que estar registrado en el sistema. - El paciente tiene que existir en la base de datos del sistema.
Descripción	Se busca al paciente en el sistema, con el objeto de mostrar la información asociada a este.
Poscondiciones	No hay.

Tabla 38: Contrato de operaciones crearUsuario.

Operación	crearUsuario(rut, nombre, apellido_p, apellido_m, contrasena, mail, tipo)
Precondiciones	El Jefe de la OIRS tiene que estar registrado en el sistema.
Descripción	Se crea un nuevo usuario en el sistema.
Poscondiciones	Se creó una nueva instancia de Usuario.

Tabla 39: Contrato de operaciones validarDatos.

Operación	validarDatos(rut, nombre, apellido_p, apellido_m, contrasena, mail, tipo)
Precondiciones	El Jefe de la OIRS tiene que estar registrado en el sistema.
Descripción	Se validan los datos ingresados por el Jefe de la OIRS.
Poscondiciones	Se modificaron los campos de la instancia creada: rut, nombre, apellido_p, apellido_m, contrasena, mail, tipo.

Tabla 40: Contrato de operaciones modificarUsuario.

Operación	modificarUsuario(nombre, apellido_p, apellido_m, contrasena, mail, tipo)
Precondiciones	El Jefe de la OIRS tiene que estar registrado en el sistema.
Descripción	Se modifica la información asociada a un usuario.
Poscondiciones	Se modificaron los campos de la instancia creada: nombre, apellido_p, apellido_m, contrasena, mail, tipo.

Tabla 41: Contrato de operaciones seleccionarUsuario.

Operación	seleccionarUsuario(rut)
Precondiciones	El Jefe de la OIRS tiene que estar registrado en el sistema.

Descripción	Se selecciona un usuario de los existentes.
Poscondiciones	No hay

Tabla 42: Contrato de operaciones buscarUsuario.

Operación	buscarUsuario(rut)
Precondiciones	El Jefe de la OIRS tiene que estar registrado en el sistema.
Descripción	Se realiza una búsqueda del usuario.
Poscondiciones	No hay

Tabla 43: Contrato de operaciones eliminarUsuario.

Operación	eliminarUsuario(rut)
Precondiciones	El Jefe de la OIRS tiene que estar registrado en el sistema.
Descripción	Se borra al usuario del sistema.
Poscondiciones	<ul style="list-style-type: none"> - Se eliminó la instancia del Usuario. - Se eliminó la asociación existente entre las instancias de Usuario y Servicio.

Tabla 44: Contrato de operaciones modificarContraseña.

--	--

Operación	modificarContraseña(actual,nueva)
Precondiciones	El Usuario tiene que estar registrado en el sistema.
Descripción	Se modifica la contraseña del usuario.
Poscondiciones	Se modificó el atributo contraseña de la instancia de Usuario.

Tabla 45: Contrato de operaciones validarContraseña.

Operación	validarContraseña(actual,nueva)
Precondiciones	El Usuario tiene que estar registrado en el sistema.
Descripción	Se validan los datos ingresados por el Usuario.
Poscondiciones	Se modificó el atributo contraseña de la instancia de Usuario.

Modelo de Clases del módulo

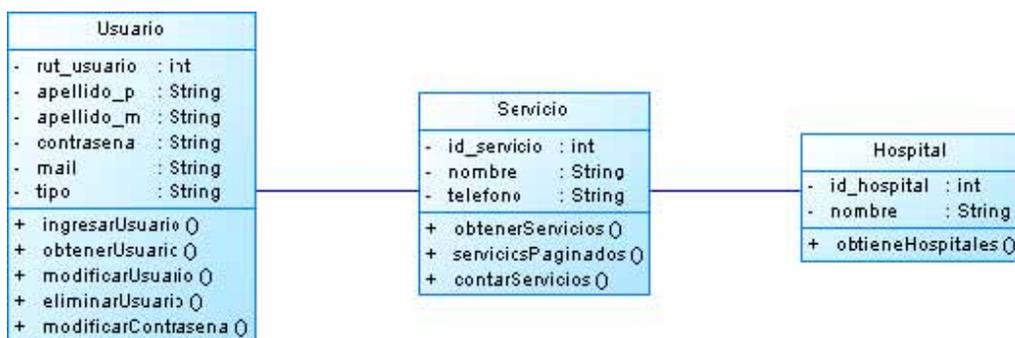


Figura 32: Modelo de clases Módulo Administración.

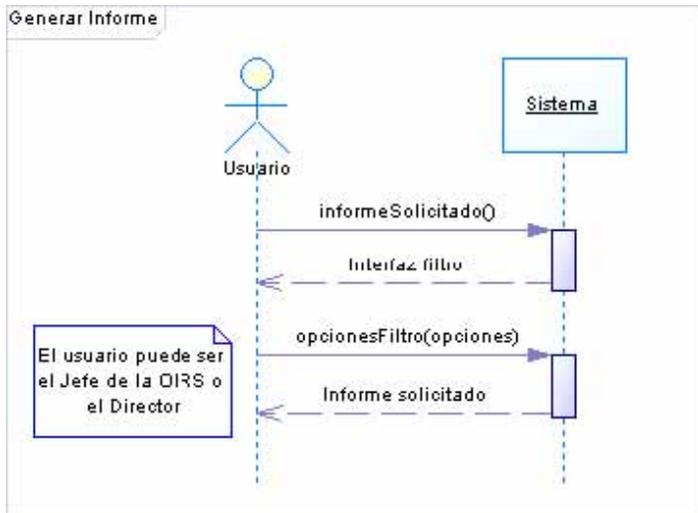
10.1.3. Iteración 3: Módulo Informes

Especificación Caso de Uso Generar Informe

Tabla 46: Especificación Caso de Uso Generar Informe

Nombre	Generar Informe	
Actores	Jefe OIRS, Director.	
Descripción	En este caso de uso se describe la acción en la cual el usuario genera uno de los informes disponibles.	
Precondición	El usuario tiene que haberse identificado en el sistema.	
Escenario Principal	Paso	Acción
	1	El usuario selecciona el tipo de informe que desea.
	2	El sistema muestra interfaz con los filtros asociados al informe.
	3	El usuario selecciona las opciones disponibles en el filtro.
	4	El sistema muestra informe con la información solicitada.
Poscondición	No hay.	
Excepciones	Paso	Acción
Frecuencia	Baja.	

Figura 33: Diagrama de secuencia Generar Informe.



Modelo del Dominio del módulo

Figura 34: Modelo del dominio Módulo Informes.

Usuario	
- rut_usuario	: int
- nombre	: String
- apellido_p	: String
- apellido_m	: String
- contrasena	: String
- mail	: String
- tipo	: String

Contratos de Operaciones

Operación	opcionesFiltro(opciones)
Precondiciones	El usuario tiene que estar registrado en el sistema.
Descripción	Se muestra el informe solicitado por el usuario de acuerdo a las opciones seleccionadas en el filtro del informe.
Poscondiciones	<ul style="list-style-type: none"> - Se creó una nueva instancia de Informe. - Se creó una asociación entre las instancias de Usuario e Informe.

Tabla 47: Contrato de operaciones opcionesFiltro.

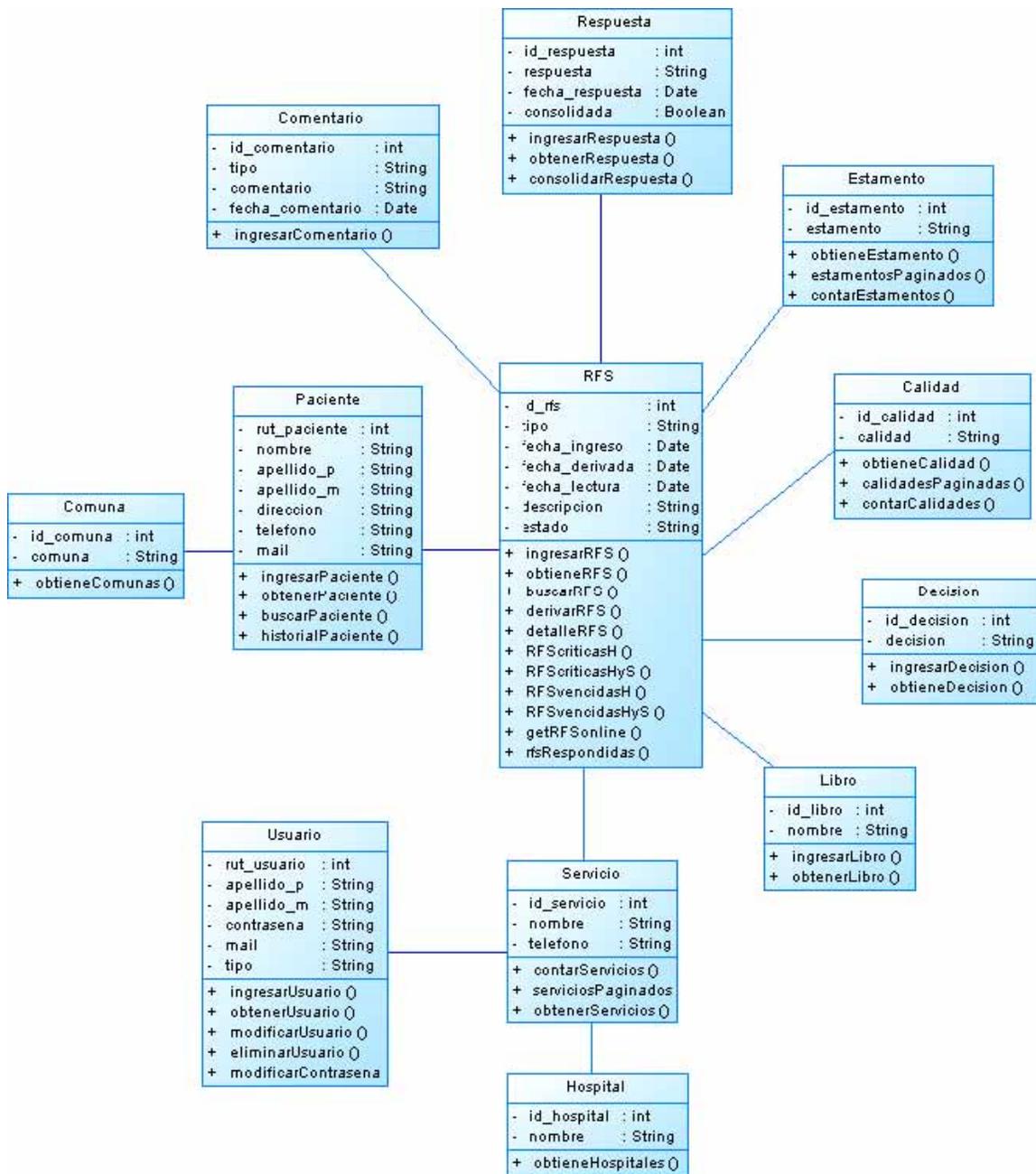
Modelo de Clases del módulo

Figura 35: Modelo de clases Módulo Informes.

Usuario	
- rut_usuario	: int
- nombre	: String
- apellido_p	: String
- apellido_m	: String
- constrasena	: String
- mail	: String
- tipo	: String
+ ingresarUsuario ()	
+ obtenerUsuario ()	
+ modificarUsuario ()	
+ eliminarUsuario ()	
+ modificarContrasena	

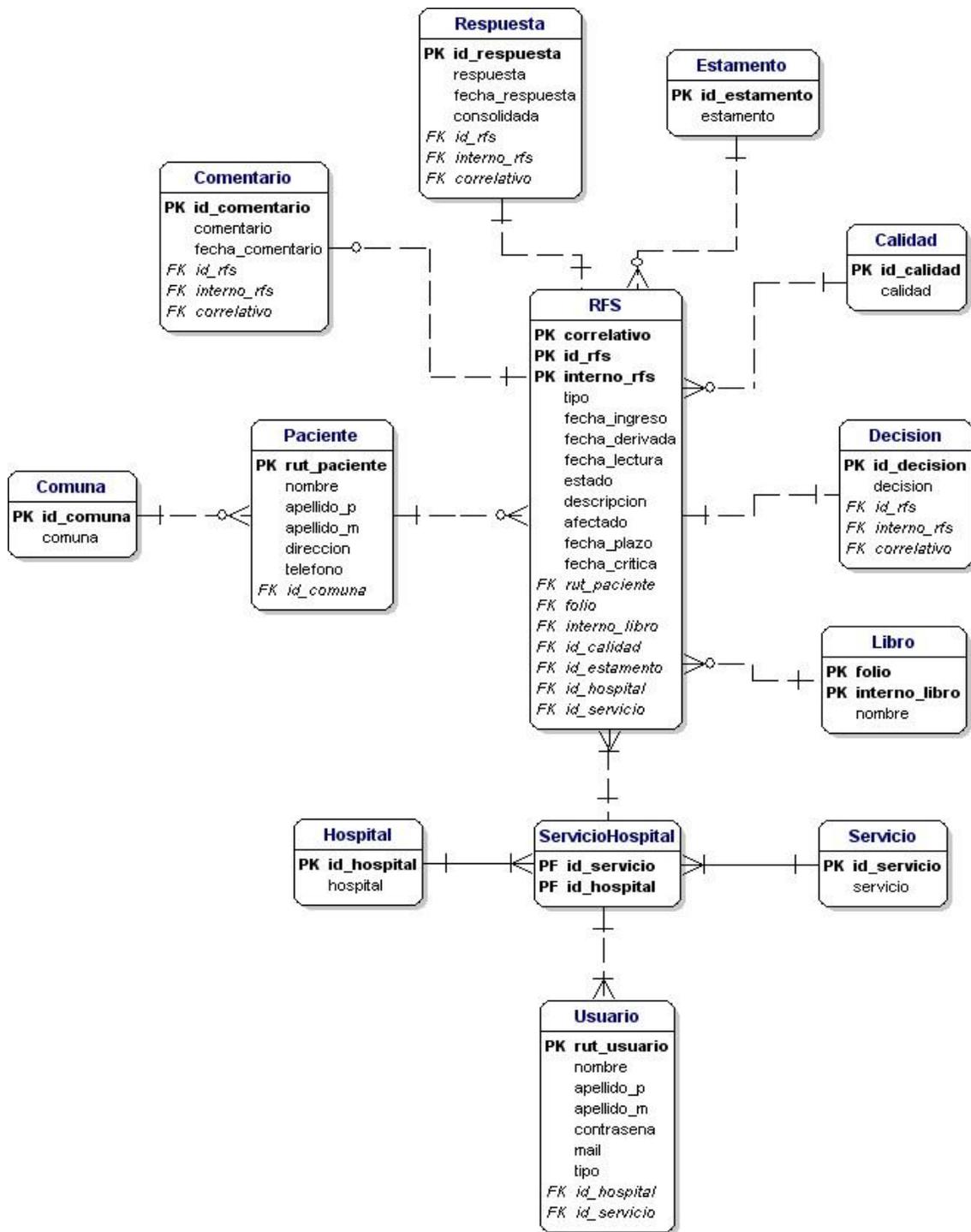
Modelo de Clases Integrado del Sistema

Figura 36: Modelo de clases integrado Sistema OIRS.



Modelo de Datos integrado del sistema

Figura 37: Modelo de datos integrado Sistema OIRS.



10.2. Sistema de Solicitudes de Atención

Diagrama General de Casos de uso

Figura 38: Casos de uso Empleado Servicio.



Figura 39: Casos de uso Trabajador.



Figura 40: Casos de uso Director.

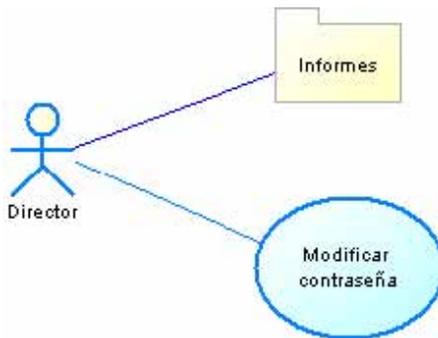
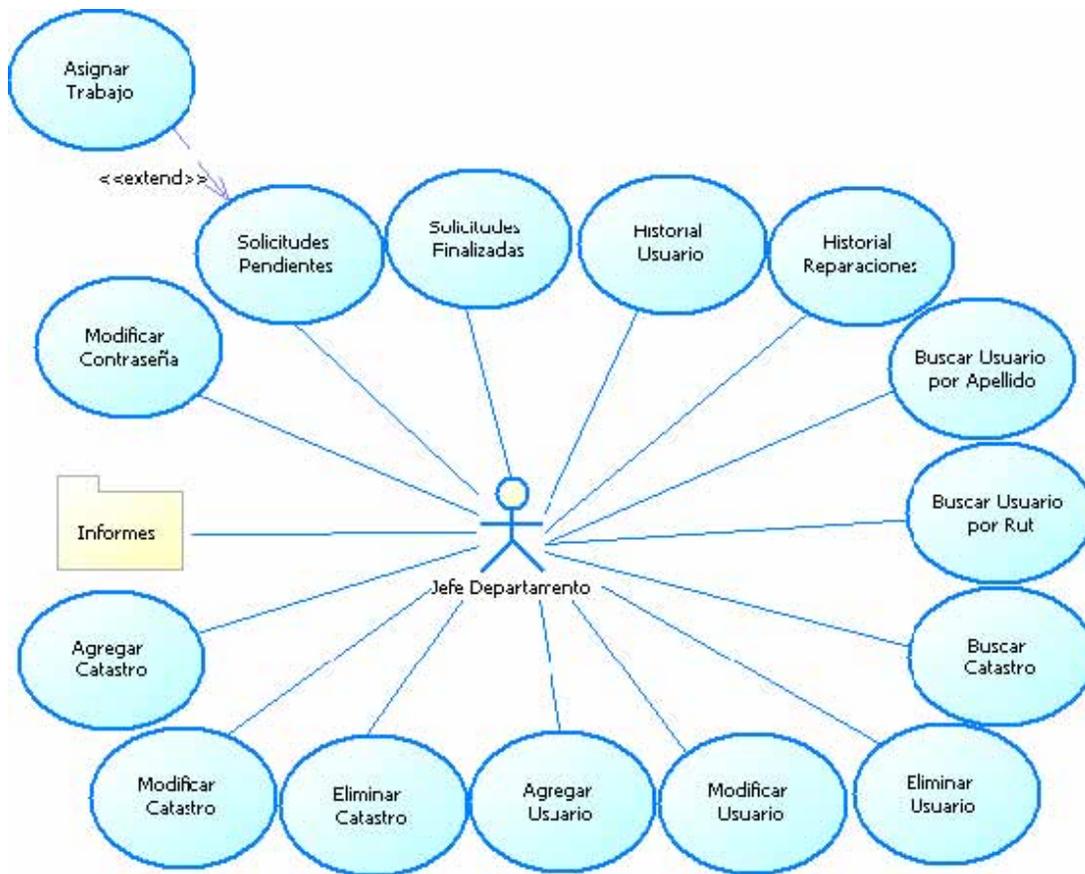


Figura 41: Casos de uso Jefe Departamento.



10.2.1. Iteración 4: Módulo Solicitudes

Especificación Caso de Uso Login

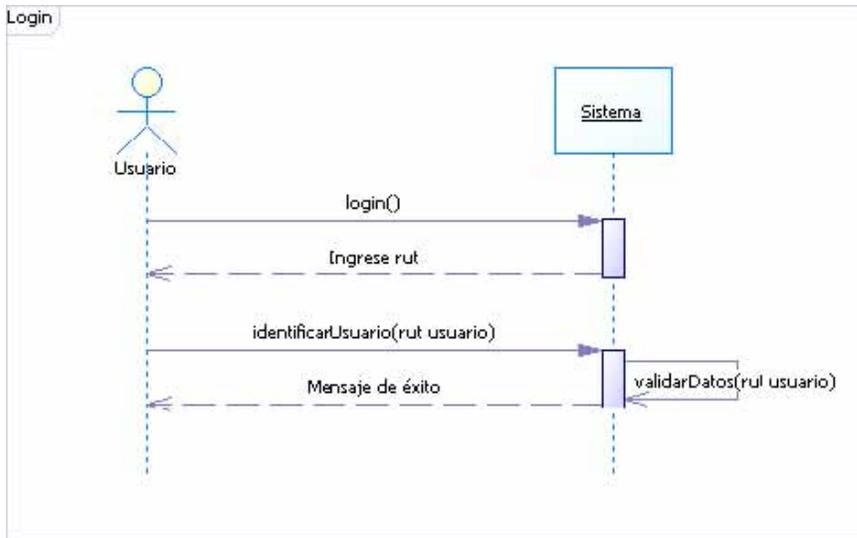
Tabla 48: Especificación Caso de Uso Login.

Nombre	Login
Actores	Usuario
Descripción	Este caso de uso permite la identificación de un usuario en el sistema.
Precondición	El Usuario existe en el sistema.

Escenario	Paso	Acción
Principal	1	El Usuario ingresa al sistema.
	2	El sistema solicita el ingreso del Rut y contraseña del usuario.
	3	El Usuario ingresa los datos solicitados.
	4	El sistema lo identifica como un usuario valido y le entrega los privilegios del perfil correspondiente.
Poscondición	No hay.	
Excepciones	Paso	Acción
	3a	Si el Rut ingresado es inválido, el sistema mostrará un mensaje solicitando que escriba uno que sí lo sea.
	3b	Si la contraseña es inválida, el sistema mostrará un mensaje solicitando que la reingrese.
	4a	Si el usuario no existe, el sistema mostrará un mensaje indicándolo.
Frecuencia	Alta	

Diagrama de Secuencia

Figura 42: Diagrama de Secuencia Login.



Especificación Caso de Uso Ingresar Solicitud

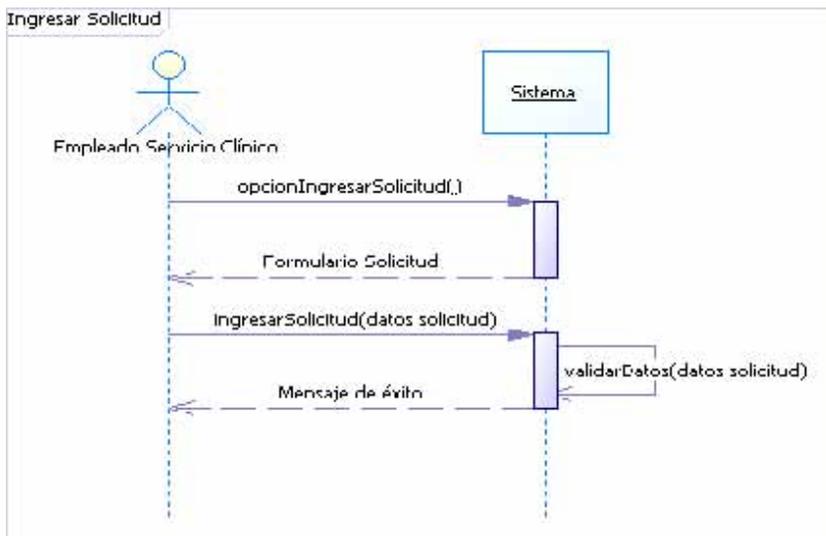
Nombre	Ingresar Solicitud.								
Actores	Empleado Servicio.								
Descripción	Este caso de uso permite el ingreso de una solicitud de atención por parte del usuario.								
Precondición	El Usuario se ha identificado en el sistema.								
Escenario	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El usuario selecciona la opción de <i>ingresar solicitud</i>.</td> </tr> <tr> <td>2</td> <td>El sistema entrega la interfaz con el formulario de solicitud.</td> </tr> <tr> <td>3</td> <td>El usuario ingresa todos los datos solicitados y envía la solicitud.</td> </tr> </tbody> </table>	Paso	Acción	1	El usuario selecciona la opción de <i>ingresar solicitud</i> .	2	El sistema entrega la interfaz con el formulario de solicitud.	3	El usuario ingresa todos los datos solicitados y envía la solicitud.
Paso	Acción								
1	El usuario selecciona la opción de <i>ingresar solicitud</i> .								
2	El sistema entrega la interfaz con el formulario de solicitud.								
3	El usuario ingresa todos los datos solicitados y envía la solicitud.								
Principal									

	4	El sistema valida la información.
Poscondición	El sistema almacenó el formulario.	
Excepciones	Paso	Acción
	3a	El usuario decide no enviar la solicitud y el sistema vuelve al inicio.
Frecuencia	Alta.	

Tabla 49: Especificación Caso de Uso Ingresar Solicitud.

Diagrama de Secuencia

Figura 43: Diagrama de secuencia Ingresar Solicitud.



Especificación Caso de Uso Asignar Trabajo

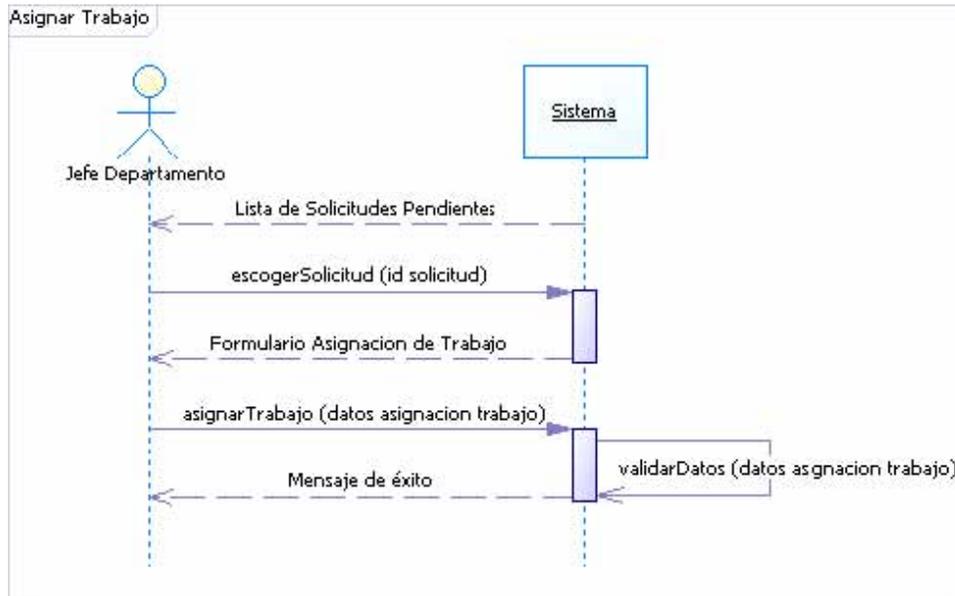
Nombre	Asignar Trabajo.
Actores	Jefe Departamento.

Descripción	Este caso de uso permite la asignación de las solicitudes de atención en espera a un trabajador o grupo de trabajo.	
Precondición	El Jefe de Departamento se ha identificado en el sistema y se encuentra en la interfaz de revisión de cola de solicitudes.	
Escenario Principal	Paso	Acción
	1	El Jefe de Departamento selecciona la opción para asignar trabajo.
	2	El sistema muestra la interfaz con el formulario de los nombres de los trabajadores disponibles.
	3	El Jefe de Departamento asigna a un(os) trabajador(es) e ingresa las horas/hombre estimadas para la realización de la tarea.
	4	El sistema valida y almacena la información.
Poscondición	La solicitud de trabajo cambia su estado a <i>en proceso</i> .	
Excepciones	Paso	Acción
	2a	En caso de no haber trabajadores disponibles, el sistema muestra mensaje indicando la inexistencia de trabajadores disponibles y vuelve a la revisión de cola de solicitudes.
	3a	En caso de que el Jefe de Departamento decida no enviar la solicitud, el sistema vuelve a la revisión de solicitudes pendientes.
Frecuencia	Alta.	

Tabla 50: Especificación Caso de Uso Asignar Trabajo.

Diagrama de Secuencia

Figura 44: Diagrama de secuencia Asignar Trabajo.



Especificación Caso de Uso Historial Usuario

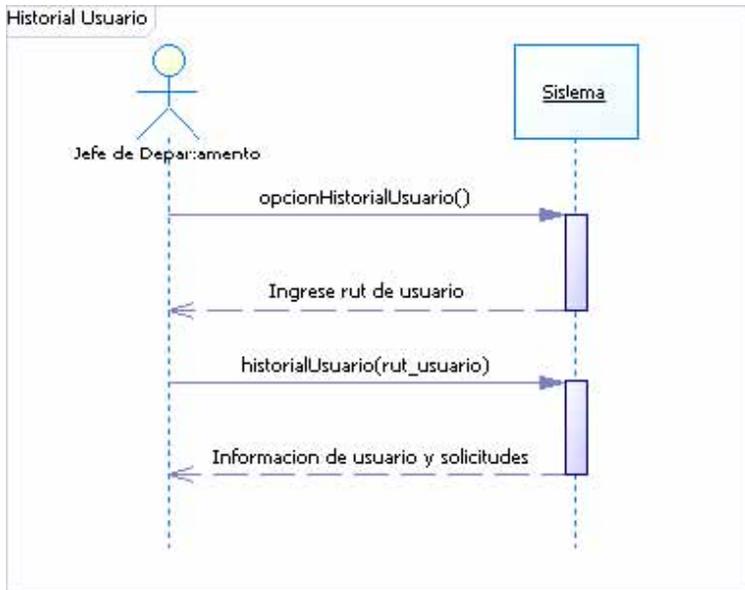
Tabla 51: Especificación Caso de Uso Historial Usuario.

Nombre	Historial Usuario.				
Actores	Jefe Departamento.				
Descripción	Este caso de uso permite la revisión en detalle de la(s) solicitud(es) asociada(s) de un usuario en particular.				
Precondición	El Jefe de Departamento se ha identificado en el sistema.				
Escenario	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td> </td> <td> </td> </tr> </tbody> </table>	Paso	Acción		
Paso	Acción				

Principal	1	El Jefe de Departamento selecciona la opción para ver el historial del usuario.
	2	El sistema solicita el ingreso del rut del usuario.
	3	El Jefe del Departamento ingresa el rut del usuario.
	4	El sistema muestra en detalle los datos del usuario y la(s) solicitud(es) asociada(s) a este.
Poscondición	No hay.	
Excepciones	Paso	Acción
	3a	En caso que el Jefe del Departamento decida no realizar la consulta, el sistema vuelve al inicio.
	3b	En caso de que el usuario no exista o no tenga solicitudes asociadas, el sistema mostrará un mensaje especificando lo sucedido.
Frecuencia	Alta.	

Diagrama de Secuencia

Figura 45: Diagrama de secuencia Historial Usuario.



Especificación Caso de Uso Solicitudes Pendientes

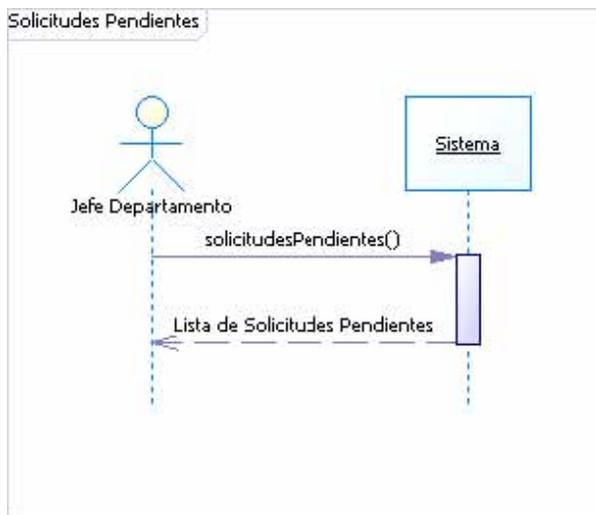
Tabla 52: Especificación Caso de Uso Solicitudes Pendientes.

Nombre	Solicitudes Pendientes	
Actores	Jefe Departamento.	
Descripción	Este caso de uso permite la revisión de las solicitudes de atención pendientes por asignar.	
Precondición	El Jefe de Departamento se ha identificado en el sistema.	
Escenario Principal	Paso	Acción
	1	El Jefe de Departamento selecciona la opción para ver las solicitudes pendientes.
	2	El sistema muestra la interfaz con todas las solicitudes pendientes por asignar a los empleados del departamento.

Poscondición	No hay.	
Excepciones	Paso	Acción
	2a	Si no hay solicitudes pendientes, el sistema muestra mensaje indicando inexistencia de solicitudes pendientes y vuelve al inicio.
Frecuencia	Alta.	

Diagrama de Secuencia

Figura 46: Diagrama de secuencia Solicitudes Pendientes.



Especificación Caso de Uso Finalizar Trabajo

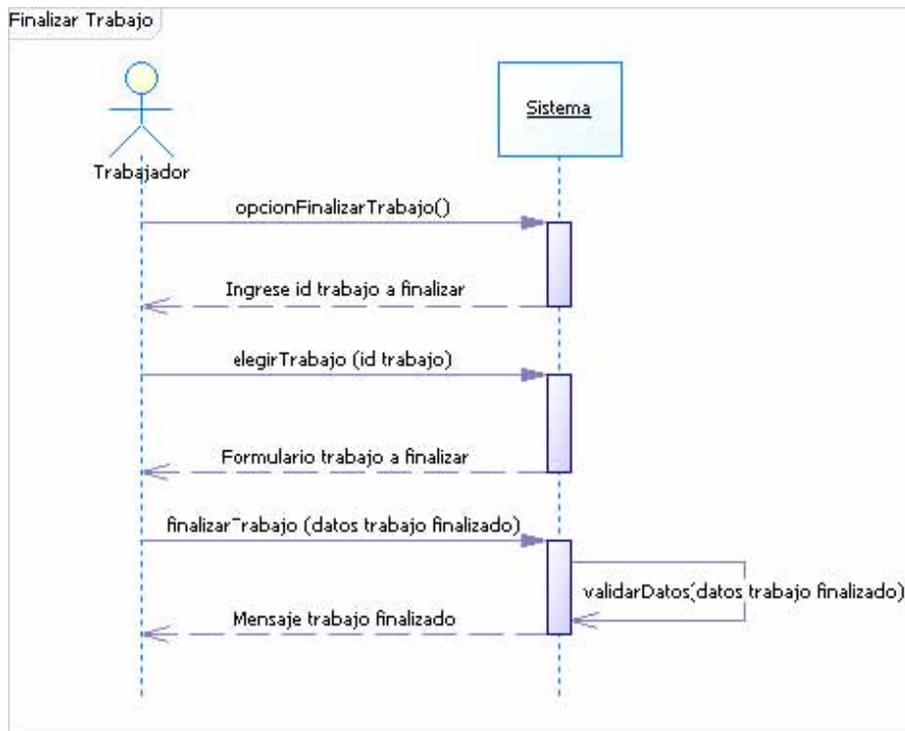
Tabla 53: Especificación Caso de Uso Finalizar Trabajo.

Nombre	Finalizar Trabajo.
Actores	Trabajador.

Descripción	Este caso de uso permite a un trabajador dar por finalizado un trabajo y especificar los detalles de este.	
Precondición	El Trabajador se ha identificado en el sistema, se encuentra revisando la cola de trabajos pendientes y ha finalizado una tarea asignada.	
Escenario Principal	Paso	Acción
	1	El trabajador selecciona la opción para finalizar trabajo de uno de los que se presentan en la lista.
	2	El sistema entrega la información del trabajo con las secciones que faltan por rellenar.
	3	El trabajador ingresa la información faltante.
	4	El sistema valida la información y finaliza la tarea.
Poscondición	El estado de la solicitud ha cambiado a <i>finalizado</i> .	
Excepciones	Paso	Acción
	1a	En caso de no haber trabajos pendientes, el sistema muestra mensaje y vuelve al inicio.
	3a	En caso de que el trabajador decida no finalizar el trabajo, el sistema vuelve al inicio.
Frecuencia	Alta.	

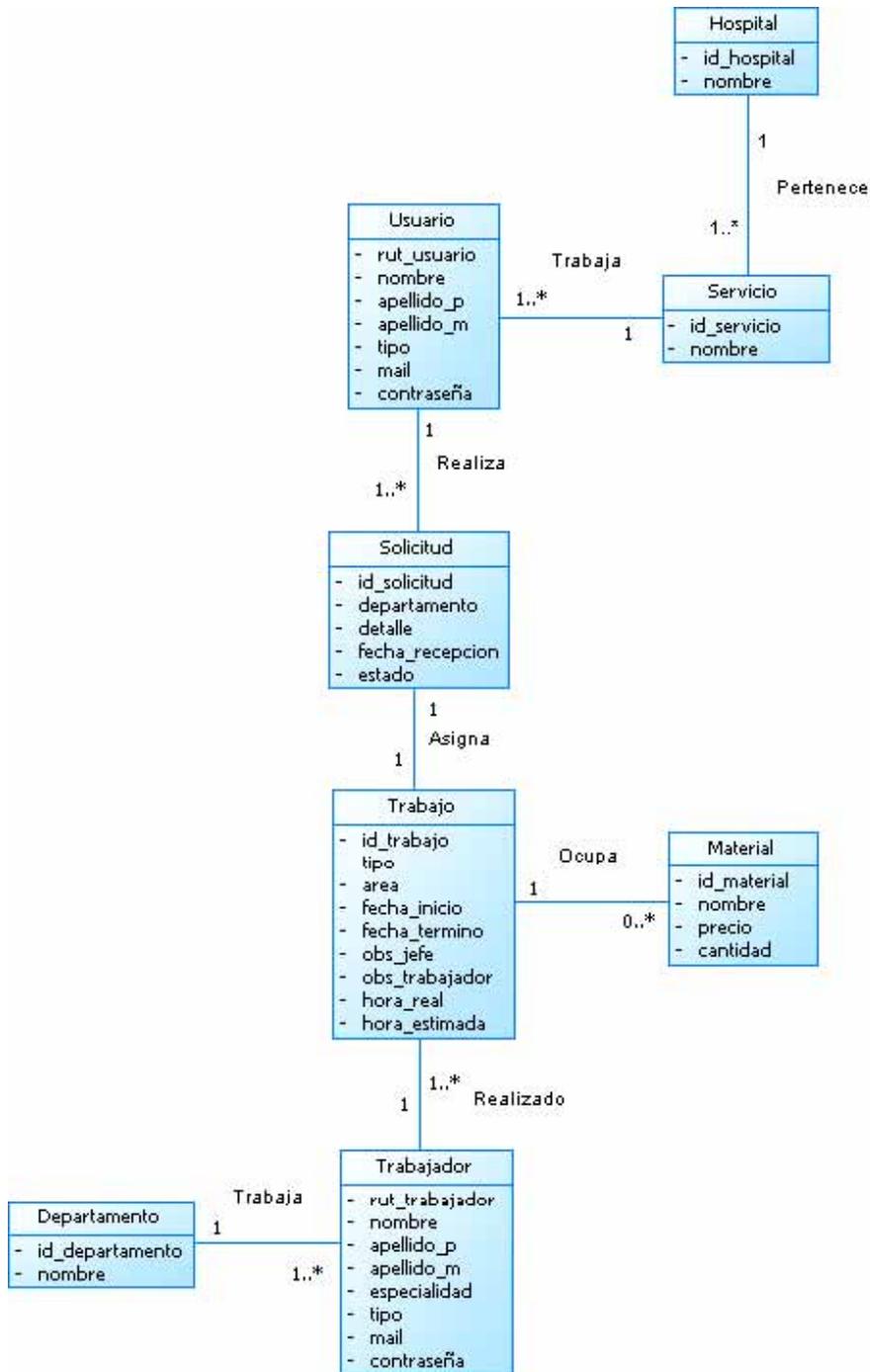
Diagrama de Secuencia

Figura 47: Diagrama de secuencia Finalizar Trabajo.



Modelo del Dominio del módulo

Figura 48: Modelo del dominio Módulo Solicitudes.



Contratos de Operaciones

Tabla 54: Contrato de operaciones ingresarSolicitud.

--	--

Operación	ingresarSolicitud(datos solicitud)
Precondiciones	El usuario se ha identificado en el sistema y los datos han sido validados.
Descripción	Se realiza el ingreso de un nuevo usuario en el sistema.
Poscondiciones	<ul style="list-style-type: none"> - Se creó una nueva instancia de Solicitud. - Se creó una asociación entre Solicitud y Servicio Clínico correspondiente.

Tabla 55: Contrato de operaciones revisarEstadoSolicitud.

Operación	revisarEstadoSolicitud(rut_usuario)
Precondiciones	El empleado se ha identificado en el sistema.
Descripción	Se muestran todas las solicitudes asociadas a un usuario.
Poscondiciones	No hay.

Tabla 56: Contrato de operaciones elegirTrabajo.

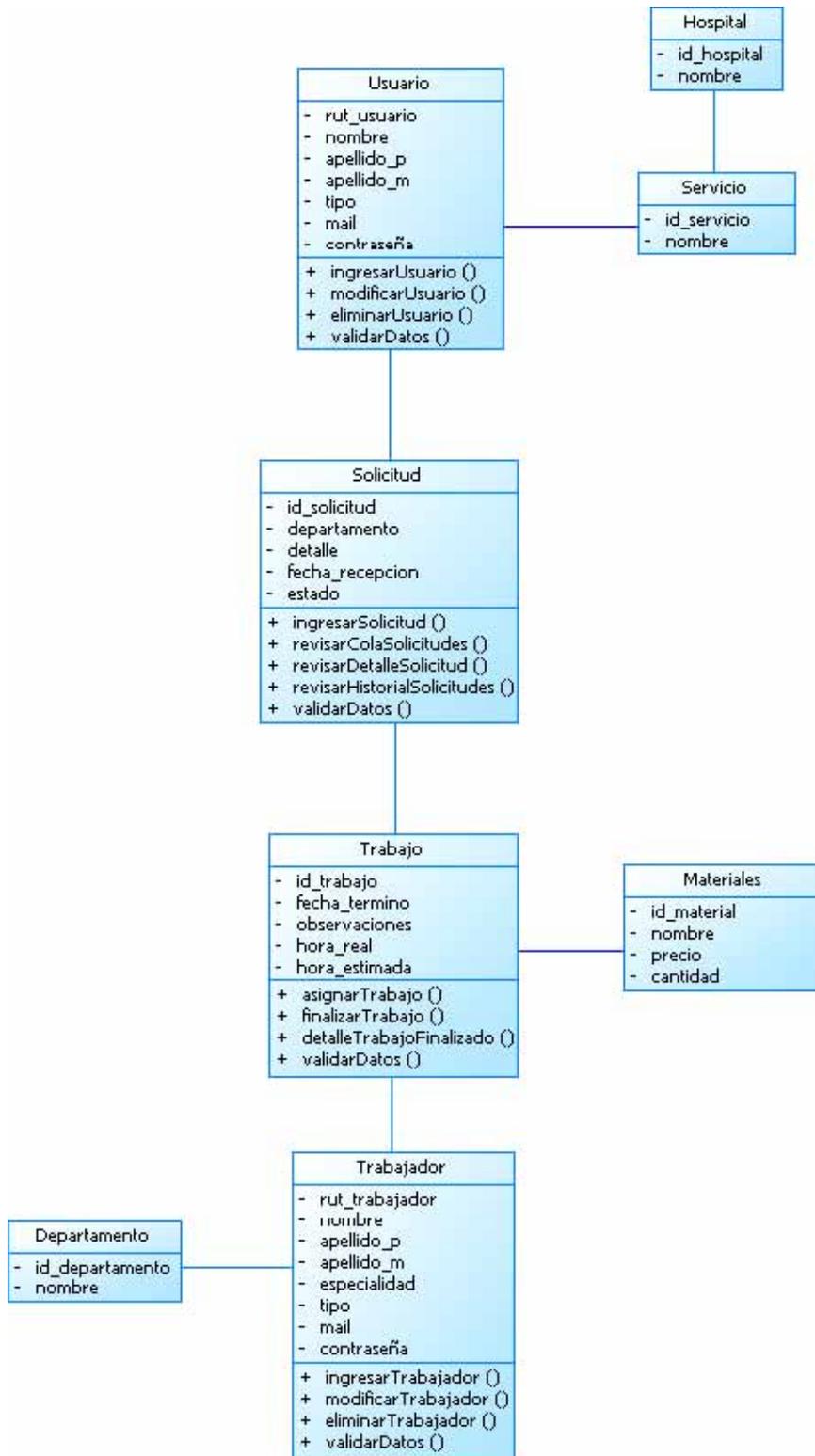
Operación	elegirTrabajo(id solicitud)
Precondiciones	El Trabajador se ha identificado en el sistema.
Descripción	Se realiza elige un trabajo a finalizar.
Poscondiciones	No hay.

Tabla 57: Contrato de operaciones finalizarTrabajo.

Operación	finalizarTrabajo(datos trabajo finalizado)
Precondiciones	La solicitud escogida existe y no ha sido finalizada.
Descripción	Se ingresan los datos del trabajo finalizado.
Poscondiciones	- Se creó una nueva instancia de trabajo finalizado. - Se modificó el estado de la solicitud a <i>finalizada</i> .

Modelo de Clases del módulo

Figura 49: Modelo de clases Módulo Solicitudes.



10.2.2. Iteración 5: Módulo Administración

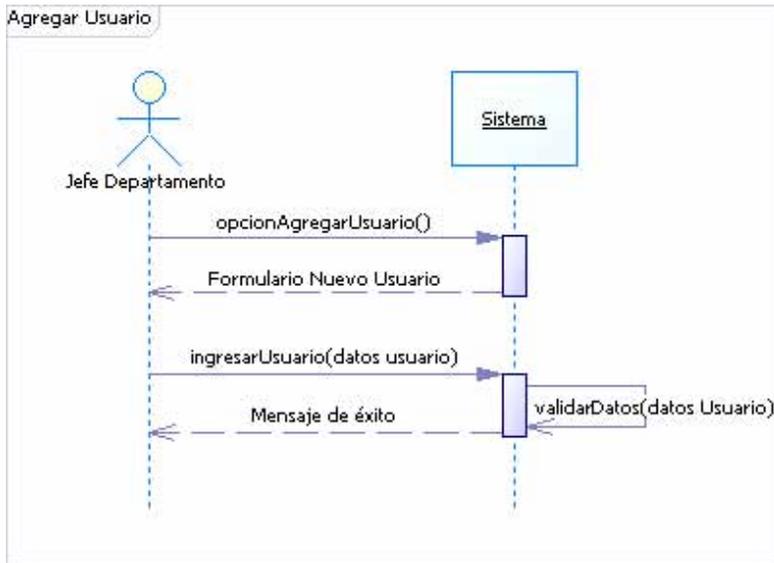
Especificación Caso de Uso Agregar Usuario

Nombre	Agregar Usuario.	
Actores	Jefe Departamento.	
Descripción	Este caso de uso describe el ingreso al sistema de un nuevo Usuario.	
Precondición	El Jefe Departamento se ha identificado en el sistema.	
Escenario Principal	Paso	Acción
	1	El Jefe Departamento selecciona la opción para agregar un nuevo Usuario al sistema.
	2	El sistema entrega la interfaz con el formulario de ingreso.
	3	El Jefe Departamento ingresa los datos del nuevo Usuario.
	4	El sistema valida la información.
Poscondición	El sistema almacena la información del nuevo Usuario.	
Excepciones	Paso	Acción
	3a	En caso de que el Jefe Departamento decida cancelar la operación, el sistema vuelve al inicio.
Frecuencia	Baja.	

Tabla 58: Especificación Caso de Uso Agregar Usuario.

Diagrama de Secuencia

Figura 50: Diagrama de secuencia Agregar Usuario.



Especificación Caso de Uso Modificar Usuario

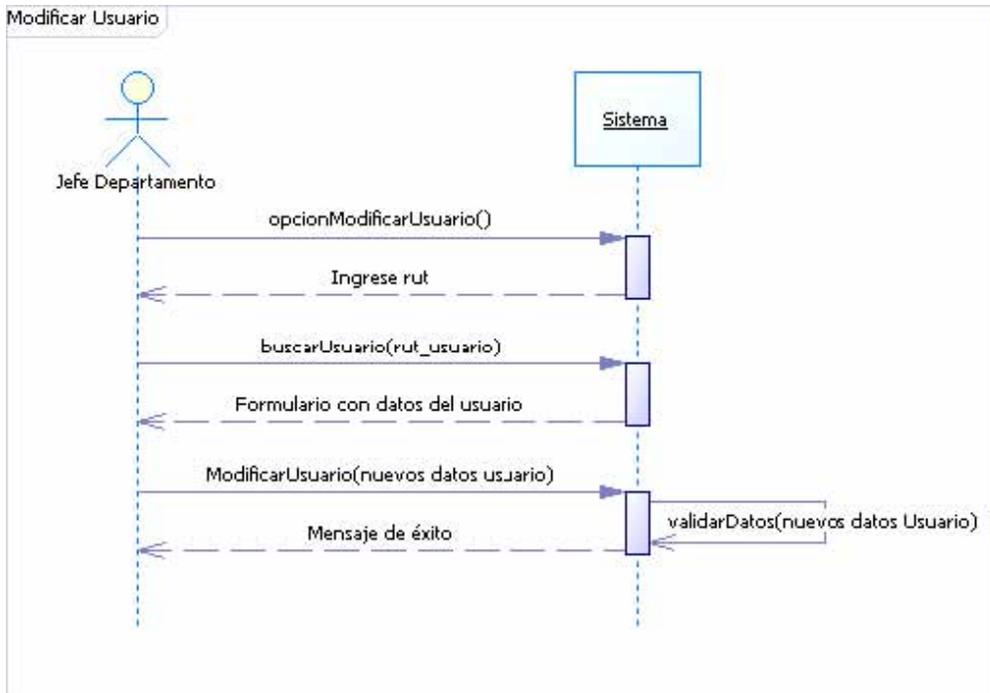
Nombre	Modificar Usuario.
Actores	Jefe Departamento.
Descripción	Este caso de uso describe la modificación de los datos de un Usuario.
Precondición	El Jefe de Departamento se ha identificado en el sistema.
Escenario	Paso Acción
Principal	1 El Jefe de Departamento selecciona la opción para modificar los datos de un Usuario.

	2	El sistema solicita el rut del Usuario al cual se le modificarán sus datos.
	3	El Jefe de Departamento ingresa el rut del Usuario.
	4	El sistema muestra la interfaz con los datos del Usuario.
	5	El Jefe de Departamento modifica los datos del Usuario.
	6	El sistema guarda los cambios y finaliza la operación.
Poscondición	El sistema almacenó la nueva información del Usuario.	
Excepciones	Paso	Acción
	3a	En caso de que el rut de Usuario no exista, el sistema muestra un mensaje indicando que no se encontró el Usuario.
	5a	En caso de que el Jefe Departamento decida no realizar modificaciones a los datos del Usuario, el sistema vuelve al inicio.
Frecuencia	Baja.	

Tabla 59: Especificación Caso de Uso Modificar Usuario.

Diagrama de Secuencia

Figura 51: Diagrama de secuencia Modificar Usuario.



Especificación Caso de Uso Eliminar Usuario.

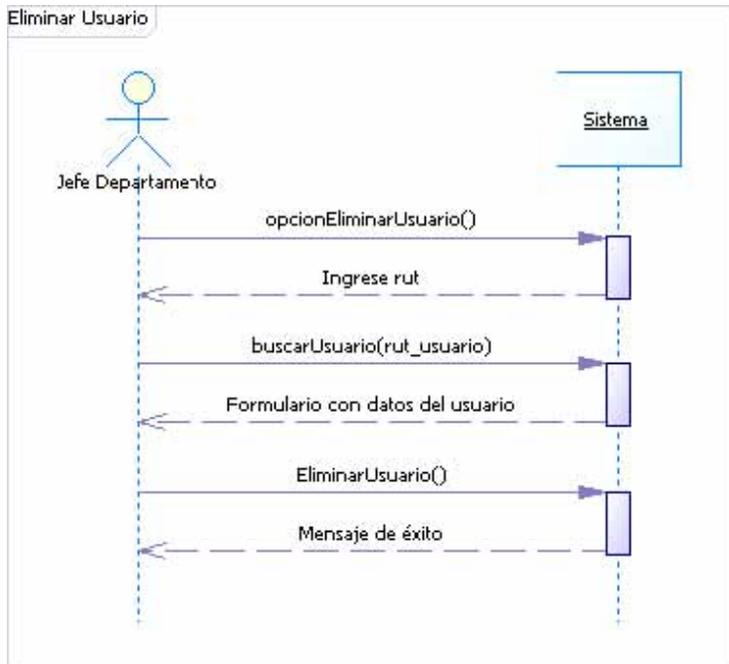
Nombre	Eliminar Usuario.						
Actores	Jefe Departamento.						
Descripción	Este caso de uso describe la eliminación un Usuario del sistema.						
Precondición	El Jefe de Departamento se ha identificado en el sistema.						
Escenario	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El Jefe de Departamento selecciona la opción para eliminar un Usuario del sistema.</td> </tr> <tr> <td>2</td> <td>El sistema solicita el rut del Usuario al cual desea eliminar.</td> </tr> </tbody> </table>	Paso	Acción	1	El Jefe de Departamento selecciona la opción para eliminar un Usuario del sistema.	2	El sistema solicita el rut del Usuario al cual desea eliminar.
Paso	Acción						
1	El Jefe de Departamento selecciona la opción para eliminar un Usuario del sistema.						
2	El sistema solicita el rut del Usuario al cual desea eliminar.						
Principal							

	3	El Jefe de Departamento ingresa el rut del Usuario.
	4	El sistema muestra la interfaz con los datos del Usuario y solicita confirmación.
	5	El Jefe de Departamento confirma la eliminación.
	6	El sistema muestra mensaje de éxito.
Poscondición	El sistema eliminó al Usuario del sistema.	
Excepciones	Paso	Acción
	3a	En caso de que el rut de Usuario no exista, el sistema muestra un mensaje indicando que no se encontró el Usuario.
	5a	En caso de que el Jefe Departamento decida no eliminar al Usuario del sistema, cancela la operación y vuelve al inicio.
Frecuencia	Baja.	

Tabla 60: Especificación Caso de Uso Eliminar Usuario.

Diagrama de Secuencia

Figura 52: Diagrama de secuencia Eliminar Usuario.



Especificación Caso de Uso Ingresar Catastro

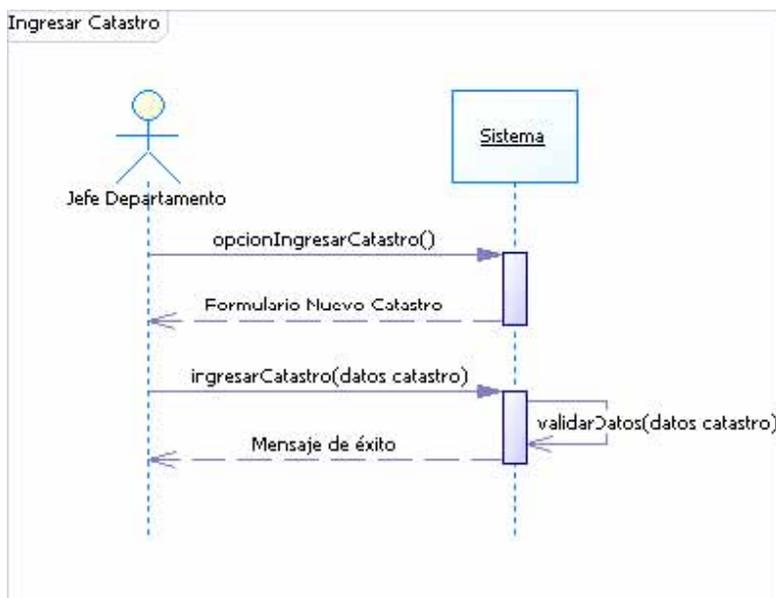
Nombre	Ingresar Catastro.						
Actores	Jefe Departamento.						
Descripción	Este caso de uso permite el ingreso de un catastro de un equipo informático.						
Precondición	El Jefe Departamento se ha identificado en el sistema.						
Escenario	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El Jefe Departamento selecciona la opción de ingresar catastro.</td> </tr> <tr> <td>2</td> <td>El sistema entrega la interfaz con el formulario de catastro.</td> </tr> </tbody> </table>	Paso	Acción	1	El Jefe Departamento selecciona la opción de ingresar catastro.	2	El sistema entrega la interfaz con el formulario de catastro.
Paso	Acción						
1	El Jefe Departamento selecciona la opción de ingresar catastro.						
2	El sistema entrega la interfaz con el formulario de catastro.						
Principal							

	3	El usuario ingresa todos los datos solicitados y los envía.
	4	El sistema valida la información.
Poscondición	El sistema almacenó el formulario.	
Excepciones	Paso	Acción
	3a	En caso de que el Jefe Departamento decida no cancelar la operación, el sistema vuelve al inicio.
Frecuencia	Baja.	

Tabla 61: Especificación Caso de Uso Ingresar Catastro.

Diagrama de Secuencia

Figura 53: Diagrama de secuencia Ingresar Catastro.



Especificación Caso de Uso Modificar Catastro

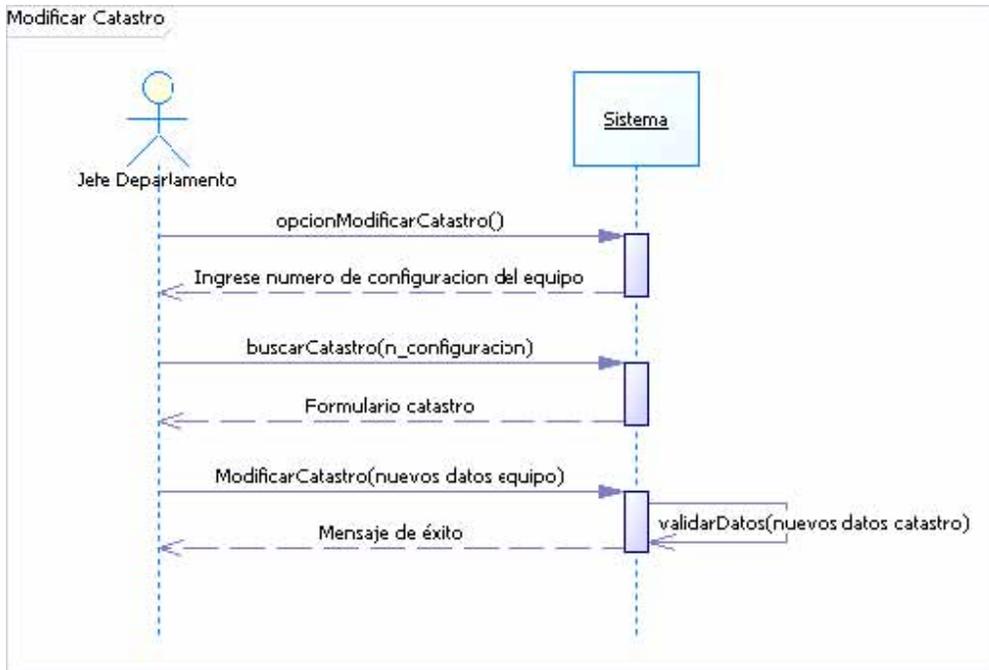
Nombre	Modificar Catastro.	
Actores	Jefe Departamento.	
Descripción	Este caso de uso describe la modificación de los datos del catastro de un equipo informático.	
Precondición	El Jefe de Departamento se ha identificado en el sistema.	
Escenario Principal	Paso	Acción
	1	El Jefe de Departamento selecciona la opción para modificar catastro.
	2	El sistema solicita el número de configuración del equipo al cual se modificarán sus datos.
	3	El Jefe de Departamento ingresa el número de configuración del equipo.
	4	El sistema muestra la interfaz con el detalle del catastro.
	5	El Jefe de Departamento modifica los datos del catastro.
	6	El sistema guarda los cambios y finaliza la operación.
Poscondición	El sistema almacenó la nueva información de catastro.	
Excepciones	Paso	Acción
	3a	En caso de que el número de configuración no

		exista, el sistema muestra un mensaje indicando que no se encontró el equipo.
	5a	En caso de que el Jefe Departamento decida no realizar modificaciones al catastro del equipo, el sistema vuelve al inicio.
Frecuencia	Baja.	

Tabla 62: Especificación Caso de Uso Modificar Catastro.

Diagrama de Secuencia

Figura 54: Diagrama de secuencia Modificar Catastro.



Especificación Caso de Uso Eliminar Catastro

Nombre	Eliminar Catastro.

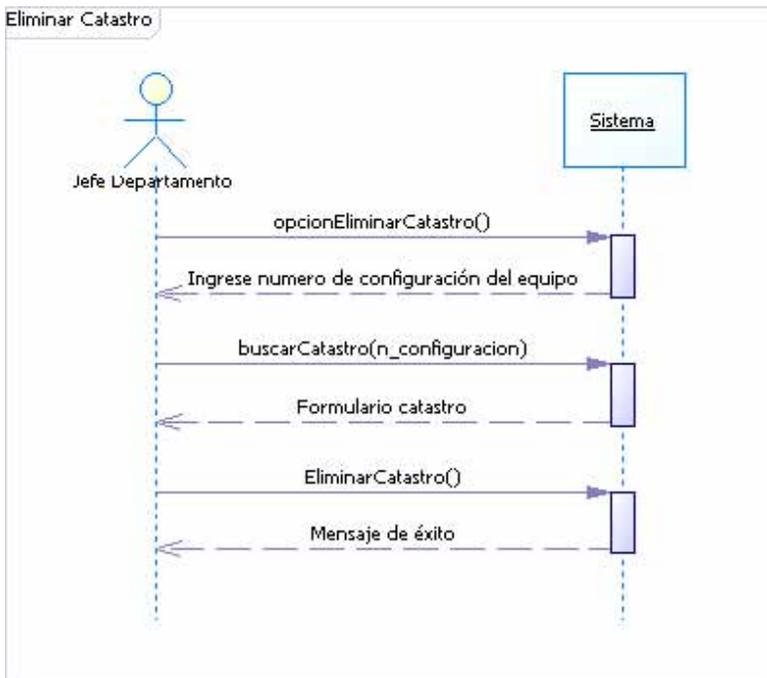
Actores	Jefe Departamento.	
Descripción	Este caso de uso describe la eliminación un catastro del sistema.	
Precondición	El Jefe de Departamento se ha identificado en el sistema.	
Escenario Principal	Paso	Acción
	1	El Jefe de Departamento selecciona la opción para eliminar un catastro del sistema.
	2	El sistema solicita el número de configuración del equipo que se desea eliminar.
	3	El Jefe de Departamento ingresa el número de configuración del equipo.
	4	El sistema muestra la interfaz con el detalle del catastro y solicita confirmación.
	5	El Jefe de Departamento confirma la eliminación.
	6	El sistema muestra mensaje de éxito.
Poscondición	El sistema eliminó el catastro del sistema.	
Excepciones	Paso	Acción
	3a	En caso de que el número de configuración de equipo no exista, el sistema muestra un mensaje indicando que no se encontró el empleado.
	5a	En caso de que el Jefe Departamento decida no eliminar el catastro del sistema, cancela la

		operación y vuelve al inicio.
Frecuencia	Baja.	

Tabla 63: Especificación Caso de Uso Eliminar Catastro.

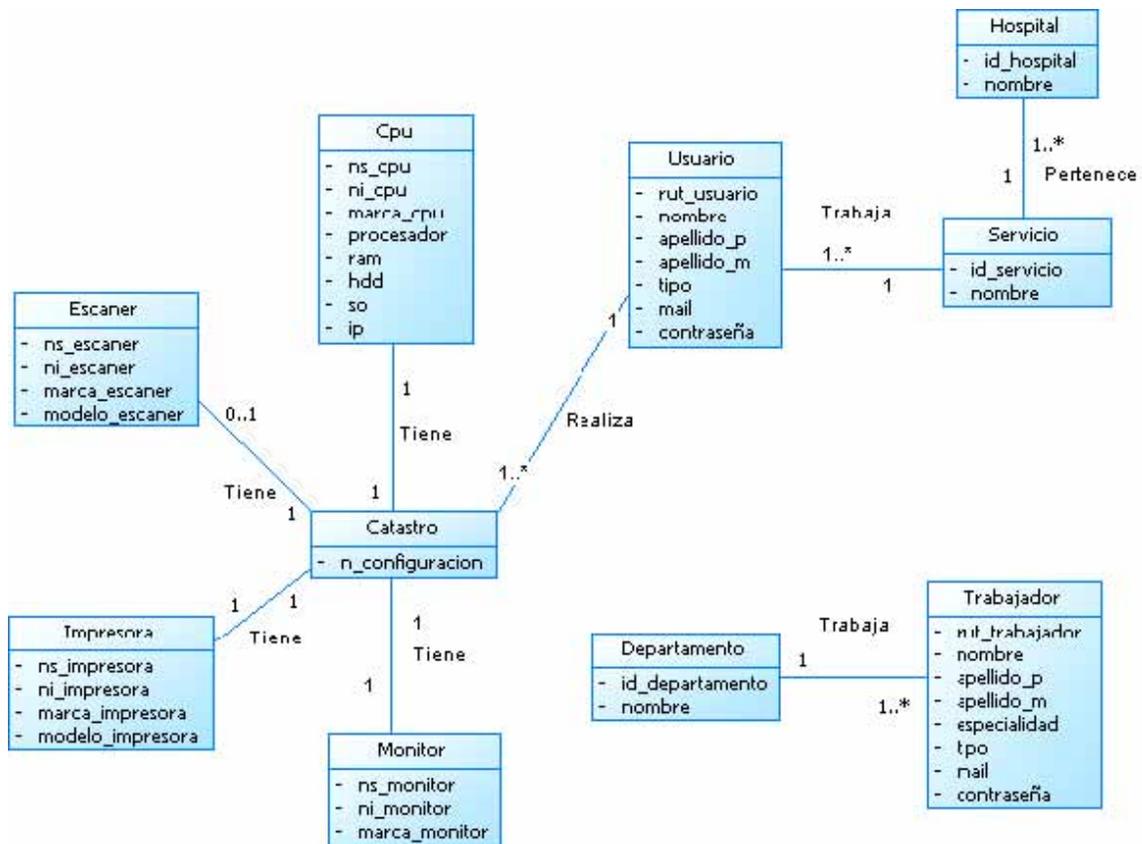
Diagrama de Secuencia

Figura 55: Diagrama de secuencia Eliminar Catastro.



Modelo del Dominio del módulo

Figura 56: Modelo del dominio Módulo Administración.



Contratos de Operaciones

Tabla 64: Contrato de operaciones ingresarUsuario.

Operación	ingresarUsuario (datos usuario)
Precondiciones	El Jefe de Departamento se ha identificado en el sistema y los datos han sido validados.
Descripción	Se realiza el ingreso de un nuevo usuario en el sistema.
Poscondiciones	Se creó una nueva instancia de usuario.

Tabla 65: Contrato de operaciones modificarUsuario.

Operación	modificarUsuario(nuevos datos usuario)
Precondiciones	El Jefe de Departamento se ha identificado en el sistema.
Descripción	Se modifica la información asociada a un usuario.
Poscondiciones	Se modificaron los campos de la instancia usuario.

Tabla 66: Contrato de operaciones buscarUsuario.

Operación	buscarUsuario(rut_usuario)
Precondiciones	El Jefe de Departamento se ha identificado en el sistema.
Descripción	Se realiza una búsqueda de un usuario.
Poscondiciones	No hay.

Tabla 67: Contrato de operaciones eliminarUsuario.

Operación	eliminarUsuario ()
Precondiciones	El Jefe de Departamento se ha identificado en el sistema.
Descripción	Se elimina un usuario del sistema.
Poscondiciones	<ul style="list-style-type: none"> - Se elimina la instancia del Usuario. - Se elimina la asociación existente entre las instancias de Usuario

	y Servicio (en caso de ser un Empleado o Director).
--	---

Tabla 68: Contrato de operaciones solicitudesPendientes.

Operación	solicitudesPendientes()
Precondiciones	El Jefe de Departamento se ha identificado en el sistema.
Descripción	Se muestran las solicitudes pendientes por asignar.
Poscondiciones	No hay.

Tabla 69: Contrato de operaciones historialUsuario.

Operación	historialUsuario(rut_usuario)
Precondiciones	El Jefe de Departamento se ha identificado en el sistema, el usuario existe en el sistema y tiene asociada al menos una solicitud.
Descripción	Se muestran las solicitudes realizadas por el usuario.
Poscondiciones	No hay.

Tabla 70: Contrato de operaciones escogerSolicitud.

Operación	escogerSolicitud(id solicitud)
Precondiciones	Hay solicitudes pendientes por asignar.

Descripción	Se escoge una solicitud para ser asignada a un grupo de trabajo.
Poscondiciones	No hay.

Tabla 71: Contrato de operaciones asignarSolicitud.

Operación	asignarSolicitud(datos asignación solicitud)
Precondiciones	Se escogió una solicitud para asignar.
Descripción	Se asigna una solicitud a un trabajador o grupo de trabajo.
Poscondiciones	Se creó una asociación entre Solicitud y Trabajador.

Tabla 72: Contrato de operaciones ingresarCatastro.

Operación	ingresarCatastro(datos catastro)
Precondiciones	El Jefe de Departamento se ha identificado en el sistema y los datos han sido validados.
Descripción	Se ingresa un nuevo catastro.
Poscondiciones	<ul style="list-style-type: none"> - Se creó una nueva instancia de Catastro. - Se creó una asociación entre Catastro y Empleado Servicio.

Tabla 73: Contrato de operaciones modificarCatastro.

--	--

Operación	modificarCatastro(nuevos datos catastro)
Precondiciones	El Jefe de Departamento se ha identificado en el sistema.
Descripción	Se modifica la información asociada a un catastro.
Poscondiciones	Se modificaron los campos de la instancia catastro.

Tabla 74: Contrato de operaciones buscarCatastro.

Operación	buscarCatastro(n_configuracion)
Precondiciones	El Jefe de Departamento se ha identificado en el sistema.
Descripción	Se realiza una búsqueda de un catastro.
Poscondiciones	No hay.

Tabla 75: Contrato de operaciones eliminarCatastro.

Operación	eliminarCatastro()
Precondiciones	El Jefe de Departamento se ha identificado en el sistema.
Descripción	Se elimina un catastro del sistema.
Poscondiciones	<ul style="list-style-type: none"> - Se elimina la instancia del Catastro. - Se elimina la asociación existente entre las instancias de Catastro y Empleado Servicio.

Tabla 76: Contrato de operaciones ingresarMateriales.

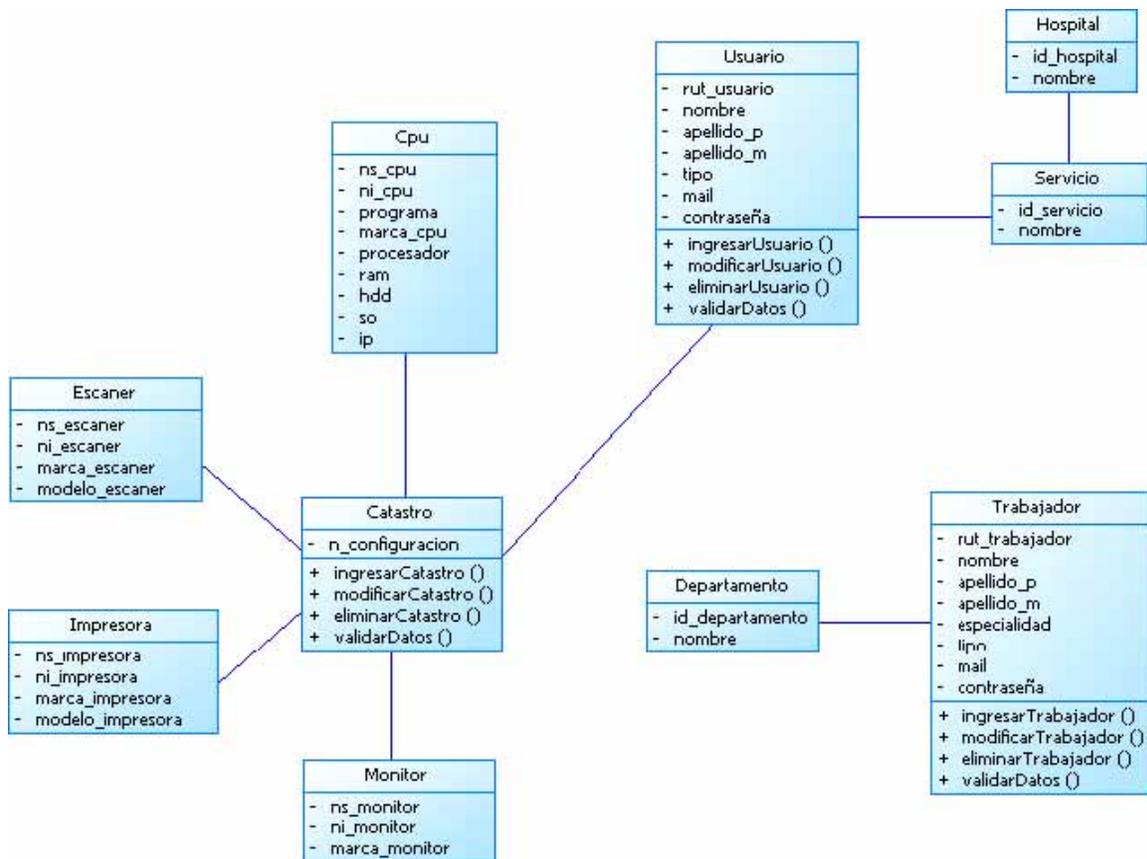
Operación	ingresarMateriales(datos materiales)
Precondiciones	El Jefe de Departamento/Trabajador se ha identificado en el sistema y los datos han sido validados.
Descripción	Se realiza el ingreso de un nuevo material en el sistema.
Poscondiciones	- Se creó una nueva instancia de Material. - Se creó una asociación entre Material y Trabajo.

Tabla 77: Contrato de operaciones validarDatos.

Operación	validarDatos()
Precondiciones	No hay.
Descripción	Se comprueba que los datos ingresados son válidos.
Poscondiciones	No hay.

Modelo de Clases del módulo

Figura 57: Modelo de clases Módulo Administración.



10.2.3. Iteración 6: Módulo Informes

Especificación Caso de Uso Generar Informes

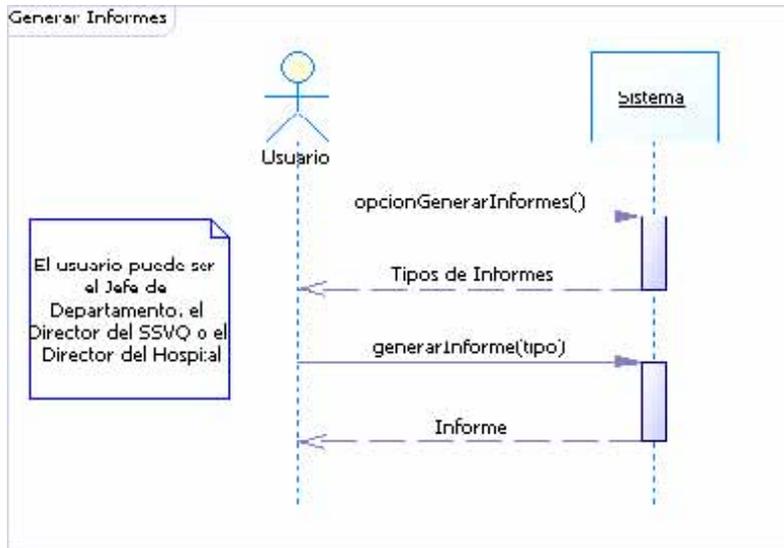
Nombre	Generar Informes.				
Actores	Jefe Departamento.				
Descripción	Este caso de uso permite la generación de informes estadísticos.				
Precondición	El Jefe Departamento se ha identificado en el sistema.				
Escenario	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> </tr> </tbody> </table>	Paso	Acción		
Paso	Acción				

Principal	1	El Jefe Departamento ingresa a la opción de generación de informes.
	2	El sistema entrega la lista de los distintos tipos de informes generables.
	3	El Jefe Departamento escoge el tipo que necesita.
	4	El sistema muestra la interfaz con el informe solicitado.
Poscondición	No hay.	
Excepciones	Paso	Acción
	3a	En caso de que el Jefe Departamento decida volver, el sistema vuelve al inicio.
Frecuencia	Media.	

Tabla 78: Especificación Caso de Uso Generar Informes.

Diagrama de Secuencia

Figura 58: Diagrama de secuencia Generar Informes.



Modelo del Dominio del módulo

Figura 59: Modelo del dominio Módulo Informes.



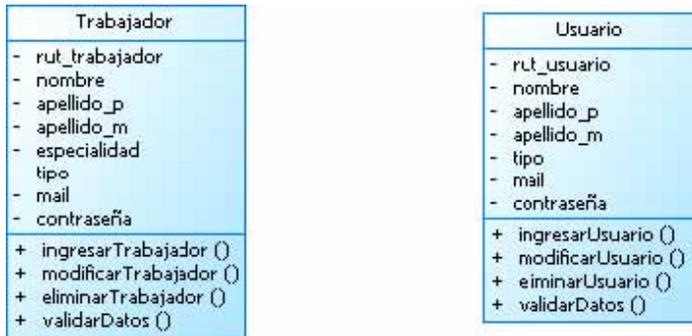
Contratos de Operaciones

Operación	generarInforme(tipo)
Precondiciones	El Jefe de Departamento se ha identificado en el sistema y escogió un tipo de informe.
Descripción	Se muestra el tipo de informe requerido por el Jefe de Departamento.
Poscondiciones	No hay.

Tabla 79: Contrato de operaciones generarInforme.

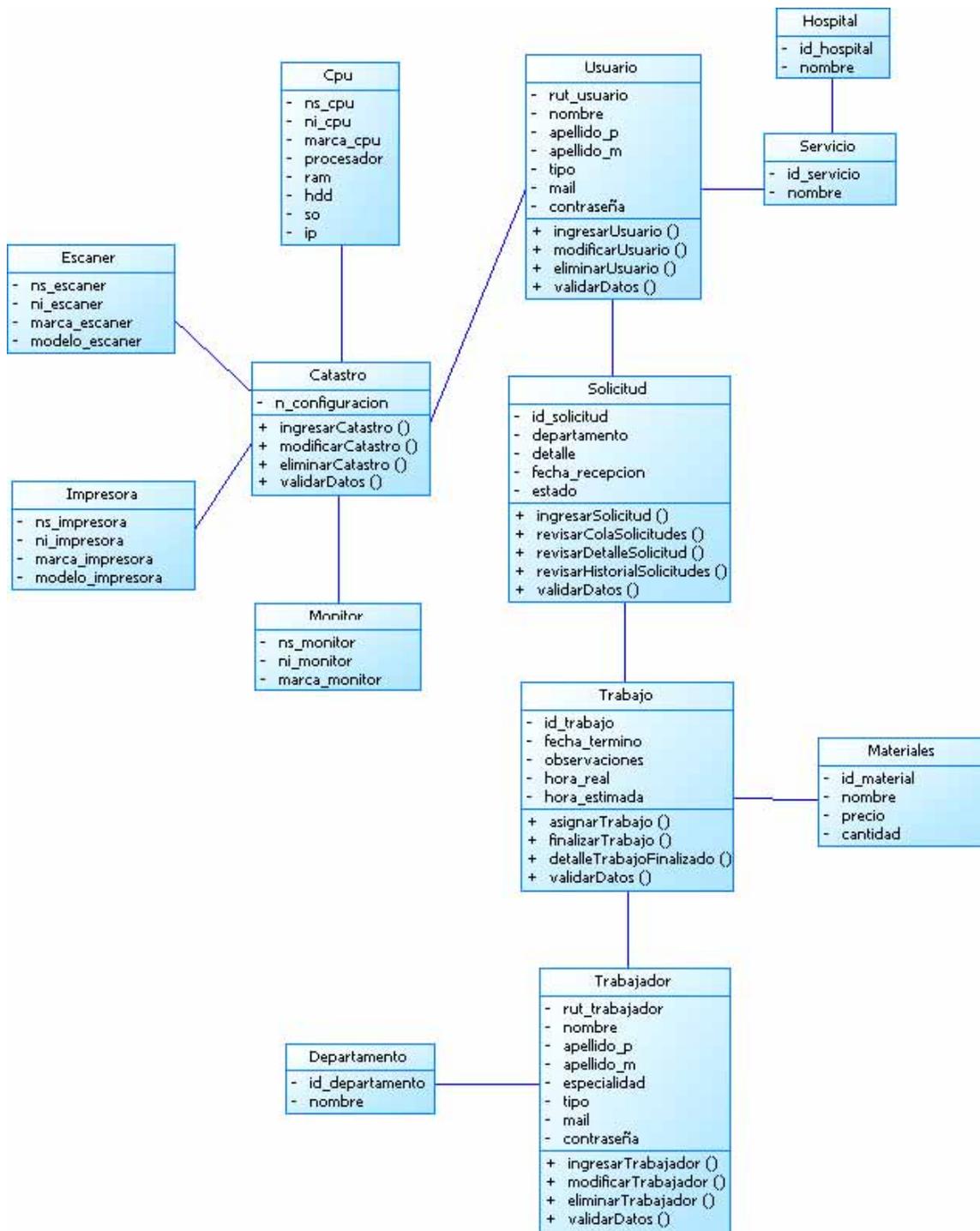
Modelo de Clases del módulo

Figura 60: Modelo de clases del Módulo Informes.



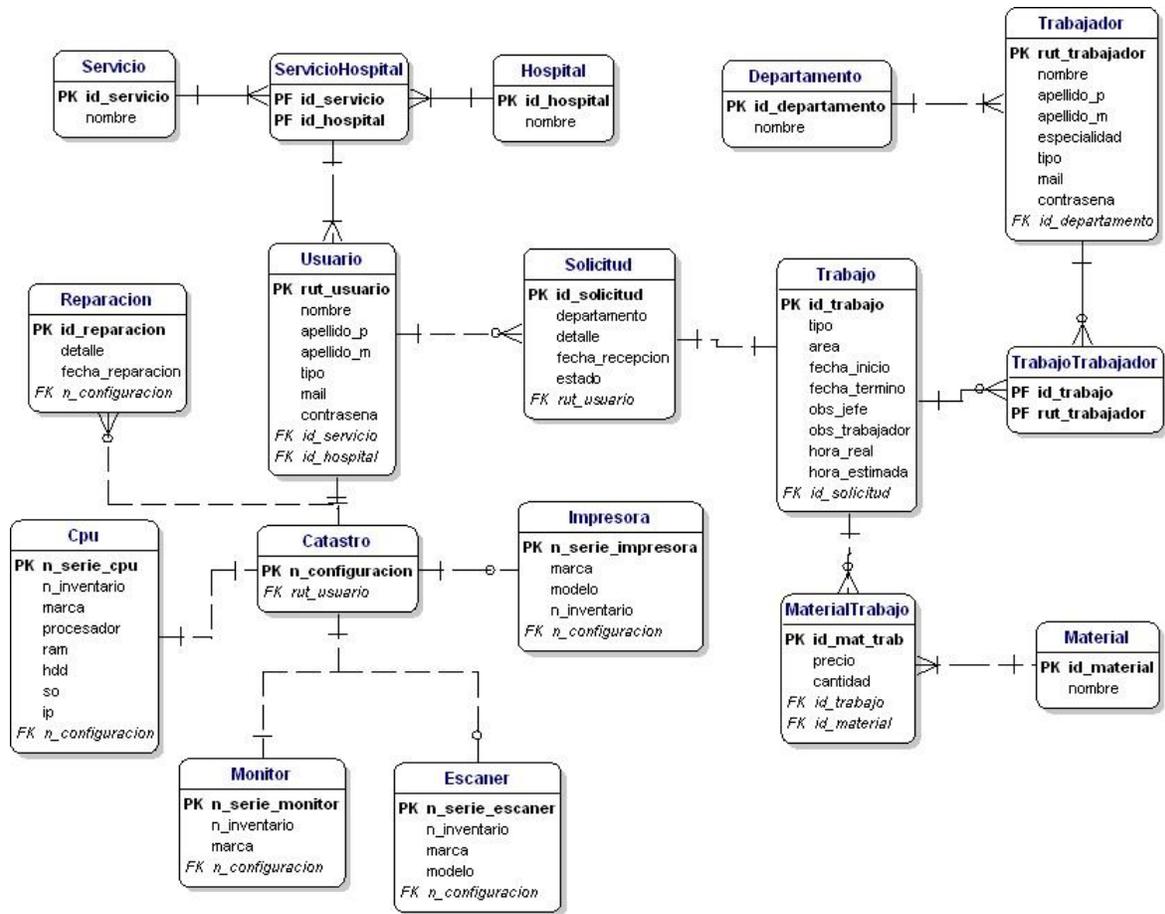
Modelo de Clases Integrado del Sistema

Figura 61: Modelo de clases integrado Sistema Solicitudes de Atención.



Modelo de Datos integrado del sistema

Figura 62: Modelo de datos integrado Sistema Solicitudes de Atención.



11. Interfaces de los Sistemas

11.1 Sistema OIRS

Ingreso de RFS

La figura 63 corresponde a la interfaz del formulario para el ingreso de una RFS. En ella, el Jefe de OIRS traspasa la información recogida en los libros y la ingresa al sistema.

Figura 63: Formulario ingreso RFS.

Oficina de Informaciones Reclamos y Sugerencias
Servicio de Salud Viña del Mar - Quillota
en red servimos a más...

Jorge Villar
Jefe OIRS
Cerrar Sesión

Lunes 19 de Noviembre de 2007

RFS
Ingresar RFS
Derivar RFS
Estado RFS
Administración
Usuario
Historial Paciente
Buscar
Mostrar Usuarios
Modificar Contraseña
Informes
Por Listado
Por Resumen
Por Tiempos
Por Totales

Ingresar RFS

Datos del Paciente
Rut: [] - []
Nombre: []
Apellido Paterno: [] Apellido Materno: []
Dirección: [] Comuna: ---Seleccione---
Teléfono: []
Afectado: []

Datos de la RFS
Nro: [] Tipo: ---Seleccione--- Fecha: []
Libro: ---Seleccione--- Folio: []
Servicio: ---Seleccione--- Calidad: ---Seleccione--- Estamento: ---Seleccione---
Descripción:
Restan 5000 caracteres

Guardar Cancelar

Creación y Diseño : Luis Astete D - Pablo Maldonado B
©2007

Derivar RFS

La figura 64, muestra la interfaz donde el Jefe de la OIRS revisa las RFS realizadas por los usuarios y derivarlas a los servicios clínicos correspondientes, para su posterior respuesta.

Figura 64: Derivar RFS.

Oficina de Informaciones Reclamos y Sugerencias
Servicio de Salud Viña del Mar - Quillota
en red servimos a más...

Jorge Villar
Jefe OIRS
Cerrar Sesión

Lunes 19 de Noviembre de 2007

Derivar RFS

Nº	Tipo	Fecha Ingreso	Calidad	Estamento	Servicio	Derivar
N/A	Reclamo	15-11-2007	Asistencial	Administrativo	Admisión	
234	Reclamo	22-10-2007	AUGE	Administrativo	Acc. Escolar	

[Derivar Todas](#)

Creación y Diseño : Luis Astete D - Pablo Maldonado B
©2007

Responder RFS

La figura 65 corresponde a la interfaz donde se muestran las RFS que fueron derivadas al servicio correspondiente y que esperan ser respondidas.

Figura 65: Responder RFS.

Menu

- Responder RFS
- RFS Criticas
- RFS Vencidas
- Modificar Contraseña

Responder RFS

Nº	Tipo	Fecha Derivada	Calidad	Estamento	Responder
234	Reclamo	19-11-2007	AUGE	Administrativo	✓

Luego de escoger una RFS para responder, el sistema muestra el detalle de ésta y le permite al Jefe de Servicio ingresar una respuesta. La interfaz correspondiente se muestra en la figura 66.

Figura 66: Ingresar Respuesta.

Menu

- Responder RFS
- RFS Críticas
- RFS Vencidas
- Modificar Contraseña

Ingresar Respuesta

Datos de la RFS

Nº RFS: 234
Tipo: Reclamo
Calidad: AUGE

Fecha Derivada: 19-11-2007
Estamento: Administrativo

Descripción:

Mala atención por parte del personal

Respuesta:

Restan 1000 caracteres

Informes por Listado

Esta opción permite ver la totalidad de RFS recibidas, según un criterio determinado. En este caso, las RFS fueron filtradas por tipo felicitación. La figura 67 muestra la interfaz de esta opción.

Figura 67: Informes por Listado.

RFS

- Ingresar RFS
- Derivar RFS
- Estado RFS

Administración

- Usuario
- Historial Paciente
- Buscar
- Mostrar Usuarios
- Modificar Contraseña

Informes

- Por Listado
- Por Resumen
- Por Tiempos
- Por Totales

Listado RFS

Nº	Tipo	Fecha Ingreso	Calidad	Estamento	Servicio	Detalle
N/A	Felicitación	02-11-2007	Asistencial	Enfermería	Auge	
N/A	Felicitación	29-10-2007	Comp. Técnica	Médico	Cirugía	

 **Exportar a Excel**

Informes por Resumen

Esta opción permite saber la cantidad mensual de RFS recibidas en un servicio para cada calidad, o de una calidad determinada para todos los servicios del hospital. La figura 68 muestra la interfaz de esta opción.

Figura 68: Informes por Resumen.

RFS

Ingresar RFS

Derivar RFS

Estado RFS

Administración

Usuario

Historial Paciente

Buscar

Mostrar Usuarios

Modificar Contraseña

Informes

Por Listado

Por Resumen

Por Tiempos

Por Totales

Resumen Mensual por Servicio

Calidad	Ene	Feb	Mar	Abr	May	Jun	Jul	Ago	Sep	Oct	Nov	Dic
Asistencial	0	0	0	0	0	0	0	0	0	1	1	1
Auge	0	0	0	0	0	0	0	0	0	0	0	0
Comp. Técnica	0	0	0	0	0	0	0	0	0	0	0	0
Información	0	0	0	0	0	0	0	0	0	0	0	0
Infraestructura	0	0	0	0	0	0	0	0	0	0	0	0
Nulo	0	0	0	0	0	0	0	0	0	0	0	0
Organizacional	0	0	0	0	0	0	0	0	0	0	0	0
Probidad Admin.	0	0	0	0	0	0	0	0	0	0	0	0
Proced. Admin.	0	0	0	0	0	0	0	0	0	0	0	0
Tiempo Espera	0	0	0	0	0	0	0	0	0	0	0	0
Trato	0	0	0	0	0	0	0	0	0	0	0	0

[Exportar a Excel](#)

Informes por Tiempos

Esta opción permite saber los tiempos de respuesta en días desde que una RFS es derivada hasta que es respondida por el Jefe de Servicio. La figura 69 muestra la interfaz de esta opción.

Figura 69: Informes por Tiempos.

RFS

- Ingresar RFS
- Derivar RFS
- Estado RFS

Administración

- Usuario
- Historial Paciente
- Buscar
- Mostrar Usuarios
- Modificar Contraseña

Informes

- Por Listado
- Por Resumen
- Por Tiempos
- Por Totales

Tiempos de Respuesta Servicio

Nº	Servicio	Fecha Ingreso	Fecha Derivación	Fecha Respuesta	Tiempo Respuesta (Días)	Detalle
123	Admisión	24-12-2007	11-10-2007	15-11-2007	26	
3344	Admisión	01-10-2007	16-10-2007	11-10-2007	0	
3344	Admisión	01-10-2007	16-10-2007	30-10-2007	10	

Informes por Totales

Esta opción permite saber la cantidad total de RFS recibidas por un servicio, calidad o estamento, dentro de un periodo determinado. Además, el sistema permitirá al usuario ver los resultados de forma gráfica para una mayor claridad. La figura 70 muestra la interfaz de esta opción.

Figura 70: Informes por Totales.

RFS

Ingresar RFS

Derivar RFS

Estado RFS

Administración

Usuario

Historial Paciente

Buscar

Mostrar Usuarios

Modificar Contraseña

Informes

Por Listado

Por Resumen

Por Tiempos

Por Totales

RFS según Estamento

Estamento	Cantidad	%
Administrativo	1	25
Aux. Servicio	1	25
Enfermería	0	0
Guardias	0	0
Matrona	1	25
Médico	0	0
Otros	1	25
Paramédico	0	0
Profesional	0	0
S/Esp. Estamento	0	0
Total	4	

Página 1 De 2 1 2 > >|

[Exportar a Excel](#)

[Ver Grafico](#)

11.2 Sistema Solicitudes de Atención

Solicitudes Pendientes

La figura 71 muestra la interfaz donde el Jefe de Departamento puede revisar las solicitudes pendientes realizadas por los empleados de los diversos servicios clínicos de la red de hospitales, para luego asignarlas a algún trabajador o grupo de trabajo para su realización.

Figura 71: Solicitudes Pendientes.



Sistema de Solicitudes de Atención
 Servicio de Salud Viña del Mar - Quillota
en red servimos a más...

Jorge Villar
 Jefe Departamento
 Informática
 Cerrar Sesión

Lunes 19 de Noviembre de 2007

Solicitudes

Pendientes

Finalizadas

Administración

⊕ Usuario

⊕ Catastro

⊕ Buscar

Informes

Por Listado

Por Resumen

Por Totales

Solicitudes Pendientes

Nº	Fecha Ingreso	Descripción	Asignar
1	15-11-2007	no prende el computador	

Creación y Diseño : Luis Astete D - Pablo Maldonado B
 ©2007

Asignar Trabajo

En la figura 72, se muestra la interfaz que presentará el sistema luego de haber revisado las solicitudes pendientes y haber seleccionado la opción para asignar alguna de ellas. Aquí, el Jefe de Departamento deberá elegir el(los) encargado(s) para realizar la tarea, quien posteriormente será el responsable de la finalización de esta.

Figura 72: Asignar Trabajo.

Solicitudes
Pendientes
Finalizadas

Administración
 Usuario
 Catastro
 Buscar

Informes
Por Listado
Por Resumen
Por Totales

Asignar Trabajo

Área:

Tipo: Interno Externo

Personal Disponible		Personal Asignado
Juan Perez Brian Gomez	<input type="button" value=">"/> <input type="button" value=">>"/> <input type="button" value="<"/> <input type="button" value="<<"/>	

Para mover haga click en el nombre y luego en los botones centrales o doble click sobre el trabajador

Tiempo Estimado: hrs.

Observaciones:

Restan caracteres

Finalizar Trabajo

La figura 73, corresponde al proceso de finalización de un trabajo por parte de uno de los asignados a dicho trabajo. En ella, se presenta el detalle del trabajo a finalizar y debe ser completada con los campos restantes (tiempo real y observaciones) antes de ser dada por finalizada.

Figura 73: Finalizar Trabajo.

Solicitudes
Trabajos por Finalizar
Administración
Cambiar Contraseña

Finalizar Trabajo

Detalle Solicitante
Nombre: Sebastian Henriquez
Servicio: Admisión
Hospital: Adriana Cousiño, de Quintero

Detalle Solicitud
Fecha Inicio: 15-11-2007
Área: Hardware
Tiempo Estimado: 1 hrs.

Detalle Trabajo
Cant. materiales:
Insumo:
Cantidad:
Precio:
Tiempo Real: hrs.

Observaciones:

Restan caracteres

Posterior a esto, el Jefe de Servicio podrá ver el detalle del trabajo finalizado, como se ve en la figura 74.

Figura 74: Detalle Trabajo Finalizado.



Sistema de Solicitudes de Atención

Servicio de Salud Viña del Mar - Quillota

en red servimos a más...

Jorge Villar
Jefe Departamento
Informática
 Cerrar Sesión

Lunes 19 de Noviembre de 2007

Solicitudes

Pendientes

Finalizadas

Administración

Usuario

Catastro

Buscar

Informes

Por Listado

Por Resumen

Por Totales

Detalle Trabajo Finalizado

Detalle Solicitante

Nombre: Sebastian Henriquez

Servicio: Admisión

Hospital: Adriana Cousiño, de Quintero

Detalle Trabajo

Fecha Inicio: 15-11-2007

Fecha Termino: 15-11-2007

Tiempo Estimado: 1 hrs.

Tiempo Real: 1 hrs.

Área: Hardware

Insumos:

Nº	Nombre	Cantidad	Precio	Total
1	Fuente de Poder	1	15000	15000
			Total	15000

Observaciones:

se reemplazo la fuente de poder

[Volver](#)

13. Arquitectura

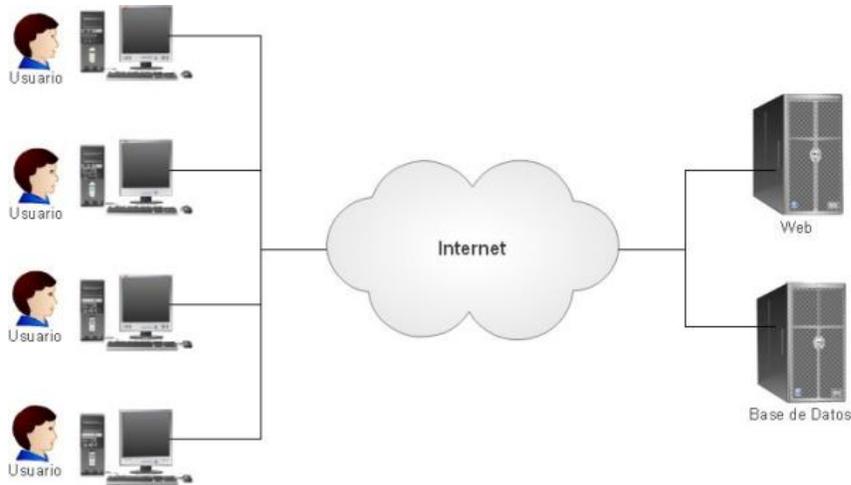
La arquitectura definida para ambos sistemas corresponde a la arquitectura de 3 capas. Esta consiste en dividir cada uno de los sistemas en una capa de presentación, una capa de negocio y una capa de datos. A continuación se explicará cada una de estas capas de la arquitectura.

- **Capa de Presentación:** representa lo que el usuario ve. A través de ésta, se le muestra la información al usuario y se realiza la captura de información ingresada por ésta al sistema.
- **Capa de Negocio:** en esta capa es donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos para almacenar o recuperar datos de él.
- **Capa de Datos:** es donde residen los datos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

De acuerdo a lo anterior, la arquitectura se puede observar desde un punto de vista lógico y uno físico, como lo muestran las figuras 75 y 76 respectivamente.

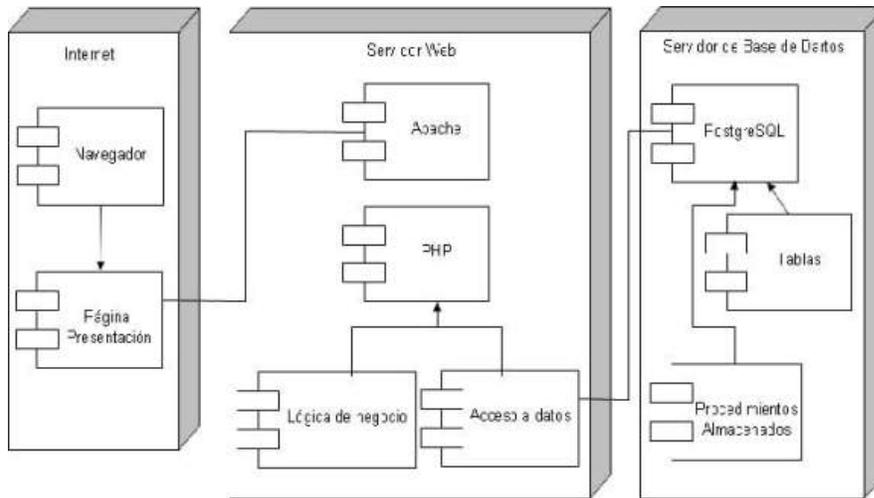
Vista Lógica

Figura 75: Vista Lógica.



Vista Física

Figura 76: Vista Física.



14. Plan de Construcción

La construcción de los sistemas se llevó a cabo en 6 iteraciones, 3 iteraciones por cada uno, correspondientes a cada módulo, de la misma manera en que se realizó el diseño de los mismos.

A continuación se detallará cada una de estas iteraciones.

a) Sistema OIRS

Iteración 1: Módulo RFS

- Ingresar RFS.
- Derivar RFS.
- Responder RFS.
- RFS Online.
- RFS en Proceso.
- RFS Críticas.
- RFS Vencidas.
- RFS Respondidas.

Iteración 2: Módulo Administración

- Crear usuario.
- Modificar usuario.
- Eliminar usuario.

- Buscar Usuario por Rut.
- Buscar Usuario por Apellido.
- Mostrar Usuarios.
- Historial Paciente.
- Buscar Paciente.
- Modificar Contraseña.

Iteración 3: Módulo Informes

- Por Listado.
- Por Resumen.
- Por Tiempos.
- Por Totales.

c) Sistema solicitudes de atención

Iteración 4: Módulo Solicitudes

- Ingresar solicitud.
- Solicitudes Pendientes.
- Historial Solicitudes.
- Solicitudes Finalizadas.
- Asignar trabajo.
- Finalizar trabajo.

Iteración 5: Módulo Administración

- Agregar Usuario.
- Modificar Usuario.
- Eliminar Usuario.
- Agregar catastro.
- Modificar catastro.
- Eliminar catastro.

Iteración 6: Módulo Informes

- Por Listado.
- Por Resumen.
- Por Totales.

15. Directrices de Pruebas

El diseño de un plan de pruebas resulta muy importante para poder verificar que lo que se construyó funcionaría correctamente y no tendría posibles errores una vez que sea puesto en producción. Para ello, se ha tomado en cuenta las llamadas Pruebas Unitarias [7], que tienen por objeto probar el código fuente y permitir una depuración posterior del mismo. Estas pruebas unitarias se pueden dividir en 2 tipos: Pruebas de Caja Blanca y Caja Negra.

a) Pruebas de Caja Blanca

Este tipo de pruebas permiten examinar la estructura interna del programa. Para ello se diseñan casos de prueba que examinen la lógica del programa. Es un método de diseño de casos de prueba que usa la estructura de control del diseño procedimental para derivar casos de prueba que garanticen que:

- Se ejercitan todos los caminos independientes de cada módulo.
- Se ejercitan todas las decisiones lógicas.
- Se ejecutan todos los bucles.
- Se ejecutan las estructuras de datos internas.

b) Pruebas de Caja Negra

Este tipo de pruebas se llevan a cabo sobre la interfaz del software, y es completamente indiferente al comportamiento interno y la estructura del programa. Estos casos de prueba pretenden demostrar que:

- Las funciones del software son operativas.
- La entrada se acepta en forma adecuada.
- Se produce una salida correcta.

- La integridad de la información externa se mantiene.

Se derivan conjuntos de condiciones de entrada que ejerciten completamente todos los requerimientos funcionales del programa. Las pruebas de caja negra intentan encontrar errores correspondientes a las siguientes categorías:

- Funciones incorrectas o que no existen.
- Errores de interfaz.
- Errores en estructuras de datos o en accesos a bases de datos externas.
- Errores de rendimiento.
- Errores de inicialización y de terminación.

Los casos de prueba deben satisfacer los siguientes criterios:

- Reducir, en un coeficiente que es mayor que uno, el número de casos de prueba adicionales.
- Que digan algo sobre la presencia o ausencia de clases de errores.

15.1. Selección de la directriz

De acuerdo a lo explicado anteriormente, los casos de pruebas diseñados fueron del tipo Caja Negra, los cuales se basaron en el ingreso de información al sistema, mediante la interfaz de usuario, y verificando que la información proporcionada sea la correcta.

En el siguiente punto serán detallados los casos de prueba de acuerdo a cada sistema.

15.1.1. Sistema OIRS

Tabla 80: Caso de prueba Ingresar RFS.

Caso de Prueba

Caso de Uso: Ingresar RFS
Descripción del caso de uso Permite el ingreso de una RFS por parte del usuario.
Descripción de la Prueba Se probará que los valores ingresados correspondan al tipo y se ingresen en el formato adecuado.
Entrada El usuario no ingresa ningún valor o estos no se encuentran dentro de las condiciones establecidas.
Resultado Esperado Un mensaje indicando que no se ingresó ningún valor o que los valores son incorrectos.

Tabla 81: Caso de prueba Responder RFS.

Caso de Prueba
Caso de Uso: Responder RFS
Descripción del caso de uso Permite el ingreso de la respuesta a la RFS hecha por el paciente.
Descripción de la Prueba Se probará que la respuesta haya sido ingresada.
Entrada

El usuario no ingresa ningún carácter o la cantidad de caracteres exceden a los permitidos.

Resultado Esperado

Un mensaje indicando que no se ingresó ningún carácter o que estos son más de los permitidos.

Tabla 82: Caso de prueba RFS Online.

Caso de Prueba
Caso de Uso: RFS Online
Descripción del caso de uso Permite la modificación de algunos de los valores de la RFS ingresada por el paciente vía online.
Descripción de la Prueba Se probará que los valores ingresados correspondan al tipo y se ingresen en el formato adecuado.
Entrada El usuario no ingresa ningún valor o estos no se encuentran dentro de las condiciones establecidas.
Resultado Esperado Un mensaje indicando que no se ingresó ningún valor o que los valores son incorrectos.

Tabla 83: Caso de prueba Crear Usuario.

Caso de Prueba
Caso de Uso: Crear Usuario
Descripción del caso de uso Permite crear un nuevo usuario en el sistema.
Descripción de la Prueba Se probará que los valores ingresados correspondan al tipo y se ingresen en el formato adecuado.
Entrada El usuario no ingresa ningún valor o estos no se encuentran dentro de las condiciones establecidas.
Resultado Esperado Un mensaje indicando que no se ingresó ningún valor o que los valores son incorrectos.

Tabla 84: Caso de prueba Modificar Usuario.

Caso de Prueba
Caso de Uso: Modificar Usuario
Descripción del caso de uso Permite modificar los datos asociados a un usuario del sistema.
Descripción de la Prueba

Se probará que los valores ingresados correspondan al tipo y se ingresen en el formato adecuado.

Entrada

El usuario no ingresa ningún valor o estos no se encuentran dentro de las condiciones establecidas.

Resultado Esperado

Un mensaje indicando que no se ingresó ningún valor o que los valores son incorrectos.

15.1.2. Sistema Solicitudes de Atención

Tabla 85: Caso de prueba Ingresar Solicitud.

Caso de Prueba
Caso de Uso: Ingresar Solicitud
Descripción del caso de uso Permite el ingreso de una solicitud de atención por parte del usuario.
Descripción de la Prueba Se probará que los valores ingresados correspondan al tipo y se ingresen en el formato adecuado.
Entrada El usuario no ingresa ningún valor o estos no se encuentran dentro de las condiciones establecidas.

<p>Resultado Esperado</p> <p>Un mensaje indicando que no se ingresó ningún valor o que los valores son incorrectos.</p>
--

Tabla 86: Caso de prueba Asignar Trabajo.

<p>Caso de Prueba</p>
<p>Caso de Uso: Asignar Trabajo</p>
<p>Descripción del caso de uso</p> <p>Permite la asignación de las solicitudes de atención en espera a un trabajador o grupo de trabajo.</p>
<p>Descripción de la Prueba</p> <p>Se probará que los valores ingresados correspondan al tipo, se ingresen en el formato adecuado y que se escoja al menos un trabajador para asignar el trabajo.</p>
<p>Entrada</p> <p>El usuario no ingresa ningún valor, no se encuentran dentro de las condiciones establecidas o no seleccionó a ningún trabajador para asignar el trabajo.</p>
<p>Resultado Esperado</p> <p>Un mensaje indicando que no se ingresó ningún valor, que los valores son incorrectos o que debe seleccionar al menos un trabajador para asignarle el trabajo.</p>

Tabla 87: Caso de prueba Finalizar Trabajo.

<p>Caso de Prueba</p>

Caso de Uso: Finalizar Trabajo
Descripción del caso de uso Permite a un trabajador dar por finalizado un trabajo y especificar los detalles de este
Descripción de la Prueba Se probará que los valores ingresados correspondan al tipo adecuado, que la solicitud exista y que su estado no sea <i>finalizado</i> .
Entrada El usuario no ingresa ningún valor, ingresa un valor incorrecto o una solicitud inexistente o que ya ha sido finalizada.
Resultado Esperado Un mensaje indicando que no se ingresó ningún valor, que el valor es incorrecto, que la solicitud no existe o ya ha sido finalizada.

Tabla 88: Caso de prueba Agregar Usuario.

Caso de Prueba
Caso de Uso: Agregar Usuario
Descripción del caso de uso Permite el ingreso de un nuevo usuario al sistema.
Descripción de la Prueba Se probará que los valores ingresados correspondan al tipo y se ingresen en el formato adecuado.

<p>Entrada</p> <p>El usuario no ingresa ningún valor o estos no se encuentran dentro de las condiciones establecidas.</p>
<p>Resultado Esperado</p> <p>Un mensaje indicando que no se ingresó ningún valor o que los valores son incorrectos.</p>

Tabla 89: Caso de prueba Ingresar Catastro.

<p>Caso de Prueba</p>
<p>Caso de Uso: Ingresar Catastro</p>
<p>Descripción del caso de uso</p> <p>Permite el ingreso de un nuevo catastro de equipo informático al sistema</p>
<p>Descripción de la Prueba</p> <p>Se probará que los valores ingresados correspondan al tipo y se ingresen en el formato adecuado</p>
<p>Entrada</p> <p>El usuario no ingresa ningún valor o estos no se encuentran dentro de las condiciones establecidas</p>
<p>Resultado Esperado</p> <p>Un mensaje indicando que no se ingresó ningún valor o que los valores son incorrectos.</p>

16. Conclusiones

El análisis de un proyecto informático resulta muy importante para el futuro desarrollo del mismo, puesto que en este se hacen estudios que se centran en la investigación del problema, la solución para este, el paradigma y la metodología que se utilizará en su desarrollo, y la factibilidad de llevarlo a cabo.

Durante la primera etapa del proyecto, la investigación del problema fue fundamental para entender el funcionamiento de la OIRS y del rol que cumplían los departamentos de soporte del SSVQ con los distintos establecimientos que pertenecen a su red hospitalaria, y de qué forma llevaban a cabo sus actividades actualmente.

Sin duda, un punto muy importante en este proceso para un equipo de desarrollo es la buena comunicación con el cliente, ya que permite entender la problemática y el funcionamiento de la entidad en cuestión. En este sentido, gracias a la buena disposición del cliente, el resultado obtenido fue una idea clara y completa, que permitió determinar una solución factible que se ajustara a sus necesidades, se enmarcara dentro de los plazos de tiempo establecidos, y que sin duda mejorará notablemente su trabajo diario.

La elección del paradigma y la metodología a utilizar durante el desarrollo del proyecto fue también objeto de estudio, ya que resultó fundamental tener claro cual sería el proceso de desarrollo de software que serviría de guía durante la elaboración del proyecto y cual sería el enfoque utilizado para especificar la solución. Para ello se seleccionó el Proceso Unificado como paradigma de desarrollo, dado las ventajas que este proveía. Como metodología de desarrollo se seleccionó la Orientación a Objetos. Esta se realizó en base al paradigma escogido, puesto que la elección de uno conllevaba a la elección del otro, y además por las ventajas que proporciona el análisis y diseño orientado a objetos.

En relación a lo anterior, la utilización de UML (Unified Modeling Language) como herramienta de modelado viene siendo ya un estándar en el desarrollo de software, siendo utilizada por muchos equipos de desarrollo alrededor del mundo. UML a través de los

diferentes diagramas que provee, facilitó mucho el análisis y diseño de sistemas, ayudando a definir las diferentes funcionalidades que tendrían los sistemas y como se comportarían estos una vez puesto en producción.

Hoy en día, la aparición de nuevas tecnologías, es algo que sin duda no deja de sorprender. Esto resalta aún más cuando estas pueden ser implementadas en las aplicaciones que son de desarrollo propio. La utilización de Ajax (Asynchronous Javascript and XML) en el proyecto fue todo un reto. Comprender de qué se trata y cómo funciona esta tecnología, cuáles serían las ventajas que le daría a la aplicación y como mejoraría su funcionamiento, sin duda fue lo que motivó su estudio.

El estudio de cada uno de los puntos mencionados resultó fundamental para evaluar cada una de las alternativas existentes y, en base a ellas, poder tomar las decisiones correctas que permitiesen desarrollar un software de calidad.

Ya terminada la construcción de ambos sistemas, es posible analizar, evaluar y mirar hacia atrás, y ver como se fue formando este proyecto, que en una primera instancia se veían como dos sistemas bien acotados en cuanto a su tamaño y cómo, con el pasar del tiempo estos fueron creciendo e iban apareciendo nuevas funcionalidades, como también la dificultad de estas.

En la actualidad, con la experiencia adquirida y la satisfacción de haber cumplido los objetivos propuestos para el proyecto, dentro de los plazos establecidos, es posible determinar que se construyeron dos aplicaciones de calidad y que su puesta en producción inminente, sin duda mejorará las actividades diarias de los futuros usuarios de los sistemas.

Referencias Utilizadas

- [1] <http://www.ssvq.cl/>, Sitio Web del Servicio de Salud Viña del Mar – Quillota.
- [2] Pressman R.S. Ingeniería del Software. Un enfoque práctico. 5ª Edición. McGraw-Hill.
- [3] <http://www.ilustrados.com/publicaciones/EpyuVuuAIEAFjVSJEI.php#CARACT>
- [4] UML y Patrones 2da. Edición (Larman, Graig).

[5] <http://www.cs.umd.edu/hcil/quis/> Sitio Web oficial.

[6] <http://sumi.ucc.ie/> Sitio Web oficial.

[7] http://lsi.ugr.es/~arroyo/inndoc/doc/pruebas/pruebas_d.php