

**UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMATICA**



PROYECTO DE GRADO

**“SISTEMA DE GESTION HOSPITALARIO MODULO CONSULTA
EXTERNA CASO: SEGURO SOCIAL UNIVERSITARIO”**

PARA OPTAR AL TÍTULO DE LICENCIATURA EN INFORMATICA
MENCION: INGENIERIA DE SISTEMAS INFORMATICOS

**POSTULANTE: MARCO AURELIO AVALOS NINA
TUTORA METODOLOGICA: M.SC. FATIMA CONSUELO DOLZ DE MORENO
ASESOR: LIC. JOSE LUIS ZEBALLOS ABASTO**

LA PAZ – BOLIVIA
2013



**UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMÁTICA**



LA CARRERA DE INFORMÁTICA DE LA FACULTAD DE CIENCIAS PURAS Y NATURALES PERTENECIENTE A LA UNIVERSIDAD MAYOR DE SAN ANDRÉS AUTORIZA EL USO DE LA INFORMACIÓN CONTENIDA EN ESTE DOCUMENTO SI LOS PROPÓSITOS SON ESTRICTAMENTE ACADÉMICOS.

LICENCIA DE USO

El usuario está autorizado a:

- a) visualizar el documento mediante el uso de un ordenador o dispositivo móvil.
- b) copiar, almacenar o imprimir si ha de ser de uso exclusivamente personal y privado.
- c) copiar textualmente parte(s) de su contenido mencionando la fuente y/o haciendo la referencia correspondiente respetando normas de redacción e investigación.

El usuario no puede publicar, distribuir o realizar emisión o exhibición alguna de este material, sin la autorización correspondiente.

TODOS LOS DERECHOS RESERVADOS. EL USO NO AUTORIZADO DE LOS CONTENIDOS PUBLICADOS EN ESTE SITIO DERIVARA EN EL INICIO DE ACCIONES LEGALES CONTEMPLADOS EN LA LEY DE DERECHOS DE AUTOR.

Dedico mi Proyecto de Grado :

A Dios por darme la vida, a quien debo todos mis logros porque siempre ha estado a mi lado, guiándome, protegiéndome.

A mi mamita Valentina, por todo el cariño, aliento, comprensión, confianza y apoyo que me brindo en todos estos años.

A mi hermanita Luz Clarita, que confió siempre en mí, y siempre supo brindarme su alegría y compañía

A la memoria de mi Papito Jaime, siempre será parte importante en mi vida, gracias por el ejemplo que me diste papá, sé que desde el cielo me cuidas.

A mi novia Erika, por brindarme todo su apoyo en las buenas y en las malas, gracias por tu comprensión y dedicación, te amo mucho.

AGRADECIMIENTOS

A Dios por iluminar mi camino y permitirme llegar hasta donde estoy.

A mis Padres y hermanita por su comprensión, por su incondicional amor, por su apoyo, por ser la razón de mi vida, por ser la fuerza que me impulsa a seguir adelante.

Agradecer al Ingeniero. Gerardo Sandoval y al Seguro Social Universitario por su colaboración, tiempo, enseñanzas y conocimientos para culminar con éxito el presente trabajo.

A mi tutora, M.Sc. Fatima Consuelo Dolz de Moreno, por todos los conocimientos impartidos y por sus valiosos aportes que apoyaron a la conclusión del trabajo.

A mi revisor, Lic. Jose Luis Zeballos Abasto por toda la dedicación y paciencia que tuvo para explicarme y revisar mi proyecto durante toda la etapa del desarrollo del trabajo. Y por sus consejos y sugerencias desinteresadas que me serán muy importantes toda mi vida.

A los licenciados Leonardo Caceres y Adelaida Baptista por el apoyo brindado durante la realización de este proyecto.

A todos los docentes de la Carrera de Informática, por inculcarme sus conocimientos, sabiduría y valores durante el transcurso de mi vida universitaria.

A una persona muy especial en mi vida y en mi corazón Erika que ha estado siempre a mi lado apoyándome incondicionalmente y dándome ánimos.

A todos mis amigos y amigas de la Carrera de Informática con quienes compartimos momentos inolvidables, gracias por su amistad y su apoyo.

A los bibliotecarios Fernando, Willy y Daniel por el apoyo brindado en el transcurso de estos años.

Muchas gracias a todos de todo corazón y que Dios y la Virgen los Bendiga siempre.

RESÚMEN

El Seguro Social Universitario de La Paz, es una entidad que presta servicios médicos a sus afiliados. El seguro cuenta actualmente con aproximadamente 12 550 afiliados cantidad que se incrementa constantemente y con ello el volumen de información que se maneja en las distintas especialidades al momento de realizar la consulta médica va creciendo.

El Hospital cuenta con un sistema de gestión hospitalario (GEHMA v2.0) que cubre los módulos de afiliación, admisión, consulta externa y otros más. Sin embargo el módulo de Consulta Externa, desarrollado en este proyecto presentaba muchas deficiencias, como ser la visualización de formularios innecesarios, entre otros, por lo cual no satisfacía los requerimientos del usuario, por estas razones que se vio la necesidad de rediseñar el sistema con el objetivo de agilizar el proceso de una manera más eficiente proporcionando de esta manera ahorro de tiempo en las distintas tareas que realizan.

Para la construcción del Sistema de Gestión Hospitalaria, se emplea la metodología RUP, que posibilita la abstracción de procedimientos con la ayuda de herramientas UML, trabajando así con el paradigma orientado a objetos y facilitando la conceptualización de los procesos administrativos y médicos del Seguro Social Universitario. Para lograr que el sistema sea de fácil mantenimiento.

Con el presente proyecto se garantiza total comunicación entre las distintas áreas del Seguro Social Universitario, abreviando procedimientos y presentando la información detallada del seguimiento médico realizado a los pacientes.

Una vez concluido el sistema se procedió a elaborar las pruebas de funcionamiento, con las que se pudo constatar que el sistema responde a los requerimientos institucionales, posibilitando la optimización de los procesos y mejorando la administración de la información, además de la obtención de datos confiables y oportunos.

ABSTRACT

Seguro Social Universitario de La Paz, is an entity that provides medical services to its members. Insurance currently has approximately 12 550 members is constantly increasing amount and thus the volume of information handled in different specialties at the time of medical consultation grows.

The Hospital has a hospital management system (GEHMA v2.0) modules covering affiliation, admission, outpatient and more. However Outpatient module , developed in this project had many shortcomings , such as the display of unnecessary forms , among others, which did not meet user requirements are for these reasons that it became necessary to redesign the system in order to expedite the process more efficiently thereby providing time savings in the different tasks they perform.

For the construction of Hospital Management System, RUP methodology was used, which allows procedural abstraction with the help of UML tools , working well with the object-oriented paradigm and facilitating the conceptualization of administrative and medical processes University Social Security . To make the system easy to maintain.

With this project ensures full communication between different areas of Social Security University, shortening procedures and presenting detailed information made medical monitoring for patients. .

Once the system function tests were performed , with which it was found that the system responds to the institutional requirements , enabling the optimization of processes and improving information management , in addition to obtaining reliable and timely data .

Índice

Capítulo 1 Presentación

1.1 Introducción	1
1.2 Antecedentes	2
1.2.1 De la institución	2
1.2.1.1 Misión	3
1.2.1.2 Visión	3
1.2.1.3 Estructura organizacional.....	3
1.2.2 De proyectos Similares	6
1.3 Problemática.....	6
1.3.1 Problema principal	9
1.4 Objetivos	9
1.4.1 Objetivo General	9
1.4.2 Objetivos Específicos.....	9
1.5 Límites y Alcances	10
1.6 Justificación.....	10
1.6.2 Justificación Técnica.....	10
1.6.3 Justificación Social.....	11
1.7 Aporte.....	11
1.7.1.1 Metodología Scrum.....	12
1.7.1.2 Metodología RUP	12
1.7.1 Justificación de las metodologías.....	12
1.7.2 Herramientas	13
1.8 Metodologías y Herramientas	12
Capítulo 2 Marco Teórico	
2.1 Introducción	15
2.2 Metodologías.....	16
2.2.1. Metodología RUP	16

2.2.1.1 Ciclo de Vida.....	17
2.2.1.2 Etapas de RUP	17
2.2.1.2.1 Etapa de ingeniería.....	18
2.2.1.3 Fases.....	19
2.2.1.4 Especificación de las Fases	23
2.2.1.5 Implementación del RUP para el proyecto	24
2.3 ¿Qué es reingeniería?	24
2.3.1 Reingeniería de sistemas	25
2.3.2 Ingeniería de Software	25
2.4 Lenguaje Unificado de Modelado (UML)	26
2.4.1 Diagrama de casos de uso	28
2.4.2 Diagrama de clases.....	28
2.4.3 Diagrama de Actividades	30
2.4.4 Diagrama de interacción	31
2.4.4.1 Diagrama de secuencia.....	31
2.4.4.2 Diagrama de colaboración.....	32
2.4.5 Diagrama de estados	33
2.5 Herramientas del Proyecto	33
2.5.1 Lenguaje de Programación PHP	33
2.5.2 Microsoft SQL-Server 2005.....	34
2.5.2.1 Plataforma de datos de SQL Server	35
2.5.3 AJAX	37
2.6. Métodos para la Obtención de Información.....	38
2.6.1. Entrevistas.	38
2.6.2. Selección de Información.....	38
2.6.3. Observación.....	38
2.7 Calidad de Software	38
2.8 Seguridad Del Sistema	41
2.8.1 ¿Qué son las Políticas de Seguridad?.....	41

Capítulo 3 Marco Aplicativo

3.1. Introduccion	44
3.2 Análisis y Requerimientos Aplicando Reingenieria de Sistemas.	45
3.2.1 Análisis del sistema GEHMA Modulo Consulta Externa.....	45
3.2.2.1 Procesos	46
3.2.2.2 Interfaces del sistema actual.....	46
3.3 Fase Inicial	47
3.3.1.1 Requerimientos Funcionales	48
3.3.1.2 Requerimientos no funcionales	49
3.3.1Requerimientos del sistema.....	48
3.3.2. Descripción del Sistema.....	49
3.4. Fase de Elaboración	49
3.4.1 Modelado del Negocio	50
3.4.2 Descripción de los actores.....	50
3.4.3 Modelo de Casos de Uso de Sistema	50
3.4.4. Identificación de Casos de Uso	51
3.4.4.1 Diagrama de casos de uso Inicio de sesión	51
3.4.4.2 Diagrama de casos de uso Administración de usuarios	53
3.4.4.3 Diagrama de casos de uso Consulta médica	56
3.4.4.4 Diagrama de casos de uso Consulta especialidad	60
3.4.4.5 Diagrama de casos de uso Examen físico	62
3.4.4.6 Diagrama de casos de uso Diagnostico	64
3.4.5 Diagrama de Actividad	66
3.4.5.1 Diagrama de actividad Ingreso al sistema.....	66
3.4.5.2 Diagrama de actividad Atencion al paciente.....	67
3.4.5.3 Diagrama de actividad Consulta detallada.....	69
3.4.5.4 Diagrama de actividad Atencion al paciente nuevo o re consulta	71
3.4.6. Diagrama de Secuencia	71
3.4.6.1 Diagrama de Secuencia Inicio de sistema.....	71

3.4.6.2 Diagrama de Secuencia Revisión paciente	72
3.4.6.3 Diagrama de Secuencia Privilegios administrador	73
3.4.7. Diagrama de Clases.....	74
3.4.8 Modelo Entidad Relación.....	77
3.4.9 Modelo Relacional	77
3.4.10 Diagramas de Colaboración	79
3.4.11 Diseño de Seguridad	80
3.4.11.1 Diseño de niveles de acceso.....	80
3.4.11.2 Diseño de seguridad de datos, hardware y software	80
3.5 Implementación.....	82
Capítulo 4 Seguridad y Calidad de Software	
4.1 Políticas de Seguridad.....	87
4.2 Criterios de Seguridad.....	87
4.2.1 Seguridad en la red.....	88
4.2.2 Seguridad de la aplicación	88
4.2.3 Seguridad de la base de datos.....	88
4.3 Métricas de Calidad del Proyecto.	88
4.3.1 Funcionalidad.....	89
4.3.2 Mantenibilidad	94
4.3.3 Usabilidad	95
4.3.4 Portabilidad	96
4.3.5 Confiabilidad.....	97
4.4 Análisis Costo – Beneficio.....	99
Capítulo 5 Conclusiones y Recomendaciones	
5.1 Conclusiones	102
5.2 Recomendaciones.....	103
Bibliografía	
Anexos	
Documentación	

Índice de Figuras

Figura 1.1 Diagrama Organizacional	5
Figura 2.1 Ciclo de vida de la metodologia RUP.....	17
Figura 2.2 Representacion de una clase	28
Figura 2.3 Diagrama de secuencia	32
Figura 2.4 Diagrama de colaboracion	33
Figura 3.1 Fases de Racional Unified Process-RUP	45
Figura 3.2 Diagrama de caso de uso Iniciar sesion	51
Figura 3.3 Diagrama caso de uso Administraacion de usuarios	53
Figura 3.4 Diagrama caso de uso Consulta medica	56
Figura 3.5 Diagrama de caso de uso Consulta especialidad	58
Figura 3.6 Diagramas de caso de uso Examen Fisico	59
Figura 3.7 Diagrama de caso de uso Diagnostico	60
Figura 3.8 Diagrama de actividad Ingreso sistema	61
Figura 3.9 Diagrama de actividad Atencion paciente	62
Figura 3.10 Diagrama de actividad Consulta detallada	63
Figura 3.11 Diagrama de actividad Atencion paciente nuevo o reconsulta	65
Figura 3.12 Diagrama de secuencia Inicio sistema	66
Figura 3.13 Diagrama de secuencia Revision paciente.....	67
Figura 3.14 Diagrama de secuencia Privilegios administrador.....	67
Figura 3.15 Diagrama de Clases	68
Figura 3.16 Metodo de las Clases	69
Figura 3.17 Modelo Entidad Relacion	70
Figura 3.18 Modelo Relacional.....	70
Figura 3.19 Diagrama de colaboracion Inicio sesion.....	71
Figura 3.20 Inicio de Sesion	74
Figura 3.21 Administracion de usuarios	75
Figura 3.22 Administracion de roles.....	75
Figura 3.23 Administracion de permisos	76
Figura 3.24 Antecedentes Familiares.....	76
Figura 3.25 Antecedentes no patològicos	77
Figura 3.26Antecedentes Patològicos	77
Figura 3.27 Administraciòn de Contenidos	78
Figura 3.28 Inicio de Consulta	78

Índice de Tablas

Tabla 1.1 Problemas de Consulta Externa	7
Tabla 2.1 Ciclo de vida de la metodologia RUP.....	12
Tabla 3.1 Requerimieentos funcionales	48
Tabla 3.2 Requerimientos no funcionales.....	49
Tabla 3.3 Referencias del sistema.....	49
Tabla 3.4 Actores del sistema	50
Tabla 3.5 Iniciar Sesion.....	52
Tabla 3.6 Administracion de Usuarios.....	53
Tabla 3.7 Consultas medicas.....	57
Tabla 3.8 Inicio de sesion consultas medicas.....	60
Tabla 3.9 Examen fisico.....	62
Tabla 3.10 Elaboracion de diagnosticos.....	64
Tabla 4.1 Evaluacion de funcionalidad.....	82
Tabla 4.2 Factores de ajuste	83
Tabla 4.3 Cuestionario de funcionalidad	83
Tabla 4.4 Comparacion de pruebas	85
Tabla 4.5 Cuestionario en el aspecto de usabilidad	87
Tabla 4.6 Comparacion de pruebas	88
Tabla 4.7 Confiabilidad.....	89
Tabla 4.8 Analisis de la confiabilidad.....	91
Tabla 4.9 Coefecientes COCOMO	92

1. PRESENTACIÓN

1.1 Introducción

En la actualidad nos enfrentamos a una tecnología que ha ido evolucionando cada vez más, debido a esto que los sistemas informáticos se han convertido en una parte importante de instituciones y empresas para mejorar la administración de la información, siendo este un motivo necesario estar a la par de las nuevas tecnologías para un mejor control y funcionamiento de las mismas.

El uso de un sistema integrado para una institución de atención médica hace que la misma sistematice la información sobre los historiales clínicos de los pacientes; obteniendo de esa manera mayor control de tratamiento de enfermedades y como resultado mejores servicios.

En ese entendido, las empresas e instituciones que brindan servicios, como es el caso del Seguro Social Universitario, va introduciendo nuevos sistemas informáticos, puesto que con la ayuda de estos se puede lograr mayores beneficios en la optimización de procesos, la clasificación de datos y búsquedas de información de manera más confiable y ágil.

Actualmente el Seguro Social Universitario atiende a 12 550 afiliados de población en la ciudad de La Paz, [Seguro Social Universitario, 2013]; generando un gran volumen de información. Entre los afiliados del seguro podemos mencionar a trabajadores de distintas empresas, administrativos, docentes y estudiantes de la Universidad Mayor de San Andrés. El incremento de afiliados del seguro conlleva a la atención de más pacientes por día y con ello la dificultad de organizarlos y gestionarlos adecuadamente el historial clínico de cada afiliado. Actualmente el seguro cuenta con un sistema

hospitalario llamado GEHMA V.1.0 que abarca los módulos: afiliación, admisión, consulta externa, farmacia, almacenes y otros.

En este proyecto se abarca el módulo de Consulta Externa que facilita la sistematización y redacción de historiales clínicos para integrar la información que se genera en otros módulos del sistema con los que cuenta el sistema GEHMA V 2.0.

El módulo actual de Consulta Externa del Seguro Social Universitario no satisface las necesidades de los doctores en diferentes ámbitos, si bien cumple con los requerimientos mínimos estos no son suficientes. Se solicitaron algunas modificaciones y correcciones al módulo, para que exista una mayor fluidez de información y optimización al tiempo de atención. Por esta razón se solicitó reingeniería de procesos que ayuden con el sistema de acuerdo a las correcciones presentadas por los médicos de las distintas especialidades.

1.2 Antecedentes

1.2.1 De la institución

El Seguro Social Universitario es una entidad autónoma miembro de la OISS (Organización Iberoamericana de Seguridad Social), forma parte integrante del Sistema de Seguros Universitarios de Bolivia. En su desenvolvimiento técnico y administrativo se rige por los principios jurídicos, economía, oportunidad y eficacia en el otorgamiento de las prestaciones de salud y regímenes especiales con arreglo al Código de Seguro Social, su decreto reglamentario, disposiciones conexas y los propios estatutos vigentes en la institución.

El Seguro Social Universitario La Paz, es una entidad de derecho público con personería jurídica y autonomía de gestión técnica, financiera y administrativa propia, institución que funciona bajo la tuición del Honorable Consejo Universitario de la Universidad

Mayor de San Andrés, teniendo su campo de aplicación a los trabajadores de las universidades del departamento de La Paz y otras instituciones afiliadas.

Actualmente la institución gestora realiza las siguientes prestaciones

- a) Prestaciones en Especie del Seguro de Enfermedad.
- b) Prestaciones en Dinero del Seguro de Enfermedad.
- c) Prestaciones en Especie del Seguro de Maternidad
- d) Prestaciones en Dinero del Seguro de Maternidad.
- e) Prestaciones en Especie del Seguro de Riesgos Profesionales a Corto Plazo.
- f) Prestaciones en Dinero del Seguro de Riesgos Profesionales a Corto Plazo.

El Seguro Universitario La Paz, se rige por los principios y valores establecidos en el Código de Ética del Seguro Social Universitario, así como las prestaciones de Salud, están regidas por el Código de Seguridad Social, su reglamento y disposiciones conexas.

1.2.1.1 Misión

Prestar servicios de salud en los regímenes de Enfermedad, Maternidad y Riesgos Profesionales a Corto Plazo, a su población asegurada y beneficiaria, mediante servicios de Medicina Preventiva, Curativa, de Rehabilitación y Promocional [Seguro Social Universitario, 2011].

1.2.1.2 Visión

Constituirse en una institución líder que conlleve la Calidad, Calidez, Eficiencia y Eficacia como premisas para llegar a la Excelencia, en la prestación de Servicios de Salud [Seguro Social Universitario, 2011].

1.2.1.3 Estructura organizacional

Actualmente el Seguro Social Universitario de La Paz cuenta con 13584 afiliados.

Posee una estructura organizacional detallada en la figura 1.1 , que nos servirá de

mucha para la identificación de las especialidades con la que cuenta el SSU , tipos de estudio que se pueden realizar , además de la relación que pueda existir entre Admisión o Fichaje , Vigencia de derechos , Aportes , Afiliación, farmacia, Bioestadística, Internación.



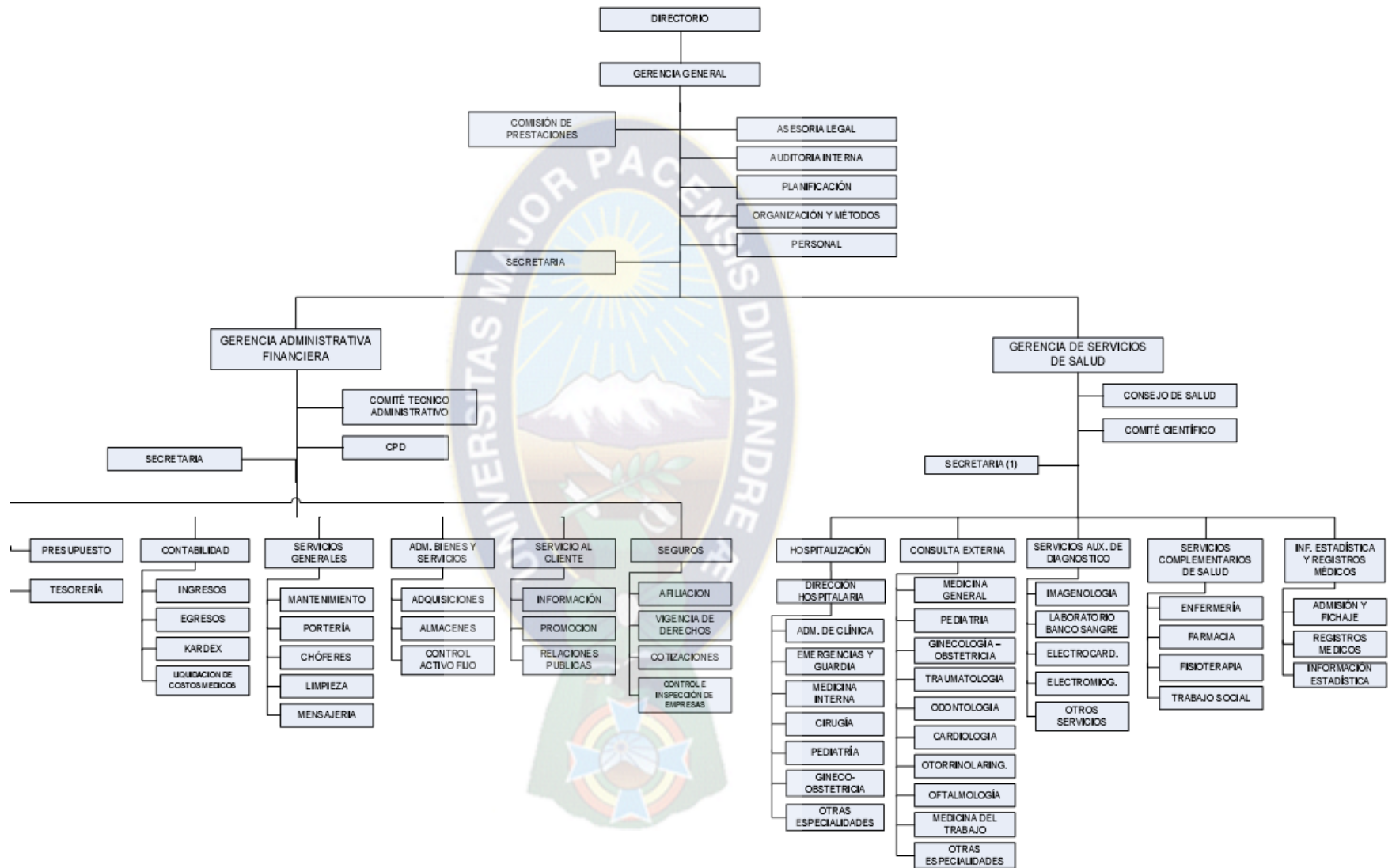


Figura 1.1 Diagrama Organizacional
Fuente: [Seguro Social Universitario, 2011]

1.2.2 De proyectos Similares

Según bibliografía revisada en biblioteca de la carrera de informática los proyectos similares que se pudo encontrar son:

- Sistema de gestión hospitalaria módulo afiliación, admisión y consulta externa, caso: Seguro Social Universitario Cochabamba. Soria Balderrama, Marcelo Hessel. Universidad Mayor de San Andrés. 2007
- Sistema de gestión hospitalaria modulo admisión caso: Seguro Social Universitario Vanessa Daniela Duarte Condori Universidad mayor de San Andrés 2012
- Sistema de gestión hospitalaria modulo Aportes y Vigencia de derechos caso: Seguro Social Universitario, Nancy Nora Zapana Coaquira Universidad Mayor de San Andrés 2012

1.3 Problemática

Habiendo hecho una reingeniería de procesos existentes en el módulo de Consulta Externa (Anexo 5) del Seguro Social Universitario GEHMA v.1.0 se pudo observar las siguientes falencias:

- Desconformidad de los doctores con el sistema actual, en vista de que no cumple con los requerimientos que solicita el médico para el llenado de formularios en la elaboración de diagnósticos, diferenciando formularios para cada especialidad.
- El sistema no verifica la existencia de fármacos recetados al paciente en farmacia.
- El sistema no cuenta con la opción de añadir nuevos términos médicos a la base de datos, como también nuevas enfermedades que puedan presentar los pacientes.

- El sistema no almacena información para el área de bioestadística, ya que el área en cuestión requiere acceso a esa información para poder generar reportes para SEDES, INLASA, etc.
- El sistema no cuenta con la opción de derivar de una especialidad a otra (Ejemplo: un médico de medicina general desea derivar a otras especialidades como ginecología, dermatología, urología, etc.)
- El sistema no cuenta con la opción de comunicación directa con laboratorios para agendar citas de manera rápida
- El sistema no cuenta con el libro CIE10 que ayuda a los doctores a generar estadísticas de enfermedades

En vista de los problemas encontrados en el actual módulo de Consulta Externa se plantean soluciones (ver tabla 1) observando a quien(es) afecta y el impacto que tiene.

Problema	Afecta a	Impacto	Solución
Los medicamentos recetados a los pacientes algunas veces no existen en farmacia	Paciente	Los medicamentos no encontrados en farmacia son comprados fuera del SSU , ocasionando disgustos de los pacientes	El sistema de consulta externa tiene que tener una relación con farmacia para ver si el medicamento recetado al paciente existe o no
Los términos médicos que se van actualizando gestión a gestión y	Medico	Como el sistema no cuenta con la opción de ingresar una nueva	El doctor al ingresar al sistema debe contar con la

<p>detección de nuevas enfermedades no están almacenadas en la base de datos</p>		<p>enfermedad o términos médicos tiende a redactar el diagnóstico a mano</p>	<p>opción de poder añadir nuevos términos y enfermedades si no los existiera para optimizar el tiempo de atención</p>
<p>El sistema no cuenta con una comunicación directa con los laboratorios de imagenología</p>	<p>Paciente</p>	<p>El paciente tiene que ir de oficina en oficina para ser atendido.</p>	<p>El doctor debe poder solicitar mediante sistema agendar al paciente a los laboratorios de imagenología</p>
<p>El sistema no cuenta con el libro CIE10 que ayuda a los doctores a generar estadísticas de enfermedades</p>	<p>Medico</p>	<p>Al introducir la enfermedad a la base de datos y dejar pasar como un simple resfrió , no se genera un reporte de la importancia de la nueva enfermedad</p>	<p>El sistema debe contar con el libro CIE10 para casos extremos de personas que tengan nuevas enfermedades y pueda ser atendido y puesto bajo vigilancia para que no pueda contagiar a otras</p>

			personas
--	--	--	----------

Tabla1.1 Problemas de consulta externa

Fuente: [Elaboración Propia]

1.3.1 Problema principal

El sistema actual de consulta externa del SSU (Seguro Social Universitario) de la ciudad de La Paz no cumple con los requerimientos necesarios de los médicos para la redacción sistematizada de diagnósticos y tratamientos.

1.4 Objetivos

1.4.1 Objetivo General

Implementa un sistema de Consulta externa para el Seguro Social Universitario de La Paz, para brindar al médico la sistematización en la redacción de diagnósticos y tratamientos a los pacientes.

1.4.2 Objetivos Específicos

- Implementar una interfaz que cuente con la opción de llenado de datos, exclusiva para cada especialidad.
- Implementar las interfaces que debe existir entre consulta externa y farmacia, para conocer la existencia de los medicamentos en farmacia
- Implementar un submódulo que permita almacenar información para el área de bioestadística.
- Implementar una opción exclusiva para que los médicos actualicen los términos que tiene cada especialidad.
- Permitir adicionar al sistema detección de nuevas enfermedades, para tener la base de datos actualizada.

- Implementar una interfaz de comunicación con laboratorios de imagenología para que el paciente que necesite sea agendado directamente.
- Implementar el acceso rápido al libro CIE10 para ayudar a los doctores a generar estadísticas de enfermedades.
- Implementar una opción que permita al médico derivar a otras especialidades y ver si el doctor se encuentra disponible en la fecha deseada.

1.5 Límites y Alcances

El sistema de consulta externa del Seguro Social Universitario de la ciudad de La Paz será desarrollado e implementado para el GEHMA v 2.0 con los respectivos testeos de funcionamiento con los siguientes requerimientos:

- Permitir al doctor el llenado de todos los campos que son necesarios y no tenga que escribir lo que falta de manera manuscrita.
- Crear la interfaz que se relaciona con todas las especialidades
- Crear la interfaz que pueda solicitar a farmacia la cantidad existente de cualquier fármaco.
- Que el paciente puede ser derivado a internación, quirófano o terapia intensiva dependiendo de la gravedad del caso. La información que el doctor genera al realizar una consulta, puede ser vista y analizada por cualquier miembro del personal médico del hospital de forma detallada.
- El doctor pueda actualizar los términos médicos que ya no se usan.
- El doctor puede añadir a la base de datos detección de nuevas enfermedades para que la base de datos este actualizada

1.6 Justificación

1.6.2 Justificación Técnica

Técnicamente se justifica porque en la implantación del sistema se utilizan herramientas informáticas que faciliten tanto el desarrollo como el mantenimiento del sistema.

El seguro social cuenta con el equipo tecnológico necesario para emprender el desarrollo del proyecto, además se actualiza con el avance tecnológico en el campo de la informática con el uso de los programas de PHP es un lenguaje que me permite la generación dinámica y el uso del gestor de base de datos SQL Server 2005 para almacenar todos los historiales Clínicos.

1.6.3 Justificación Social

Socialmente el sistema se justifica porque los beneficios que brinda el sistema será tanto para los doctores que lo administran, facilitando sus labores cotidianas y reduciendo el tiempo de atención a los pacientes, como para las personas que reciban la atención, puesto que esta será más rápida y eficaz porque se tendrá más claro el seguimiento y evolución de sus enfermedades.

Con la implementación del sistema integrado de consulta externa facilita a todos los usuarios una demora en la tardanza de encontrar sus historias clínicas y solicitar estudios, medicamentos, estudios, etc.

1.7 Aporte

El proyecto de grado tiene como aporte el desarrollo de un sistema de consulta externa para el Seguro Social universitario de la ciudad de La Paz que será de mucha utilidad para los médicos al momento de la redacción de diagnósticos Para tener una comunicación con el paciente en un tiempo moderado.

El paciente también será beneficiado con la implementación del sistema, agilizando su tiempo de espera en farmacia, en laboratorios de imagenología, derivación a otras especialidades, etc.

1.8 Metodologías y Herramientas

El proyecto utilizara dos metodologías

- Scrum: para la planificación y control de calidad.
- RUP: para la parte de la documentación

1.7.1 Justificación de las metodologías

1.7.1.1 Metodología Scrum

La metodología Scrum se utiliza más que todo para el desarrollo del sistema desde la reingeniería de los procesos con la que cuenta el sistema de consulta externa del Seguro Social Universitario de la ciudad de La Paz hasta la implementación del nuevo sistema , sobre todo la metodología se utiliza para el control de calidad del producto debido a su gran facilidad de adaptarse a cambios que ocurran en el transcurso del desarrollo del sistema y su posterior cambio con la relación que existirá entre los doctores , pacientes y el equipo de desarrollo para nuevas modificaciones , llevando a la elaboración de un sistema que cumpla con todos los requerimientos de los médicos .

La metodología Scrum se caracteriza por hacer reuniones diarias de 15 minutos donde se da un detalle detallado de los avances que existe cada día con los requerimientos que puedan surgir en el transcurso del desarrollo y la implementación del sistema de consulta externa.

1.7.1.2 Metodología RUP

La metodología RUP se utilizara para la parte de la documentación, para llevar un control continuo del comportamiento del sistema al momento de realizar la relación que tendrán los doctores con los pacientes, doctores con doctores al momento de la derivación del paciente a otra especialidad, etc.

Con esta metodología podremos realizar los manuales de usuario, las tarjetas CRC, los diagramas UML para que el doctor se sienta familiarizado con el sistema y no tenga ningún problema en el uso.

1.7.2 Herramientas

Entre las herramientas que se utiliza son:

Base De Datos: “Base de Datos es un conjunto de información estructurada en registros y almacenada en un soporte electrónico legible desde un ordenador. Cada registro constituye una unidad autónoma de información que puede estar a su vez estructurada en diferentes campos o tipos de datos que se recogen en dicha base de datos” [Matsukawa, 2005]

Una Base de Datos es un conjunto auto descriptivo de registros integrados persistente que es utilizado este sistema. Por persistentes queremos decir, de manera intuitiva, que el tipo de datos de la base de datos difieren de otros datos más efímeros como los datos de entrada y salida.

Motor de Base de Datos SQL SEVER: “Es el servicio para almacenamiento, procesamiento y manejo de la seguridad de los datos. Proporciona acceso controlado y procesamiento rápido de transacciones para satisfacer los

requerimientos de las aplicaciones empresariales que demandan gran consumo de datos, y soporte para alta disponibilidad de datos” [Matsukawa, 2005].

El sistema se implementa con la ayuda de este motor de base de datos ya que el Seguro Social Universitario cuenta las respectivas licencias.

Servidor Web Apache, Xampp: Es un servidor local que permite que podamos acceder a nuestro sitio en forma local (en nuestro ordenador) como si estuviésemos accediendo a él a través de internet, es decir, con una dirección. Xampp una distribución de Apache que incluye MySQL, PHP y otras herramientas para el desarrollo de aplicaciones web, como phpMyAdmin.

Lenguaje de Programación PHP: Es un lenguaje de script del lado del servidor, los scripts PHP están incrustados en los documentos HTML y el servidor los interpreta y ejecuta antes de servir las páginas al cliente, el cliente no ve el código PHP sino los resultados que produce.

Es un lenguaje de programación con variables, sentencias condicionales, ciclos (bucles), funciones, no es un lenguaje de marcado como podría ser HTML, XML o WML. Está más cercano a JavaScript o a C, es por tal motivo que se decide realizar con este sistema con este lenguaje.

Lenguaje de Modelado Unificado (UML)

Lenguaje estándar para describir planos de software, el cual puede utilizarse para visualizar, construir y documentar los artefactos de los sistemas.

Para ayuda en el análisis del sistema se empleara el siguiente diagrama:

Diagrama de Clases.- Que muestra el conjunto de clases y objetos importantes que forman parte de un sistema, junto con las relaciones existentes entre las clases y objetos

2. MARCO TEORICO

2.1 Introducción

El presente proyecto de grado tiene por objetivo desarrollar e implementar un sistema que pertenezca al módulo de consulta externa para el GEHMA V 2.0 del Seguro Social Universitario de la ciudad de La Paz que cumplirá con todos los requerimientos planteados por los doctores de las diferentes especialidades (ver anexo 6).

Se ayuda al doctor a llenar el diagnóstico presuntivo y final del paciente de manera más rápida con la ayuda del sistema, el paciente también se beneficiará con el sistema ya que el mismo presentará una relación más directa con farmacia, imagenología, etc.

Con la reingeniería de los procesos se pudo observar muchas falencias que tiene el actual sistema de consulta externa del GEHMA V 1.0, que no cubre con todos los requerimientos de los doctores mencionados a continuación:

- Términos médicos no actualizados
- No existe comunicación con farmacia, para ver si el medicamento recetado está en farmacia o no, para poder informar al paciente
- No existe comunicación entre los laboratorios de imagenología, para que el doctor pueda agendar al paciente y que los estudios realizados lleguen de manera rápida en caso de emergencia
- El sistema no cuenta con el archivo CIE10 para que le facilite al doctor en los reportes que debe enviar el doctor de pacientes críticos
- El sistema no tiene la facilidad de comunicación con todas las especialidades al momento de la derivación de un paciente a otra especialidad.

- El sistema no cuenta con un control riguroso de todos los campos que son necesarios al momento de la elaboración del diagnóstico.

2.2 Metodologías

2.2.1. Metodología RUP

El Rational Unified Process o Proceso Unificado de Rational. Es un proceso de ingeniería de software que suministra un enfoque para asignar tareas y responsabilidades dentro de una organización de desarrollo. Su objetivo es asegurar la producción de software de alta calidad que satisfaga la necesidad del usuario final dentro de un tiempo y presupuesto previsible. Es una metodología de desarrollo iterativo enfocada hacia “los casos de uso, manejo de riesgos y el manejo de la arquitectura”.

El RUP mejora la productividad del equipo ya que permite que cada miembro del grupo sin importar su responsabilidad específica acceda a la misma base de datos de conocimiento. Esto hace que todos compartan el mismo lenguaje, la misma visión y el mismo proceso acerca de cómo desarrollar software.

2.2.1.1 Ciclo de Vida

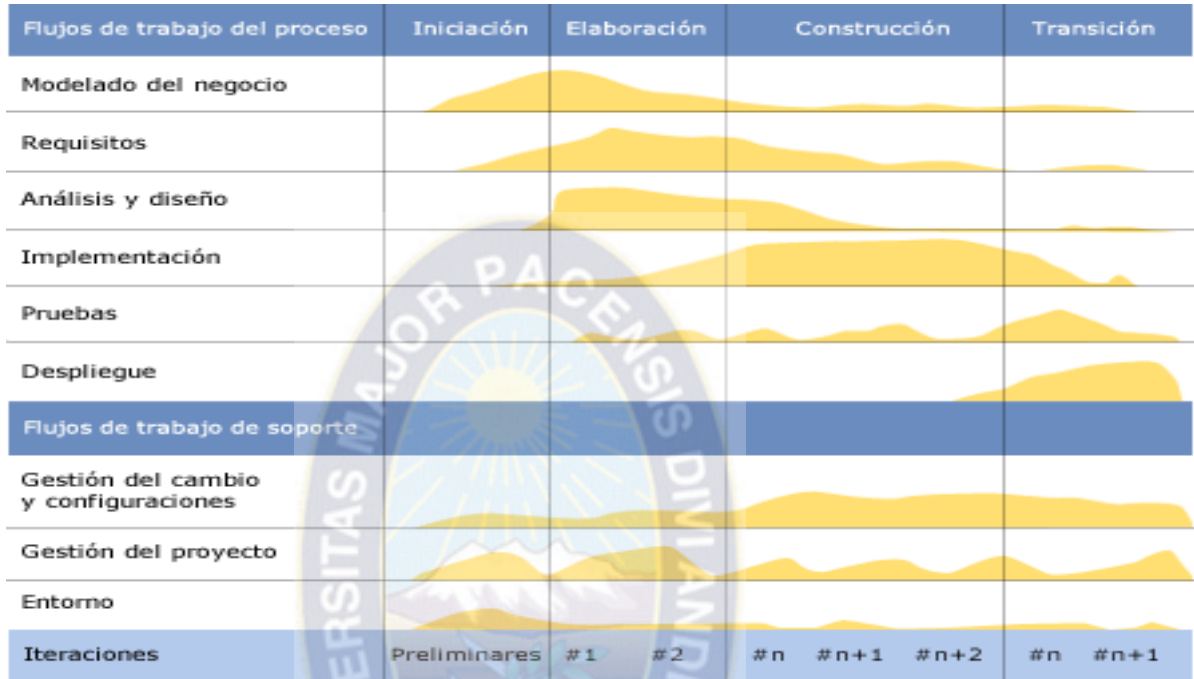


Figura 2.1.Ciclo de vida de la metodología RUP

Fuente: [Pressman, 2005]

En el ciclo de vida RUP (ver figura 2.1) veremos una implementación del desarrollo en espiral. Con el ciclo de vida se establecen tareas en fases e iteraciones. El RUP maneja el proceso en cuatro fases, dentro de las cuales se realizan varias iteraciones en número variable

Las primeras iteraciones (en las fases de Inicio y Elaboración) se enfocan hacia la comprensión del problema y la tecnología, la delimitación del ámbito del proyecto, la eliminación de los riesgos críticos, y al establecimiento de una base de inicio

2.2.1.2 Etapas de RUP

Estructuralmente RUP está dividido en dos etapas: de ingeniería y de producción.

Donde la etapa de ingeniería cubre las fases de inicio y elaboración y la etapa de producción cubre las etapas de construcción y transición.

2.2.1.2.1 Etapa de ingeniería

Esta etapa agrupa las fases de inicio y de elaboración, lo que básicamente da por objetivo la conceptualización del sistema y el diseño inicial de la solución del problema.

Se identifican los riesgos y se establece su plan de manejo, se ajusta ese plan según la priorización de riesgo y la de casos de usos vs. Riesgos, para determinar en qué orden y en que iteraciones se desarrollarán los artefactos de software que son la solución a los casos de uso.

Se identifican los recursos necesarios, tanto económicos como humanos, acordes con las necesidades del proyecto y se da comienzo al proceso de estimación y planificación inicial a un nivel macro para todo el proyecto, posteriormente se realiza una estimación detallada de tiempo y recursos de las fases de concepción y elaboración.

Fase 1: Preparación inicial (Incepción)

Su función principal es establecer los objetivos para el ciclo de vida del producto.

En esta fase se establece el caso del negocio con el fin de delimitar el alcance del sistema, saber que se cubrirá y delimitar el alcance del proyecto.

Fase 2: Preparación detallada (Elaboración)

Su objetivo principal es plantear la arquitectura para el ciclo de vida del producto.

En esta fase se realiza la captura de la mayor parte de los requerimientos funcionales, manejando los riesgos que interfieran con los objetivos del sistema, acumulando la información necesaria para el plan de construcción y obteniendo suficiente información para hacer realizable el caso del negocio.

2.3.1.3 Etapa de producción

En esta etapa se realiza un proceso de refinamiento de las estimaciones de tiempos y recursos para las fases de construcción y transición, se define un plan de mantenimiento para los productos entregados en la etapa de ingeniería, se implementa los casos de uso pendientes y se entrega el producto al cliente, garantizando la capacitación y soporte adecuados.

Fase 3: Construcción

Su objetivo principal es alcanzar la capacidad operacional del producto. En esta fase a través de sucesivas iteraciones e incrementos se desarrolla un producto software, listo para operar, éste es frecuentemente llamado operación beta.

Fase 4: Transición

Su objetivo principal es realizar la entrega del producto operando, una vez realizadas las pruebas de aceptación por un grupo especial de usuarios y habiendo efectuado los ajustes y correcciones que sean requeridos.

2.2.1.3 Fases

Fase de inicio

La fase de inicio puede tomar varias formas. Para algunos proyectos, es una plática en la máquina de café: “Por favor pongan los catálogos de servicio en la Web”. Para proyectos mayores, puede ser un estudio de factibilidad completo que tome meses.

Durante la fase de inicio se desarrolla una descripción del producto final, y se presenta el análisis del negocio. Esta fase responde las siguientes preguntas:

- ¿Cuáles son las principales funciones del sistema para los usuarios más importantes?
- ¿Cómo podría ser la mejor arquitectura del sistema?

- ¿Cuál es el plan del proyecto y cuánto costará desarrollar el producto?

El objetivo de esta fase es ayudar a decidir cuáles son los verdaderos objetivos del proyecto. Las iteraciones exploran diferentes soluciones posibles, y diferentes arquitecturas posibles.

Puede que todo el trabajo físico realizado en esta fase sea descartado. Lo único que normalmente sobrevive a la fase de inicio es el incremento del conocimiento del equipo.

Los artefactos que típicamente sobreviven a esta fase son:

- Un modelo del negocio.
- Un enunciado de los mayores requerimientos, generalmente como casos de uso.
- Un boceto inicial de la arquitectura.
- Una descripción de los objetivos del proyecto.
- Una versión muy preliminar del plan del proyecto.

Debe poder responder las siguientes cuestiones:

- ¿Se ha llegado a un acuerdo con todas las personas involucradas sobre los requisitos funcionales y no funcionales del sistema?
- ¿Se ha determinado con claridad el ámbito del sistema?
- ¿Se ha determinado lo que va a estar dentro del sistema y fuera del sistema?
- ¿Se vislumbra una arquitectura que pueda soportar estas características?
- ¿Se identifican los riesgos críticos?
- ¿Es factible para su organización llevar adelante el proyecto?

Al final de la fase de inicio, se examinan los objetivos del ciclo de vida del proyecto y se decide si se procede o no con el desarrollo.

Fase de elaboración

Durante esta fase se especifican en detalle la mayoría de los casos de uso del producto (resultado del análisis a los requisitos) y se diseña la arquitectura del sistema.

La arquitectura se expresa en forma de planos (modelados) trazados desde diferentes vistas, los cuales juntos representan el sistema entero. Esto implica que hay vistas arquitectónicas expresadas mediante los diagramas, del modelo de casos de uso, del modelo de análisis, del modelo de diseño, del modelo de implementación y modelo de despliegue.

Las iteraciones en la fase de elaboración:

- Establecen una firme comprensión del problema a solucionar.
- Eliminan los mayores riesgos.
- Establecen la fundación arquitectural para el software.
- Forman un plan detallado para las siguientes iteraciones.

Fase de construcción

Durante esta fase de construcción, se lleva a cabo la construcción del producto por medio de una serie de iteraciones las cuales se seleccionan algunos Casos de Uso, se redefine su análisis y diseño y se procede a su implantación y pruebas. En esta fase se realiza una pequeña cascada para cada ciclo, se realizan tantas iteraciones hasta que se termine la nueva implementación del producto.

Al final de esta fase, el producto contiene todos los casos de uso implementados, sin embargo puede que no esté libre de defectos. Los artefactos producidos durante esta fase son:

- El sistema software.
- Los casos de prueba.

- Los manuales de usuario

Hay proyectos en los cuales las pruebas y la integración son dejadas al final. Las pruebas y la integración son tareas grandes, y siempre toman más tiempo de lo que la gente piensa. Atrás en el tiempo, en los días de OS/360, se estimaba que la mitad de un proyecto era pruebas y corrección de errores. Las pruebas y la integración son más difíciles cuando se dejan al final, y más desmoralizadoras. Todo este esfuerzo lleva a un gran riesgo. Con el desarrollo Iterativo se realiza el proyecto entero en cada iteración, lo que conduce al hábito de lidiar con todo los problemas cada vez.

Fase de transición

Durante esta fase de transición busca garantizar que se tiene un producto preparado para su entrega al usuario.

La fase de transición cubre el periodo durante el cual el producto se convierte en versión beta. En la versión beta un número reducido de usuarios con experiencia prueba el producto e informa de defectos y deficiencias. Los desarrolladores corrigen los problemas e incorporan algunas de las mejoras sugeridas en una versión general dirigida a la totalidad de la comunidad de usuarios. La fase de transición conlleva actividades como la fabricación, formación del cliente, el proporcionar una línea de ayuda y asistencia, y la corrección de los defectos que se encuentran tras la entrega. El equipo de mantenimiento suele dividir esos defectos en dos categorías: los que tienen suficiente impacto en la operación para justificar una versión incrementada y los que pueden corregirse en la siguiente versión normal. La fase de transición finaliza con el hito de lanzamiento del producto.

También al final de la fase de transición se decide si los objetivos del ciclo de vida han sido cumplidos, y posiblemente sí se debe iniciar otro ciclo de desarrollo. Éste es también un punto donde se empacan algunas de las lecciones aprendidas en este proyecto para mejorar el proceso.

2.2.1.3 Principales características

- Forma disciplinada de asignar tareas y responsabilidades (quién hace qué, cuándo y cómo)
- Pretende implementar las mejores prácticas en Ingeniería de Software
- Desarrollo iterativo
- Administración de requisitos
- Uso de arquitectura basada en componentes
- Control de cambios
- Modelado visual del software
- Verificación de la calidad del software

El RUP es un producto de Rational (IBM). Se caracteriza por ser iterativo e incremental, estar centrado en la arquitectura y guiado por los casos de uso. Incluye artefactos (que son los productos tangibles del proceso como por ejemplo, el modelo de casos de uso, el código fuente, etc.) y roles (papel que desempeña una persona en un determinado momento, una persona puede desempeñar distintos roles a lo largo del proceso).

2.2.1.4 Especificación de las Fases

- Establece oportunidad y alcance
- Identifica las entidades externas o actores con las que se trata
- Identifica los casos de uso

RUP comprende 2 aspectos importantes por los cuales se establecen las disciplinas:

Proceso: Las etapas de esta sección son:

- Modelado de negocio
- Requisitos

- Análisis y Diseño
- Implementación
- Pruebas
- Despliegue

Soporte: En esta parte nos conseguimos con las siguientes etapas:

- Gestión del cambio y configuraciones
- Gestión del proyecto
- Entorno

La estructura dinámica de RUP es la que permite que este sea un proceso de desarrollo fundamentalmente iterativo, y en esta parte se ven inmersas las 4 fases descritas anteriormente:

- Inicio(También llamado Incepción)
- Elaboración
- Desarrollo(También llamado Implementación, Construcción)
- Cierre (También llamado Transición)

2.2.1.5 Implementación del RUP para el proyecto

La metodología RUP es más apropiada para proyectos grandes (Aunque también pequeños), dado que requiere un equipo de trabajo capaz de administrar un proceso complejo en varias etapas. En proyectos pequeños, es posible que no se puedan cubrir los costos de dedicación del equipo de profesionales necesarios.

2.3 ¿Qué es reingeniería?

Reingeniería es el diseño rápido y radical de los procesos estratégicos de valor agregado y de los sistemas, las políticas y las estructuras organizacionales que los sustentan para optimizar los flujos del trabajo y la actividad de una organización.

La reingeniería no es cuestión de hacer mejoras marginales o incrementales sino de dar saltos gigantescos en rendimiento [Calatayud, 2005]

2.3.1 Reingeniería de sistemas

Recupera información sobre el diseño de un programa existente y utiliza esta información para reestructurar o reconstruir el programa existente, con vistas a adaptarlo a un cambio, ampliarlo o mejorar su calidad general, con el objetivo de conseguir una mayor facilidad de mantenimiento en el futuro (esto es lo que se denomina mantenimiento preventivo) [Hammer, 1995]

2.3.2 Ingeniería de Software

Lewis define software como la suma de los programas de computadora, procedimientos, reglas, la documentación asociada y los datos que pertenecen a un sistema de cómputo [Lewis, 1974].

Por otro lado Cota menciona que un producto de software es un producto diseñado para un usuario. En este contexto, la ingeniería de software es un enfoque sistemático del desarrollo, operación, mantenimiento y retiro del software, que en palabras más llanas, se considera que la ingeniería de Software es la rama de la ingeniería que aplica los principios de la ciencia de la computación y las matemáticas para lograr soluciones costo-efectivas (eficaces en costo o económicas) a los problemas de desarrollo de software, es decir, permite elaborar consistentemente productos correctos, utilizables y costo-efectivos [Cota, 1994].

Mientras que Jacobson dice que el proceso de desarrollo de software es aquel en que las necesidades del usuario son traducidas en requerimientos de software, estos requerimientos transformados en diseño y el diseño implementado en código, el código es probado, documentado y certificado para su uso operativo. Concretamente define

quien está haciendo qué, cuándo hacerlo y como alcanzar un cierto objetivo [Jacobson, 1992].

Sin embargo Pressman define Ingeniería de Software como el análisis, diseño, construcción, verificación y gestión de unidades [Pressman, 2005].

El proceso de desarrollo de software requiere por un lado un conjunto de conceptos, una metodología y un lenguaje propio. A este proceso también se le llama el ciclo de vida del software que comprende cuatro grandes fases: concepción, elaboración, construcción y transición.

2.4 Lenguaje Unificado de Modelado (UML)

Se define a “el Lenguaje Unificado de Modelado pre-escribe un conjunto de notaciones y diagramas estándar para modelar sistemas orientados a objetos, y describe la semántica esencial de lo que estos diagramas y símbolos”.

UML es una consolidación de muchas de las notaciones y conceptos más usados orientados a objetos. Empezó como una consolidación del trabajo de Grade Booch, James Rumbaugh, e Ivar Jacobson, creadores de tres de las metodologías orientadas a objetos más populares, UML pre-escribe una notación estándar y semánticas esenciales para el modelado de un sistema orientado a objetos.

Previamente, un diseño orientado a objetos podría haber sido modelado con cualquiera de la docena de metodologías populares, causando a los revisores tener que aprender las semánticas y notaciones de la metodología empleada antes que intentar entender el diseño en sí. Ahora con UML, diseñadores diferentes modelando sistemas diferentes pueden sobradamente entender cada uno los diseños de los otros, gracias al modelo que representan a un sistema de software desde una perspectiva específica. Al igual que la planta y el alzado de una figura en dibujo técnico nos muestran la misma figura vista

desde distintos ángulos, cada modelo nos permite fijarnos en un aspecto distinto del sistema.

El desarrollo de sistemas con UML siguiendo el proceso unificado incluye actividades específicas, cada una de ellas a su vez contiene otras subactividades las cuales sirven como una guía de cómo deben ser las actividades desarrolladas y secuenciadas con el fin de obtener sistemas exitosos; consecuentemente el desarrollo de sistemas puede variar de desarrollador en desarrollador, de proyecto en proyecto, de empresa en empresa adoptando siempre un proceso de desarrollo. UML es un lenguaje de modelado unificado basado en una notación gráfica la cual permite: especificar, construir, visualizar y documentar los objetos de un sistema programado.

Los diagramas del UML son los siguientes:

- Diagrama de Casos de Uso
- Diagrama de Clases
- Diagrama de Objetos
- Diagrama de Secuencia
- Diagrama de Colaboración
- Diagrama de Estados
- Diagrama de Actividades
- Diagrama de Componentes
- Diagrama de Despliegue (o Implementación)

Los diagramas empleados por UML son los siguientes:

2.4.1 Diagrama de casos de uso

Un diagrama de casos de uso, muestra las distintas operaciones que se esperan de una aplicación o sistema y cómo se relaciona con su entorno (usuario u otras aplicaciones).

Los casos de uso se representan en el diagrama de una elipse que denota un requerimiento solucionado por el sistema. Cada caso de uso es una operación completa desarrollada por los actores y por el sistema de diálogo.

2.4.2 Diagrama de clases

Un diagrama de clases o estructura estática muestra el conjunto de clases y objeto importante que forman parte de un sistema, junto con las relaciones existentes entre clases y objetos.

Clase: representa un conjunto de entidades que tienen propiedades comunes.

Una clase de constructor que define la estructura y comportamiento de una colección de objetos denominados instancia de la clase. En UML la clase está representada por un rectángulo con tres divisiones internas, son los elementos fundamentales del diagrama.

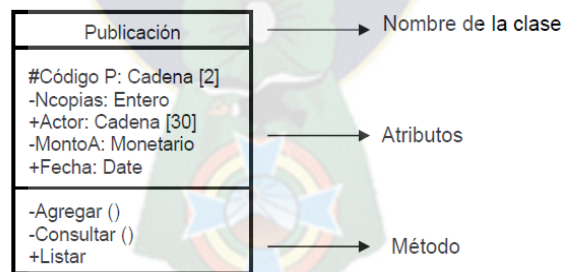


Figura 2.2 Representación de una clase

Fuente: [Rumbaugh 2000]

Atributo: Representa una propiedad de una entidad.

Operación: El conjunto de operaciones que describen el comportamiento de los objetos de una clase.

Objeto: Es una instancia de una clase. Se caracteriza por tener una identidad única, un estado definido por un conjunto de valores de atributos y un comportamiento representado por sus operaciones y métodos.

Asociación (rol, multiplicidad, calificador): representan las relaciones entre instancias de clase. Una asociación es una línea que une dos o más clases.

Dependencia: Es una relación donde existen entidades independientes y otras dependientes, lo que implica que cambiar el elemento independiente puede requerir cambios en los dependientes. Se representa con una línea punteada direccional, indicando el sentido de la dependencia.

Los tipos de asociaciones entre clases presentes en un diagrama estático son:

- Asociación Binaria
- Asociación n-aria
- Composición

Dependencia existencial: El elemento dependiente desaparece al destruirse el que lo contiene y, si es de cardinalidad 1, es creado al mismo tiempo. Hay una pertenencia fuerte, se puede decir que el objeto contenido es parte constitutiva y vital del que lo contiene.

Los objetivos contenidos no son compartidos, esto es lo no hacen parte del estado de otro objeto.

Agregación Relaciona una clase ya ensamblada con una clase componente. Es también una relación de composición menos fuerte (no se exige dependencia existencial) y se denota por un rombo sin rellenar en uno de los extremos.

Generalización: Es un proceso de abstracción en el cual un conjunto de clases existentes, que tienen atributos y métodos comunes, es referido por una clase genérica a

un nivel mayor de abstracción. La relación de generalización denota una relación de herencia entre clases.

La subclase hereda todos los atributos y mensajes descritos en la superclase.

Refinamiento: Es una relación que representa la especificación completa e lado que ya ha sido especificado con cierto nivel de detalle. Por ejemplo, una clase del diseño es un refinamiento de una clase de análisis.

2.4.3 Diagrama de Actividades

Un diagrama de actividades es un caso especial de un diagrama de estados en el cual casi todos los estados son estados de acción (identifican que acción se ejecuta al que esta en él) y casi todas las transiciones son enviadas al terminar la acción ejecutada en el estado anterior.

Generalmente modelan los pasos de un algoritmo y puede dar detalle a un caso de uso, un objeto o un mensaje en un objeto, sirven para representar transiciones internas, sin hacer mucho énfasis en transiciones o eventos externos. Los elementos que conforman el diagrama son: **acción y transición.**

Estado de Acción: Representa un estado con una acción interna, con lo menos una transición que indica la culminación de la acción por medio de un evento implícito). Permite modular un paso dentro del algoritmo. Se representan por un rectángulo con bordes redondeados.

Transición: Es la relación entre dos estados y se encuentran unidos por flechas; indicando que un objeto que está en el primer estado, realizara una acción específica y entrara en el segundo estado, cuando un evento implícito ocurra y unas condiciones específicas sean satisfechas.

2.4.4 Diagrama de interacción

En los diagramas de interacción se muestra un patrón de interacción entre objetos. Por lo general, un diagrama de interacción captura el comportamiento de un único caso de uso.

Hay dos tipos de diagramas de interacción: **diagrama de secuencia** y **diagrama de colaboración**.

2.4.4.1 Diagrama de secuencia

Un diagrama de secuencia muestra una interacción ordenada según la secuencia temporal de eventos. En particular, muestra los objetos participantes en la interacción y en los mensajes que intercambian ordenados según su secuencia en el tiempo. El eje vertical representa el tiempo, y en el eje horizontal se colocan los objetos y actores participantes en la interacción, sin un orden prefijado. Cada objeto o actor tiene una línea vertical, y los mensajes se representan mediante flechas junto a las transiciones o activaciones a las que se refieren los objetos. El tiempo fluye de arriba abajo.

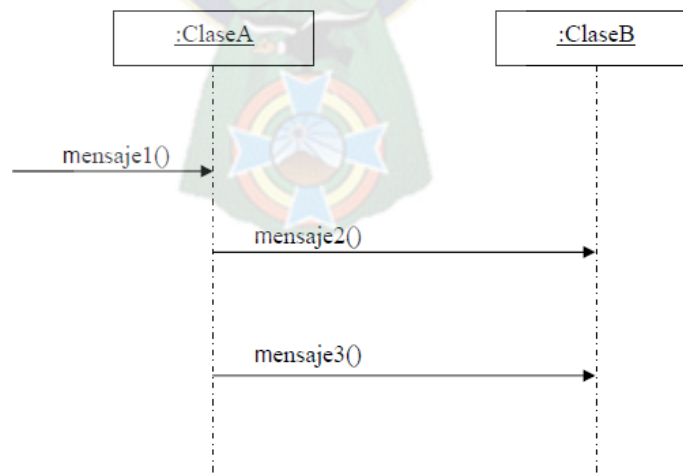


Figura 2.3 Diagrama de Secuencia

Fuente: [Rumbaugh 2000]

2.4.4.2 Diagrama de colaboración

Un diagrama de colaboración muestra una interacción organizada basándose en los objetos que toman parte en la interacción y los enlaces entre los mismos (en cuanto a la interacción se refiere). A diferencia de los diagramas de secuencia, los diagramas de colaboración muestran las relaciones entre los roles de los objetos. La secuencia de los mensajes y los flujos de ejecución concurrentes deben determinarse explícitamente mediante números de secuencia.

En cuanto a la representación, un diagrama de colaboración muestra a una serie de objetos con los enlaces entre los mismos, y con los mensajes que se intercambian dichos objetos. Los mensajes son flechas que van junto al enlace por el que “circulan”, y con el nombre del mensaje y los parámetros (si los tiene) entre paréntesis. Cada mensaje lleva un número de secuencia que denota cuál es el mensaje que le precede, excepto el mensaje que inicia el diagrama, que no lleva número de secuencia.

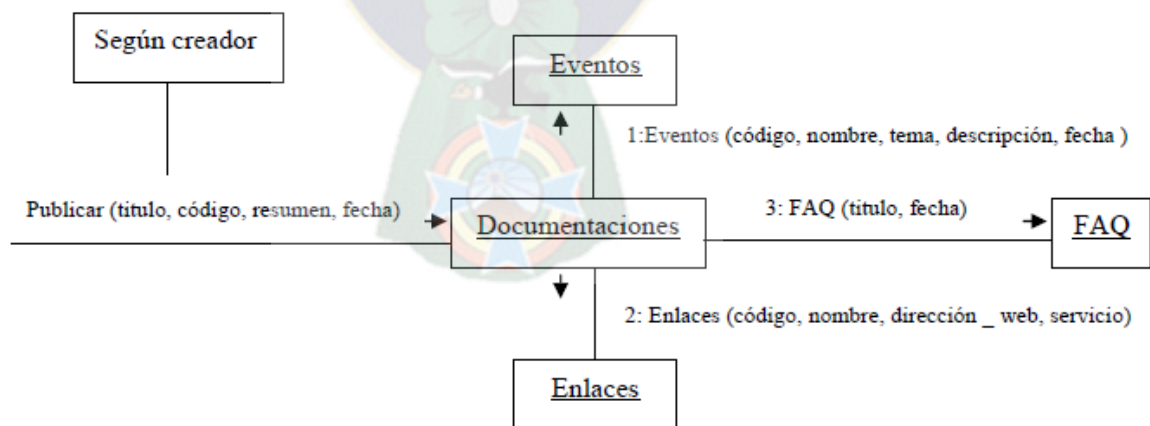


Figura 2.4 Diagrama de Colaboración

Fuente: [Rumbaugh, 2000]

2.4.5 Diagrama de estados

Muestra el conjunto de estados por los cuales pasa un objeto durante su vida en una aplicación junto con los cambios que permiten pasar de un estado a otro. Está representado principalmente por los siguientes elementos: **estado, elemento y transición.**

2.5 Herramientas del Proyecto

2.5.1 Lenguaje de Programación PHP

PHP, acrónimo de "PHP: Hypertext Preprocessor", es un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor. Es utilizado para la programación de aplicaciones web, es una herramienta multiplataforma aunque es utilizado primordialmente en servidores Web bajo Linux.

Quizá la característica más resaltante de PHP sea el amplio soporte para una gran cantidad de base de datos como ser: dBase, mSQL, MySQL, Oracle, PostgreSQL, entre otras.

PHP también soporta una serie de características tecnológicas. Esto incluye autenticación, XML, creación de imágenes dinámicas, soporte de memoria compartida y creación dinámica de documentación PDF.

2.5.2 Microsoft SQL-Server 2005

SQL Server 2005 es una plataforma global de base de datos que ofrece administración de datos empresariales con herramientas integradas de inteligencia empresarial.

El motor de la base de datos SQL Server 2005 ofrece almacenamiento más seguro y confiable tanto para datos relacionales como estructurados, lo que le permite crear y administrar aplicaciones de datos altamente disponibles y con mayor rendimiento para utilizar en su negocio.

El motor de datos SQL Server 2005 constituye el núcleo de esta solución de administración de datos empresariales. Asimismo, SQL Server 2005 combina lo mejor en análisis, información, integración y notificación. Esto permite que su negocio cree y despliegue soluciones de BI rentables que ayuden a su equipo a incorporar datos en cada rincón del negocio a través de tableros de comando, escritorios digitales, servicios Web y dispositivos móviles.

La integración directa con Microsoft Visual Studio, el Microsoft Office System y un conjunto de nuevas herramientas de desarrollo, incluido el Business Intelligence Development Studio, distingue al SQL Server 2005.

El siguiente diagrama, ver figura, ilustra los componentes básicos en SQL Server 2005, muestra cómo SQL Server 2005 es una parte importante de Windows Server System y se integra con la plataforma Microsoft Windows, incluidos Microsoft Office System y Visual Studio, para ofrecer soluciones que aportan datos a cada rincón de su organización Fuente

SQL Server 2005 introduce cientos de nuevas y mejores características. Estas características le ayudarán a progresar en su negocio en tres áreas clave:

- a) En Administración de datos empresariales donde SQL Server 2005 ofrece una plataforma de datos más confiable, segura y productiva para aplicaciones de unidad de negocios y analíticas. La última versión de SQL Server no sólo es la versión más grande de SQL Server, sino también la versión más segura.
- b) En la Productividad del encargado del desarrollo donde SQL Server 2005 brinda un entorno de desarrollo de extremo a extremo que incluye diversas tecnologías nuevas que otorgan poder a los encargados del desarrollo y aumentan considerablemente la productividad del encargado del desarrollo.
- c) En la Inteligencia empresarial, las capacidades globales analíticas, de integración y migración de datos de SQL Server 2005 permiten que las compañías amplíen el valor de sus aplicaciones existentes, sin perjuicio de la plataforma subyacente.

Las soluciones de BI creadas en SQL Server 2005 ofrecen información crítica y oportuna a todos los empleados, permitiéndoles tomar mejores decisiones más rápidamente.

2.5.2.1 Plataforma de datos de SQL Server

SQL Server es una solución de datos globales, integrados y de extremo a extremo que habilita a los usuarios en toda su organización mediante una plataforma más segura, confiable y productiva para datos empresariales y aplicaciones de BI.

SQL Server 2005 provee herramientas sólidas y conocidas a los profesionales de IT, así como también a trabajadores de la información, reduciendo la complejidad de la creación, despliegue, administración y uso de aplicaciones analíticas y de datos empresariales en plataformas que van desde los dispositivos móviles hasta los sistemas de datos empresariales.

A través de un conjunto global de características, la interoperabilidad con sistemas existentes y la automatización de tareas rutinarias, SQL Server 2005 ofrece una solución completa de datos para empresas de todos los tamaños. La Figura 8, a continuación muestra el diseño de la plataforma de datos SQL Server 2005.

Servidor Web Apache, Xampp: Es un servidor local que permite que podamos acceder a nuestro sitio en forma local (en nuestro ordenador) como si estuviésemos accediendo a él a través de internet, es decir, con una dirección. Xampp una distribución de Apache que incluye MySQL, PHP y otras herramientas para el desarrollo de aplicaciones web, como phpMyAdmin. Lenguaje de Programación PHP: Es un lenguaje de script del lado del servidor, los scripts PHP están incrustados en los documentos HTML y el servidor los interpreta y ejecuta antes de servir las páginas al cliente, el cliente no ve el código PHP sino los resultados que produce. Es un lenguaje de programación con variables, sentencias condicionales, ciclos (bucles), funciones, no es un lenguaje de marcado como podría ser HTML, XML o WML. Está más cercano a JavaScript o a C,

es por tal motivo que se decide realizar con este sistema con este lenguaje. Lenguaje de Modelado Unificado (UML)

Lenguaje estándar para describir planos de software, el cual puede utilizarse para visualizar, construir y documentar los artefactos de los sistemas. Para ayuda en el análisis del sistema se empleara el siguiente diagrama: Diagrama de Clases.- Que muestra el conjunto de clases y objetos importantes que forman parte de un sistema, junto con las relaciones existentes entre las clases y objetos.

Clase: Representa un conjunto de entidades que tienen propiedades comunes.

Objeto: Denominado instancia de la clase, representada por un rectángulo con tres divisiones internas.

Atributo: Representa una propiedad de una entidad.

Operación: El conjunto de operaciones que describen el comportamiento de los objetos de una clase.

2.5.3 AJAX

[Eguiluz, 2008]

El termino AJAX es JavaScript asíncrono + XML. Ajax no es una tecnología en sí mismo. Se trata de varias tecnologías independientes que se unen de formas nuevas y sorprendentes.

Las tecnologías que forman AJAX son:

- « XHTML (es básicamente HTML lenguaje de marcado de hipertexto expresado como XML valido) y CSS (hojas de estilo en cascada), para crear una presentación basada en estándares.

- « DOM (modelo de objetos del documento o modelo en objetos para la representación de documentos), para la interacción y manipulación dinámica de la presentación.
- « XML (lenguaje de marcas extensible) y JSON (Notación de objetos de JavaScript es un formato ligero de intercambio de datos), para el intercambio y la manipulación de Información.
- « XMLHttpRequest (interfaz empleada para realizar peticiones HTTP protocolo de transferencia de hipertexto y HTTPS es la versión segura de HTTP; a servidores Web), para el intercambio asíncrono de información.
- « JavaScript, para unir todas las demás tecnologías.

2.6. Métodos para la Obtención de Información.

Existen diferentes métodos para obtener datos relevantes para determinar los requerimientos de los usuarios, entre ellos podemos citar:

2.6.1. Entrevistas.

Para el proyecto se realiza especialmente entrevistas con el personal de la institución que consiste en preguntas abiertas en forma oral y escrita, con el objeto de determinar la información necesaria, procedimientos y opiniones de los requerimientos y necesidades existentes.

2.6.2. Selección de Información

También se tomó en cuenta información que ya existía disponible en la institución normas, reglas, manual de procedimientos y funciones de usuarios.

2.6.3. Observación

Se visita el ambiente de trabajo, con el fin de observar la forma de trabajo y los procesos que se realizan para obtener de información acerca del tema; mediante una guía de observación.

2.7 Calidad de Software

Un elemento primordial en el proceso de desarrollo de software es la medición, la cual se emplea para valorar la calidad de los productos que se construyen, esto con el propósito de alcanzar metas importantes en el desarrollo de software, la calidad de software.

Puesto que el objetivo implícito de los proyectos de software y más aun de los proyectos de grado es la calidad del producto.

La calidad de software es medible y puede variar de un sistema a otro o de un programa a otro. Un producto de software para ser explotado durante un largo periodo (10 años o más), necesita ser confiable, mantenible y flexible para disminuir los costos de mantenimiento y perfeccionamiento durante el tiempo de explotación.

Los factores de calidad se centralizan en tres aspectos importantes de un producto software: sus características operativas, su capacidad de cambios y su adaptabilidad a nuevos entornos [Pressman, 2005].

Por tanto los factores de calidad se basan en: Funcionalidad, Mantenibilidad, Confiabilidad y Portabilidad.

a) Funcionalidad

La funcionalidad se puede definir como la capacidad del producto software para proporcionar funciones que satisfagan las necesidades especificadas. Esta métrica no puede ser medida directamente es por eso que su cálculo debe derivar indirectamente de otras medidas directas. Una de estas medidas es el punto función que fue propuesta por Albretch. Los puntos de función se obtienen utilizando una función empírica basada en

medidas cuantitativas del dominio de información del software y valoraciones subjetivos de la complejidad del software.

b) Mantenibilidad

Este aspecto de calidad involucra los elementos que simplifica la labor de prevención, corrección o ampliación del código de programa.

La Mantenibilidad es la detección de corrección de fallas, también se basan en los cambios requeridos por el usuario, el tiempo del sistema es indefinido pero está sujeta a las nuevas necesidades del usuario:

El IEEE Std. 982.1-1998 sugiere un índice de madurez del software (IMS) que proporciona una indicación de la estabilidad de un producto de software (basados en los cambios que ocurren con cada versión del producto) [Pressman, 2005]. Se determina la siguiente información:

MT = el número de módulos en la versión actual.

Fc = el número de módulos cambiados en la versión actual.

Fa = el número de módulos añadidos a la versión actual.

Fd = el número de módulos de la versión anterior que se eliminaron en la actual.

El índice de la madurez del software se calcula de la siguiente manera:

A medida que el IMS se acerca a 1.0 el producto empieza a estabilizarse.

c) Confiabilidad

Este es el grado de confianza esperado por parte del usuario en la operación adecuada del sistema, es decir el grado que se puede esperar de una aplicación que lleve a cabo las operaciones especificadas y con la precisión requerida.

La confiabilidad se puede establecer en relación a:

Seguridad: Representa la capacidad de que el sistema no afecte ni sea afectado por su entorno y de quien los utiliza.

Disponibilidad: Define la probabilidad de que el sistema este funcionando en un tiempo determinado.

Protección: Represente la capacidad del sistema para protegerse a si mismo de instrucciones accidentales o programadas.

La confiabilidad del sistema es directamente proporcional a la calidad de sus componentes y se calcula de la siguiente manera:

Dónde:

$R(t)$: La función de confiabilidad de un tiempo t

λ : Error de tasa constante de fallo

T : Tiempo de operación del sistema.

d) Portabilidad

La portabilidad se define como el esfuerzo necesario para transferir el programa de un entorno (hardware/software) a otro entorno diferente.

La portabilidad tiene que ver con las variaciones no solo del hardware físico sino mas generalmente de la maquina hardware-software, la que realmente programamos y que incluye el sistema operativo, el sistema de ventanas y otras herramientas fundamentales. Muchas de las incompatibilidades existentes entre las plataformas son injustificadas, y convierte a la portabilidad en un asunto primordial tanto para los que desarrollan como para los que usan el software.

2.8 Seguridad Del Sistema

2.8.1 ¿Qué son las Políticas de Seguridad?

Políticas de seguridad son los documentos que describen, principalmente, la forma adecuada de uso de los recursos de un sistema de cómputo, las responsabilidades y derechos tanto de usuarios como administradores, describe lo que se va a proteger y de lo que se está tratando de proteger, estos documentos son el primer paso en la construcción de Firewalls efectivos. Las políticas son parte fundamental de cualquier esquema de seguridad eficiente [Pressman, 2005].

Como se mencionó anteriormente Intranet a pesar de ser es una red privada en la que se tienen grupos bien definidos y limitados, no se encuentra exenta de ataques que pudiesen poner en riesgo la información que maneja, ya que la mayoría de estos son provocados por sus mismos usuarios. La mayoría de las estadísticas de seguridad en cómputo indican que cerca del 80% de los fraudes relacionados con las computadoras provienen de los usuarios internos, por esto las intranets son las más vulnerables a ataques de esta índole.

Por tal es importante establecer normas y políticas de seguridad a fin de detener un sistema confiable, y seguro. Un punto muy importante dentro de las políticas es el que tienen que ir acompañadas de sanciones, las cuales deberán también ser redactadas, revisadas, autorizadas, aplicadas y actualizadas.

- **Políticas de contraseñas:** Son una de las políticas más importantes, ya que por lo general, las contraseñas constituyen la primera y tal vez única manera de autenticación y, por tanto, la única línea de defensa contra ataques. Estas establecen quien asignara la contraseña, que longitud debe tener, a que formato deberá apegarse, como será comunicada.

En PHP se utiliza la función MD5 (MessageDigest 5), que es una función hash irreversible (de un solo sentido), es decir, inscripta el password tecleado por el usuario y es imposible que partiendo desde la cadena encriptado se vuelva a la

contraseña origen. Por esto mismo no hay problema de que alguien pueda acceder al campo encriptado de la base de datos.

- **Políticas de uso adecuado:** Especifican lo que se considera un uso adecuado o inadecuado del sistema por parte de los usuarios, así como lo que está permitido y lo que está prohibido dentro del sistema de cómputo.



3. MARCO APLICATIVO

3.1. Introducción

En este capítulo se explicara en detalle los aspectos relacionados con la funcionalidad, organización, descripción de funciones y diferentes procesos que existen en el módulo de Consulta Externa.

Posteriormente detallaremos los procesos del nuevo sistema mediante la metodología RUP descrita en la figura 3.1 y diagramas del lenguaje de modelado unificado UML.

En este capítulo también se definirán los objetivos basados en el marco de trabajo del proyecto de grado cumpliendo así con las tareas y los procedimientos requeridos por el módulo de Consulta Externa.

A continuación enmarcaremos los procesos de Iteración que comprende el RUP:

Realizar la correcta planificación de cada proceso iterativo (se realiza un estudio de alto nivel en los riesgos del sistema).

- Se analiza los casos de uso y los escenarios requeridos.
- Se diseña cada opción arquitectónica que comprende el sistema.
- Se realiza la codificación y las pruebas pertinentes en el sistema integrando el nuevo código que se emplea con el ya existente de forma que se realiza las iteraciones de forma gradual durante su construcción.

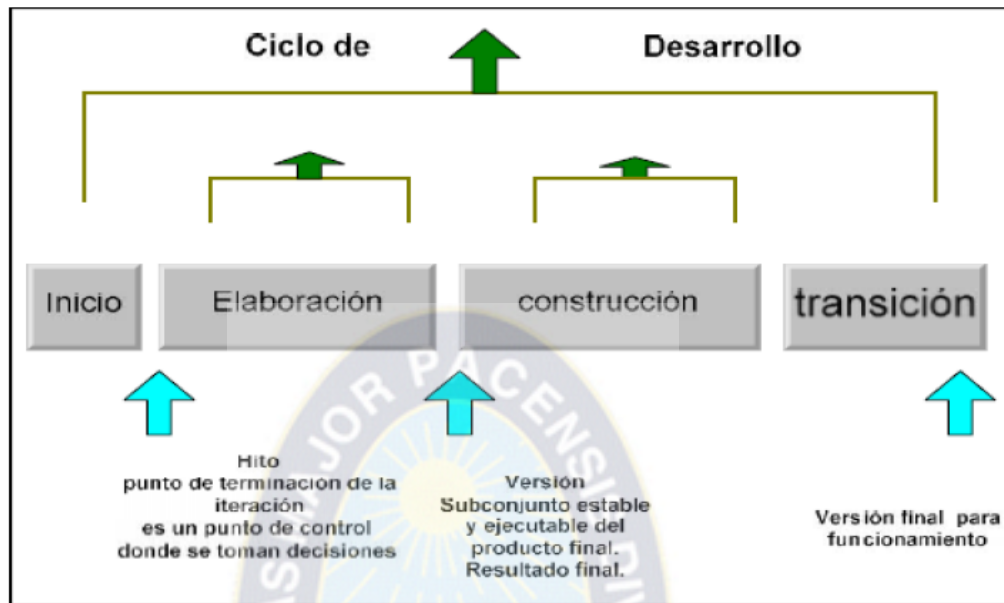


Figura 3.1 Fases de Racional Unified Process-RUP

Fuente [José, 2006-PDF]

3.2 Análisis Y Requerimientos Aplicando Reingeniería De Sistemas.

El proceso de análisis y requerimientos para la reingeniería del sistema, consiste en conocer y comprender las actividades que a diario la unidad de Consulta Externa realiza; así como el detalle de la información procesada por el Sistema Informático existente.

Este análisis permite plantear los requerimientos en los cuales se basara el sistema de gestión hospitalario, modulo Consulta Externa.

3.2.1 Análisis del sistema GEHMA v1.0 Modulo: Consulta Externa

El sistema GEHMA es una aplicación que conlleva diferentes módulos que han sido Desarrollados en Visual Studio 2005 y SQL SERVER 2005.

3.2.2.1 Procesos

Consiste en describir los procesos que se llevan a cabo en el área de Consulta Externa, debido a que estos proporcionan información de vital importancia al Seguro Social Universitario los cuales son:

- Registro de Diagnósticos Médicos, se procede al registrar y guardar los datos de la consulta por la cual vino el paciente.
- Registro de Solicitudes que se necesita en cada consulta médica.
- Registro de tratamientos
- Actualización de historias clínicas

3.2.2.2 Interfaces del sistema actual

Las principales interfaces con las que cuenta el sistema para dar soporte a los procesos son:

- Interfaz de Usuarios
 - Búsqueda de usuarios
 - Modificación de usuarios
 - Creación de usuarios
 - Eliminación de usuarios
- Interfaz de Roles
 - Creación de Roles
 - Modificación de Roles
 - Eliminación de Roles
 - Búsqueda de Roles
- Interfaz de Permisos
 - Habilitación de acciones
 - Acceso a Contenidos y programas
- Interfaz de Consulta Externa
 - Atención de pacientes de acuerdo a cita médica programada

- Especialidad
- Horario de atención
- Número de consultorio
- Búsqueda de Historia Clínica
- Clasificación Internacional de Enfermedades (CIE-10)
- Signos vitales
- Solicitud de exámenes de laboratorio, imagenología, fisioterapia, internación
- Solicitud de farmacia
- Solicitud de Junta Médica
- Solicitud de referencia (Derivar a otra especialidad)
- Intervienen las siguientes entidades:
 - Agenda de Doctor
 - Paciente
 - Médicos
 - Consultorio
 - Especialidad
 - Farmacia
 - Laboratorio
 - Consulta

3.3 Fase Inicial

En esta fase se analiza todos los problemas que presenta Consulta Externa, para luego representarlos como requisitos del sistema.

Siguiendo el organigrama de la institución el modelado de negocios se centrara en las áreas de: atención de Pacientes en las distintas especialidades

Con las siguientes actividades fundamentales:

- Registro de Usuarios
- Registro de Roles
- Registro de permisos con la que cuenta cada usuario

- Registro de diagnóstico de pacientes
- Registro de Historias Clínicas
- Registro de fármacos recetados
- Registro de Estudios clínicos
- Registro de exámenes de laboratorio
- Registro de Exámenes Físicos

En esta fase se establece el contexto del sistema mediante casos de uso logrando definir el alcance del proyecto. Para lograr tal efecto se debe conocer el ámbito del sistema, se define los límites del mismo para proponer una arquitectura, identificar los riesgos, identificar los actores y la interacción de los mismos.

3.3.1 Requerimientos del sistema

Se tiene dos tipos de requerimientos funcionales y no funcionales.

3.3.1.1 Requerimientos Funcionales

En la tabla 3.1 observamos requerimientos funcionales del proyecto

Poseer una estructura para administrar todos los módulos e usuarios del SSU.
Tener una interfaz para crear roles.
Tener una interfaz para el doctor, para poder administrar los formularios a ser llenados.
Tener una interfaz para la enfermera, para poder llenar los signos vitales del paciente.
Tener una interfaz para conectarse con farmacia.
Registrar datos de la consulta.
Registrar solicitudes hospitalarias.
Actualizar datos del paciente.
Actualizar datos de autenticación.
Registro de exámenes de laboratorio.
Registro de re consultas.

Tabla 3.1 Requerimientos funcionales

Fuente (Elaboración Propia)

3.3.1.2 Requerimientos no funcionales

En la Tabla 3.2 observamos requerimientos no funcionales que presenta el proyecto

EL sistema debe tener soporte con los tres navegadores IE , Chrome y Firefox
EL sistema debe ser fácil de manipular
El sistema debe contar con normas de seguridad

Tabla 3.2 Requerimientos no funcionales

Fuente (Elaboración Propia)

3.3.2. Descripción del Sistema

En la Tabla 3.3 observamos funciones que debe realizar el proyecto.

Ref.	Función	Categoría
R1	Ingreso de usuarios al sistema	Evidente
R2	Verificación de datos	Oculto
R3	Administrar roles	Oculto
R4	Administrar y actualizar datos del paciente	Evidente
R5	Llenar datos de los pacientes	Evidente
R6	Verificar datos del paciente	Evidente
R7	Llenar solicitudes	Evidente
R8	Recetar medicamentos	Evidente
R9	Actualizar datos de pacientes	Evidente
R10	Actualizar la autenticación de usuario	Oculto
R11	Registrar Historias Clínicas	Evidente

Tabla 3.3 Referencias del Sistema

Fuente (Elaboración Propia)

3.4. Fase de Elaboración

Analizar el problema y el modelo completo del negocio para establecer el dominio, además construir la arquitectura del proyecto.

3.4.1 Modelado del Negocio

A través del modelo de negocios hallaremos la técnica para comprender los procesos de organización del entorno. El modelo de negocios describe el proceso en términos de casos de uso y actores que intervienen en el contexto del sistema que corresponde al proceso de usuarios del mismo. El modelo de negocios esta detallado e identifica los procesos más importantes del contexto del sistema, describiendo procesos exactos relacionados con los actores. El modelo del negocio será representado por el diagrama de casos de uso general en donde se encuentra las solicitudes de los usuarios y las respuestas que brindara el nuevo sistema.

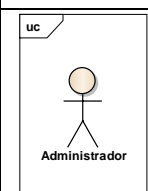
3.4.2 Descripción de los actores

Los actores o usuarios del sistema los cuales interactúan con el sistema está conformado por lo siguiente:

3.4.3 Modelo De Casos De Uso De Sistema

Un modelo de caso de uso del sistema describe lo que hace el sistema, este modelo contiene actores, casos de uso y sus relaciones.

En la tabla 3.4 se observa todos los casos de uso identificados.

Actores	Clasificación	Descripción
	Nivel 1	Este actor tiene acceso a todos los módulos del sistema el usuario puede adicionar, eliminar y modificar a la vez el privilegio que tiene cada usuario de acuerdo a la necesidad esta clasificación se encarga de administrar usuarios y roles dando permisos a ciertos programas

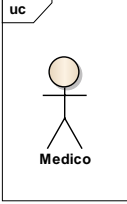
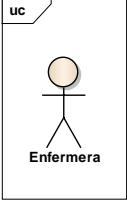
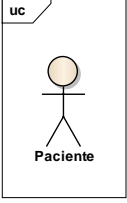
 <p>uc Medico</p>	<p>Nivel 2</p>	<p>Este actor tiene privilegios para atender pacientes ,llenar síntomas del paciente recetar medicamentos exámenes complementarios y lo más importante actualizar y guardar datos del paciente</p>
 <p>uc Enfermera</p>	<p>Nivel 3</p>	<p>Este actor tiene privilegios solo para llenar los signos vitales del paciente</p>
 <p>uc Paciente</p>	<p>Nivel4</p>	<p>Este actor solo accede a los servicios con la que cuenta el SSU</p>

Tabla 3.4 Actores del sistema

Fuente (Elaboración Propia)

3.4.4. Identificación de Casos de Uso

Los casos de uso nos ayudan a ver los procesos necesarios para poder realizar el sistema.

3.4.4.1 Diagrama de casos de uso Inicio de sesión

En la Figura 3.1 se puede observar el inicio de sesión que tienen los Usuarios del SSU (Seguro Social Universitario) y su posterior validación por motivos de seguridad.

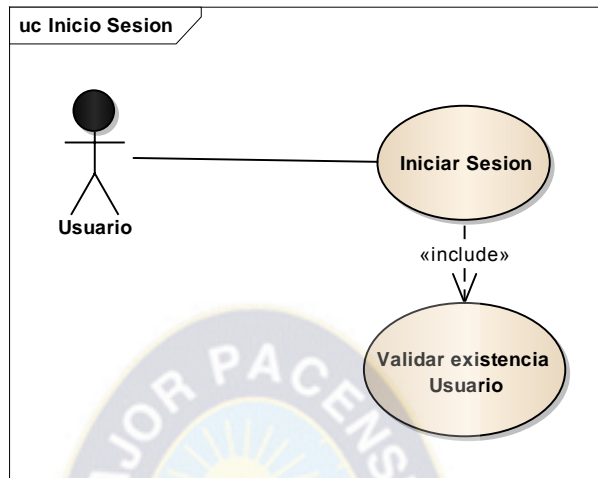


Figura 3.2 Diagrama de caso de Uso iniciar Sesión
Fuente (Elaboración Propia)

En la tabla 3.5 observamos con más detalle las funciones que cumple el usuario.

No. 1	Nombre de caso de uso: Iniciar sesión
Actor Principal: Usuario del Sistema, Sistema	
Personal involucrado e intereses: Usuario del Sistema: Es el usuario que desea ingresar al sistema para la realización de actividades de configuración y consulta	
Precondiciones: El Usuario, debe existir en el sistema	
Post-condiciones: Se presenta el menú personalizado	
Referencias Cruzadas (Casos de Uso): Todos los casos de uso.	
Flujo Básico: <ol style="list-style-type: none"> 1. El Usuario del Sistema ingresa usuario y contraseña 2. El Sistema valida los datos introducidos sean correctos y que el usuario se encuentre activo. 3. El Sistema presenta en pantalla el menú correspondiente al usuario 	
Flujos Alternativos: <ol style="list-style-type: none"> 2.a. El Sistema muestra que los datos ingresados no son correctos que usuario no existe o está bloqueado. 2.b. El Sistema indica que contraseña no es correcta y bloquea el usuario al 3er mal ingreso. 	

Tabla 3.5 Iniciar Sesión
Fuente Elaboración Propia

3.4.4.2 Diagrama de casos de uso Administración de usuarios

Se puede observar en la figura 3.3 la interfaz de usuario para crear, modificar, eliminar y búsqueda para que el usuario tenga ciertos privilegios.

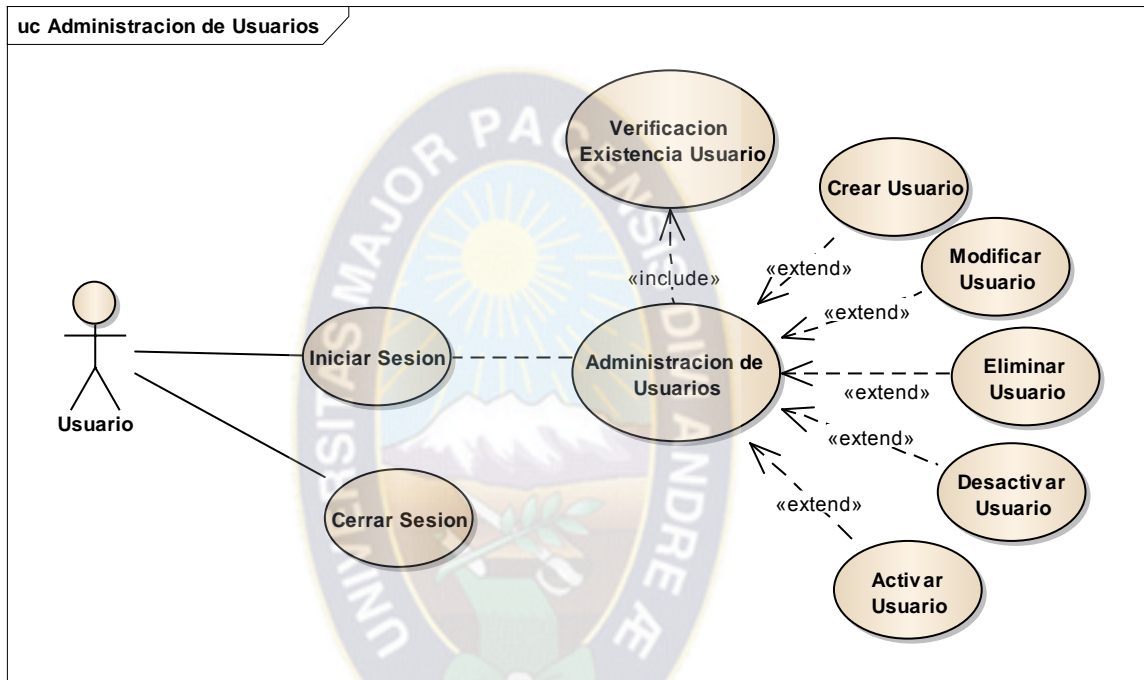


Figura 3.3 Diagrama Caso Uso Administración de Usuarios

Fuente (Elaboración Propia)

La tabla 3.6 describe con detalle todas las funciones que cumple el administrador en sistema

No. 2	Nombre de caso de uso: Administración de usuarios
Actor Principal: Administrador del Sistema	
Personal involucrado e intereses: Administrador del Sistema: Realiza la actividad solicitadas con el visto bueno del jefe de unidad o director de la institución	
Precondiciones: El Administrador del Sistema, debe haber iniciado sesión, autenticando su usuario y contraseña.	
Post-condiciones: Se registra la gestión de usuarios satisfactoriamente	
Referencias Cruzadas (Casos de Uso): Caso de uso Iniciar sesión	
<p>Flujo Básico:</p> <ol style="list-style-type: none"> 1. El Administrador del Sistema selecciona Administración de usuarios 2. El Sistema presenta: crear, actualizar, activar, desactivar o eliminar usuario 3. El Administrador del Sistema selecciona crear usuario o flujo Alternativo 2, o flujo Alternativo 3, o flujo alternativo 4, o flujo alternativo 5 o flujo alternativo 6. 4. El Sistema muestra la información a ser complementada 5. El Administrador del Sistema ingresa los datos solicitados y selecciona guardar. 6. El Sistema verifica que los datos sean correctos y los almacena. 	
<p>Flujos Alternativos:</p> <p>Flujo Alternativo 1:</p> <ol style="list-style-type: none"> 6.a El Sistema muestra que los datos ingresados no son correctos, regresa al paso 4. <p>Flujo Alternativo 2: Actualizar Usuario</p> <ol style="list-style-type: none"> 1. El Administrador del Sistema selecciona actualizar usuario 2. El Sistema pide los datos del usuario a actualizar 3. El Administrador del Sistema ingresa los datos solicitados 	

4. El Sistema verifica que el usuario exista
5. El Sistema indica que el usuario no existe y regresa al paso 2
6. El Sistema presenta la información del usuario solicitado
7. El Administrador del Sistema realiza los cambios respectivos y selecciona guardar
8. El Sistema muestra que los datos no son correctos y regresa al paso 7
9. El Administrador del Sistema confirma actualizar o flujo alternativo 6

Flujo Alternativo 3: Eliminar Usuario

1. El Administrador del Sistema selecciona eliminar usuario
2. El Sistema pide los datos del usuario a eliminar
3. El Administrador del Sistema ingresa los datos solicitados
4. El Sistema verifica que el usuario exista y que no haya realizado ninguna transacción.
5. El Sistema indica que el usuario no existe y regresa al paso 2
6. El sistema solicita confirmación de eliminación del usuario o flujo 6..

Flujo Alternativo 4: Desactivar Usuario

1. El Administrador del Sistema selecciona desactivar usuario
2. El Sistema pide los datos del usuario a desactivar.
3. El Administrador del Sistema ingresa los datos solicitados
4. El Sistema verifica que el usuario exista
5. El Sistema indica que el usuario no existe y regresa al paso 2
6. El Sistema presenta la información del usuario solicitado.
8. El Sistema solicita confirmación de desactivación del usuario.
9. El Administrador del Sistema confirma desactivación o flujo alternativo 6

Flujo Alternativo 5: Activar Usuario

1. El Administrador del Sistema selecciona activar usuario.
2. El Sistema pide los datos del usuario a activar.
3. El Administrador del Sistema ingresa los datos solicitados.
4. El Sistema verifica que el usuario exista y que se encuentre desactivado o bloqueado.
5. El Sistema indica que el usuario no existe y regresa al paso 2
6. El sistema solicita confirmación de activación del usuario.
7. El administrador confirma la activación o flujo 6.

Flujo 6: Cancelar operación

1. El Administrador del Sistema no confirma la operación
2. El Sistema cancela la operación

Tabla 3.6 Administración de Usuarios

Fuente Elaboración Propia

3.4.4.3 Diagrama de casos de uso Consulta médica

La figura 3.6 muestra la interacción del médico con el sistema al momento de atender a un paciente y su posterior elaboración de diagnóstico médico para ser almacenado en la base de datos para que la próxima consulta que tenga el paciente se pueda cargar el diagnóstico las referencias que se le dio.

La figura 3.4 muestra las opciones con las que cuenta el sistema para proceder cualquier médico con la atención médica al momento de apersonarse a los consultorios en la hora asignada que se sacó ficha en la especialidad requerida además el doctor tiene a disposición las opciones para elaboración del diagnóstico.

El doctor al concluir la consulta, tiene la posibilidad de realizar solicitudes de laboratorio, imagenología, fisioterapia, o la especialidad que vea conveniente, de acuerdo al diagnóstico que se maneje en ese momento. Las solicitudes también pueden ser realizadas para confirmar un diagnóstico. Para cada solicitud el sistema tiene preparadas diferentes pantallas con los formatos requeridos en cada especialidad.

Condiciones de ejecución.

La condición es que el doctor ya haya realizado la respectiva consulta al paciente y esté finalizada dicha consulta. En el caso de que el paciente tenga una referencia a una especialidad y no haya sido atendida, el doctor no podrá realizar otra.

Entrada

- Selecciona el tipo de servicio que realizará.
- Introducimos el texto o informe que llegará a la especialidad a la cual se está solicitando el servicio.
- En el caso de laboratorios e imagenología se podrá seleccionar el tipo de laboratorio o imagen que se solicitará, de un catálogo establecido por dicha unidad.
- Para cada solicitud, el doctor puede introducir un comentario o recomendación.

- Presionamos el botón solicitar y dicha solicitud será enviada a la unidad correspondiente.
- Para cada solicitud, el doctor puede introducir un comentario o recomendación.
- Presionamos el botón solicitar y dicha solicitud será enviada a la unidad correspondiente.

Resultado esperado.

Cuando la especialidad o unidad tenga resultados del paciente derivado, estos ya estarán a disponibilidad del doctor para poder continuar con el tratamiento o simplemente darlo de alta.

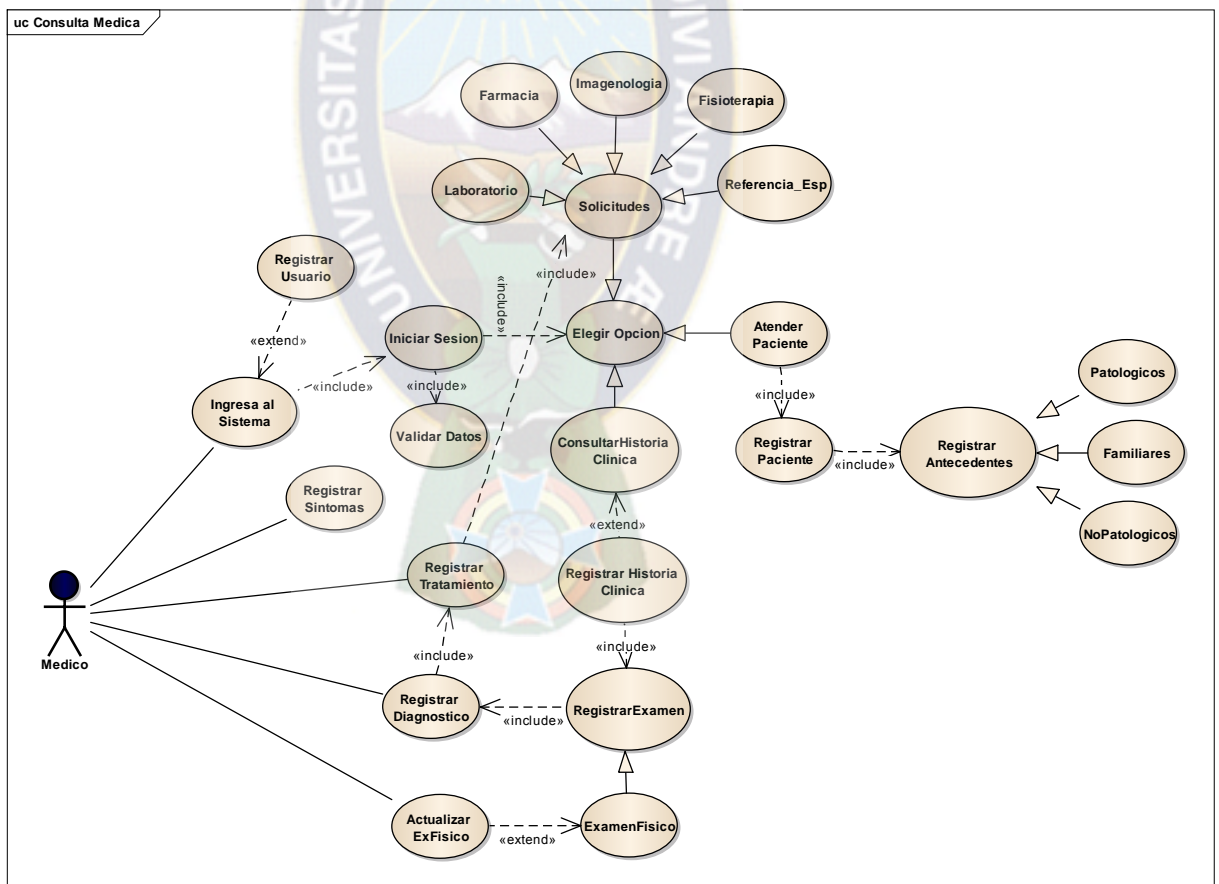


Figura 3.4 Diagrama Caso de Uso Consulta Medica

Fuente (Elaboración Propia)

La tabla 3.7 describe el funcionamiento que tienen las consultas médicas

Caso de Uso	Consulta Medica
Medico	Encargado de la consulta
Descripción	Se encarga de atender al paciente en el horario que saco su ficha
Flujo de Trabajo	<p>Ingresar al sistema</p> <p>Inicia sesión</p> <p>Elegir opción</p> <p>Registra diagnostico</p> <p>Registra tratamiento</p> <p>Registrar síntomas del paciente</p> <p>Actualizar examen clínico</p> <p>Registrar solicitudes</p>
Caso de uso	Ingresar al sistema
Actores	Medico
Descripción	El medico ingresa al sistema con el fin de atender a los pacientes agendados a la fecha
Caso de Uso	Verificar Iniciar sesión
Actores	Medico
Descripción	Se verifica la autenticación de médico.
Caso de Uso	Modificar Elegir opción

Actor	Medico
Descripción	El medico escoge la pestaña para realizar una nueva consulta o re consulta de acuerdo a los datos recabados por el paciente.
Caso de Uso	Registrar Diagnostico
Actor	Medico
Descripción	EL medico elabora el diagnóstico del paciente de acuerdo a los datos recabados.
Caso de Uso	Registra tratamiento
Actor	Medico
Descripción	El medico se encarga de elaborar el tratamiento del paciente de acuerdo al diagnóstico detectado.
Caso de Uso	Registrar síntomas del paciente
Actor	Medico
Descripción	El medico se encarga de registrar todos los síntomas que el paciente presenta al momento de ingresar el paciente a consultorios.
Caso de Uso	Actualizar examen clínico
Actor	Medico
Descripción	Una vez terminada la consulta del paciente el medico procede a actualizar la historia clínica.
Caso de Uso	Registrar solicitudes
Actor	Medico
Descripción	El medico tiene la opción de mandar a realizar exámenes de laboratorio si el

	paciente lo necesitara para detectar el problema que presenta
--	---

Tabla 3.7 Consulta medica
Fuente (Elaboración propia)

3.4.4.4 Diagrama de casos de uso Consulta especialidad

La figura 3.5 describe la interacción que existe entre el sistema y las especialidades con la que se cuenta y poder cargar los datos que se necesitan y pueda llevarse la consulta de una manera más óptima y no exista confusiones

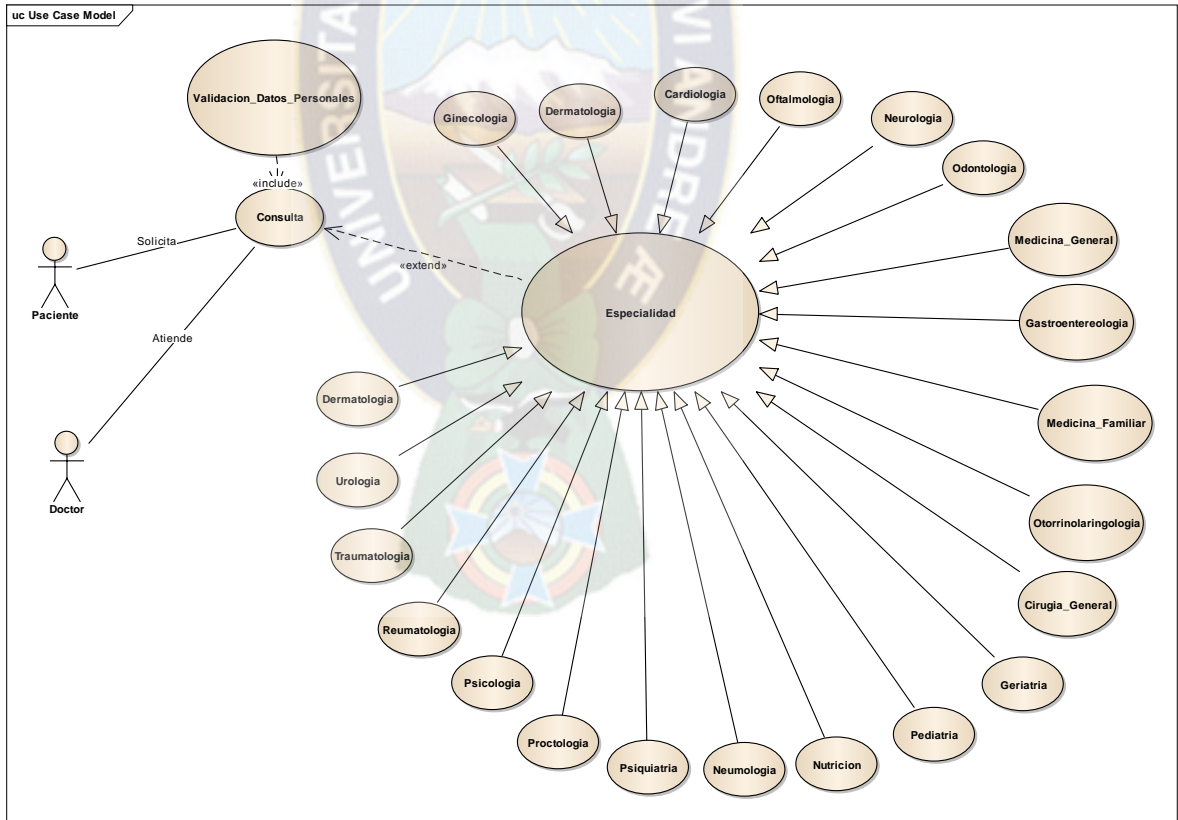


Figura 3.5 Diagrama de Caso de Uso Consulta especialidad
Fuente (Elaboración Propia)

La tabla 3.8 describe con detalle el inicio de consultas

Caso de uso	Inicia consulta
Actores	Medico
Descripción	El medico ingresa al sistema y verifica el listado de pacientes que se agendaron en su turno
Caso de Uso	Validación de datos del paciente
Actores	Medico
Descripción	El medico verifica los datos del paciente que ingresa al consultorio.
Caso de Uso	Especialidad
Actor	Medico
Descripción	El médico de la especialidad tiene un nuevo formulario para llenar y generar .un informe detallado de las solicitudes que requiere el paciente

Tabla 3.8 Consulta especialidad

Fuente (Elaboración Propia)

3.4.4.5 Diagrama de casos de uso Examen físico

La figura 3.6 describe la interacción de tres entidades doctor paciente y enfermera para el llenado del examen clínico que recibe cada paciente más que todo está enfocado en la primera consulta

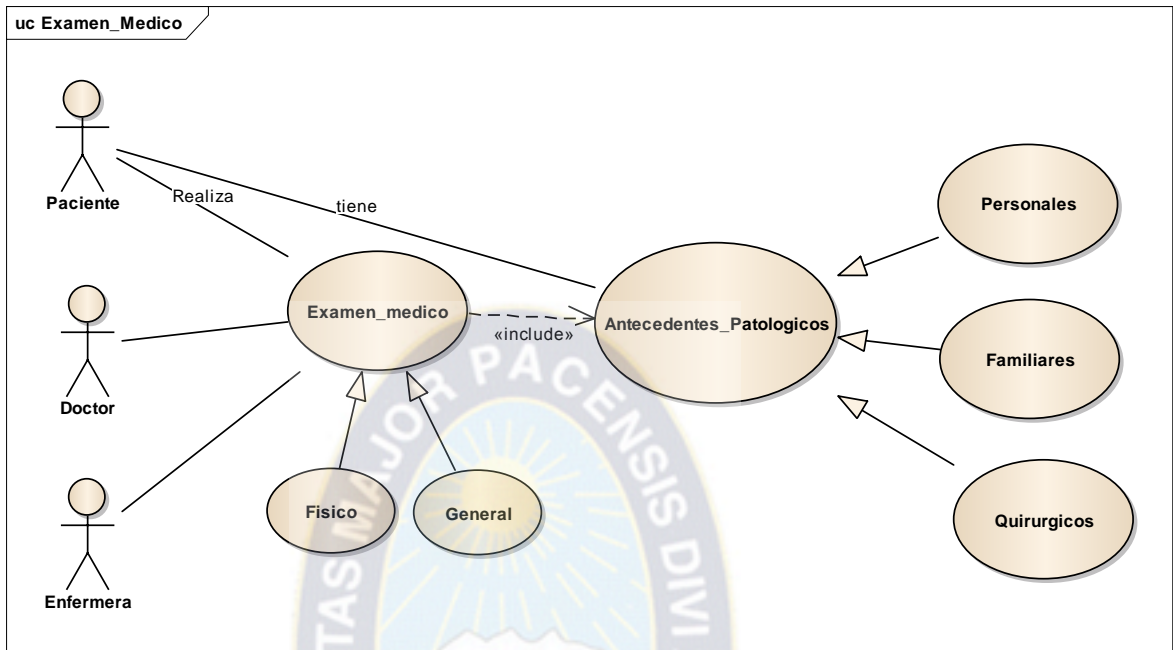


Figura 3.6 Diagrama de Caso de Uso Examen Físico
Fuente (Elaboración Propia)

La tabla 3.9 describe con detalle los exámenes médicos que se realizan al paciente

Caso de Uso	Examen Físico
Enfermera	Encargada de llenar el examen físico de los pacientes
Descripción	El examen físico se la realiza a los pacientes que vienen a consultorios por primera vez o están visitando las especialidades de medicina familiar y medicina general
Flujo de Trabajo	Inicia sesión Registra examen medico Registro de antecedentes patológicos Registro de antecedentes personales Registro de antecedentes familiares
Caso de uso	Inicia sesión

Actores	Enfermera
Descripción	La enfermera ingresa a los formularios de exámenes médicos para proceder a llenar y estén todos los datos que pueda requerir el medico en la consulta
Caso de Uso	Registro de examen medico
Actores	Médico, Enfermera y paciente
Descripción	La enfermera prosigue a llenar los formularios de examen físico y general para que los datos recabados por el paciente sean interpretados por los médicos
Caso de Uso	Registro de antecedentes patológicos
Actor	Medico
Descripción	El medico recaba información del paciente que viene a consultorio por primera vez llenando el formulario de antecedentes patológicos.
Caso de Uso	Registro de antecedentes no patológicos
Actor	Medico
Descripción	El medico recaba información del paciente que viene a consultorio por primera vez llenando el formulario de antecedentes no patológicos
Caso de Uso	Registro de antecedentes familiares
Actor	Medico
Descripción	El medico recaba información del paciente que viene a consultorio por primera vez llenando el formulario de antecedentes familiares

Tabla 3.9 Examen Físico
Fuente (Elaboración propia)

3.4.4.6 Diagrama de casos de uso Diagnostico

La figura 3.7 describe la interfaz de solicitud de exámenes clínicos en caso de ser necesitado para elaborar el diagnostico en caso de no saber cuál es el problema que ocasiona la anomalía y necesite de más exámenes clínicos como ser laboratorio, farmacia, imagenología, etc.

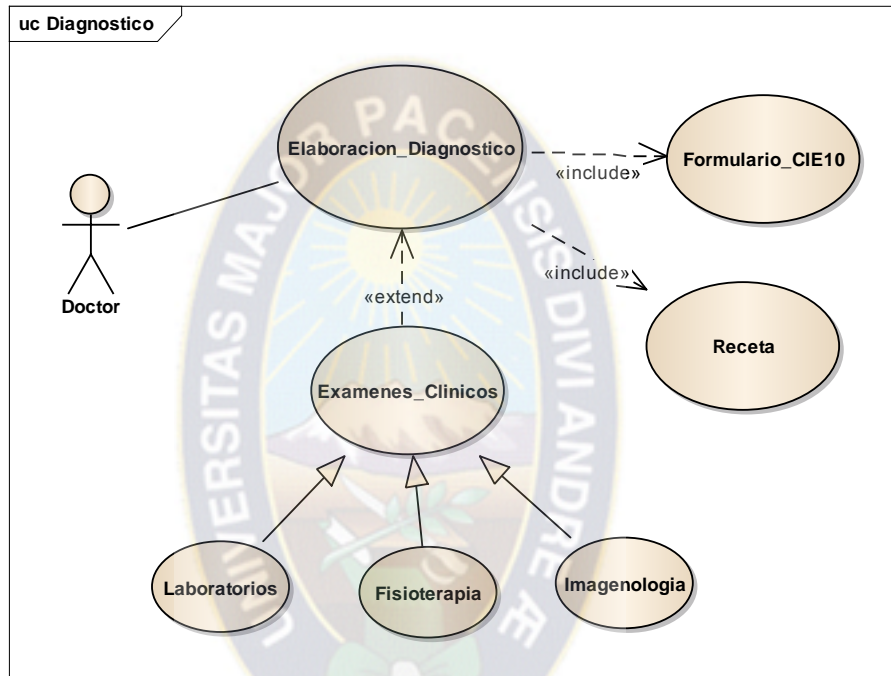


Figura 3.7 Diagrama de Caso de Uso Diagnostico
Fuente (Elaboración Propia)

La tabla 3.10 describe con detalle la elaboración de diagnósticos

Caso de Uso	Elaboración de Diagnostico
Medico	Encargada de llenar los diagnósticos de los pacientes agendados
Descripción	La elaboración del diagnóstico la realiza cada médico especialista dependiendo al tipo de consulta
Flujo de Trabajo	Elaboración de diagnostico

	Recabar información del formulario CIE-10 Registro de exámenes clínicos Registro de recetas
Caso de uso	Elaboración del diagnóstico
Actores	Medico
Descripción	El medico se encarga de elaborar el diagnostico de cada paciente que ingresa a consultorio con todos los síntomas que presenta el paciente
Caso de Uso	Recabar información del formulario CIE-10
Actores	Médico
Descripción	El medico consulta el formulario CIE-10 para diagnosticar al paciente.
Caso de Uso	Exámenes clínicos
Actor	Medico
Descripción	El medico solicita exámenes complementarios a los pacientes para tener un diagnóstico certero.
Caso de Uso	Registro de receta
Actor	Medico
Descripción	El medico elabora una receta con todos los medicamentos para su posterior mejora del paciente

Tabla 3.10 Elaboración diagnostico

Fuente (Elaboración propia)

3.4.5. Diagrama de Actividad

Es un diagrama de flujo del proceso multiproposito que se usa para modelar el comportamiento de sistema, muestra como las actividades fluyen y las dependencias

entre ellas. A continuación se detalla cada uno de los diagramas de actividades que se utilizarán.

3.4.5.1 Diagrama de actividad Ingreso al sistema

La figura 3.8 muestra la autenticación del sistema para el ingreso en donde se almacenará el registro y contará con una interfaz en donde se cargan los pacientes del día de acuerdo a fichaje

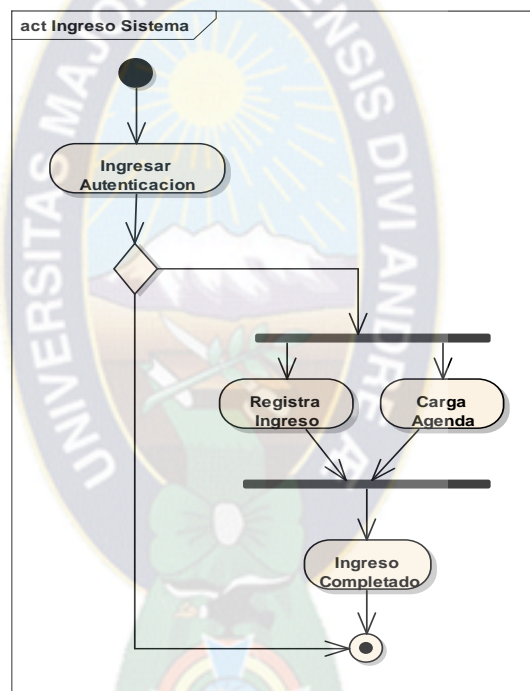


Figura 3.8 Diagrama de Actividad Ingreso al Sistema

Fuente (Elaboración Propia)

3.4.5.2 Diagrama de actividad Atencion al paciente

La figura 3.9 muestra la interacción del médico con el sistema para el llenado de historia clínica si fuera el caso.

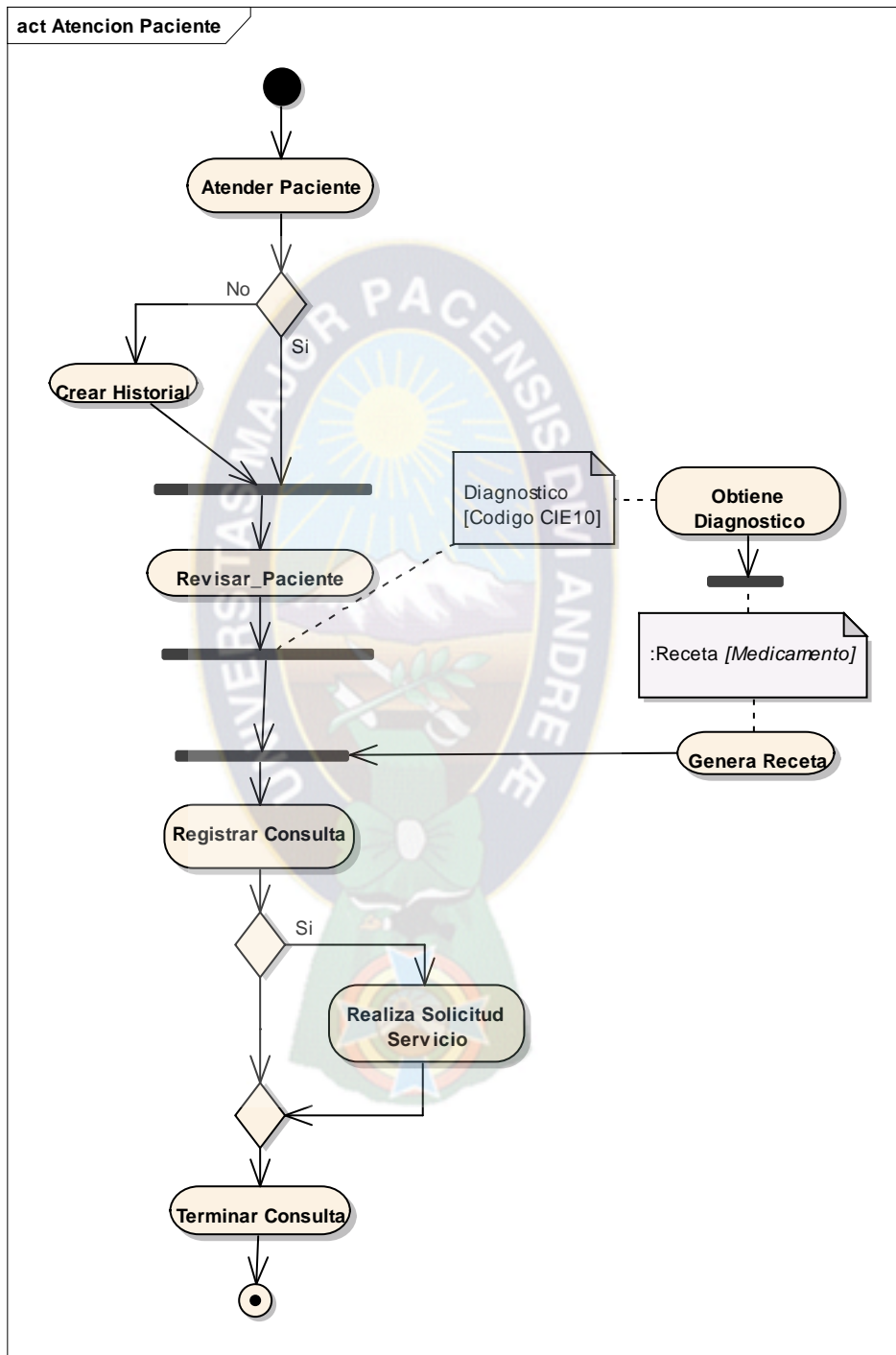


Figura 3.9 Diagrama de Actividad Atención Paciente
Fuente (Elaboración Propia)

3.4.5.3 Diagrama de actividad Consulta detallada

La figura 3.10 muestra la interacción completa que tiene el doctor con el sistema para la atención y ser almacenado después en la base de datos.

Descripción.

Nos introducimos en el sistema, en el caso del doctor, quien llama al paciente que tiene que recibir la atención médica. En caso de ser paciente nuevo, el doctor llena los datos generales médicos, con los cuales se considera abierta una historia clínica, posteriormente el doctor revisa al paciente y este ya está en condición de emitir un diagnóstico y de recetar al paciente o bien solicitar algún servicio, para estas actividades, el doctor tiene los catálogos necesarios, tanto de las enfermedades a diagnosticar, los medicamentos disponible y los formularios necesarios para realizar cualquier solicitud.

Condiciones de Ejecución.

Para la ejecución del caso de uso, el doctor tiene que tener pacientes agendados para ese día, la enfermera deberá haber tomado los signos vitales del paciente antes de la consulta.

Entrada.

- Seleccionamos al paciente que será atendido
- Pulsamos el botón para nueva consulta o evolución, para que el sistema identifique como guardará los datos.
- Aparece la pantalla de revisión por sistemas y el llenado de los signos vitales.
- Mostrará una interfaz con botones de siguiente y atrás, parecidos a programas de Windows, para seguir una secuencia en el llenado del historial clínico.
- En la pestaña de diagnóstico, se seleccionará una de las enfermedades, de acuerdo al catálogo de enfermedades.
- En la pestaña de medicación, se generará la receta con los medicamentos existentes en farmacia.

- Antes de finalizar una consulta, el sistema verificara los datos que se introdujo en la consulta, así como también preguntara si desea realizar alguna solicitud.
- Al pulsar el botón finalizar se retornará a la pantalla donde se encuentra la lista de los pacientes que deben ser atendidos.

Resultado Esperado.

El sistema nos mostrará mensajes que nos indicará si se llenó correctamente los datos de la consulta o si ocurrió algún error.

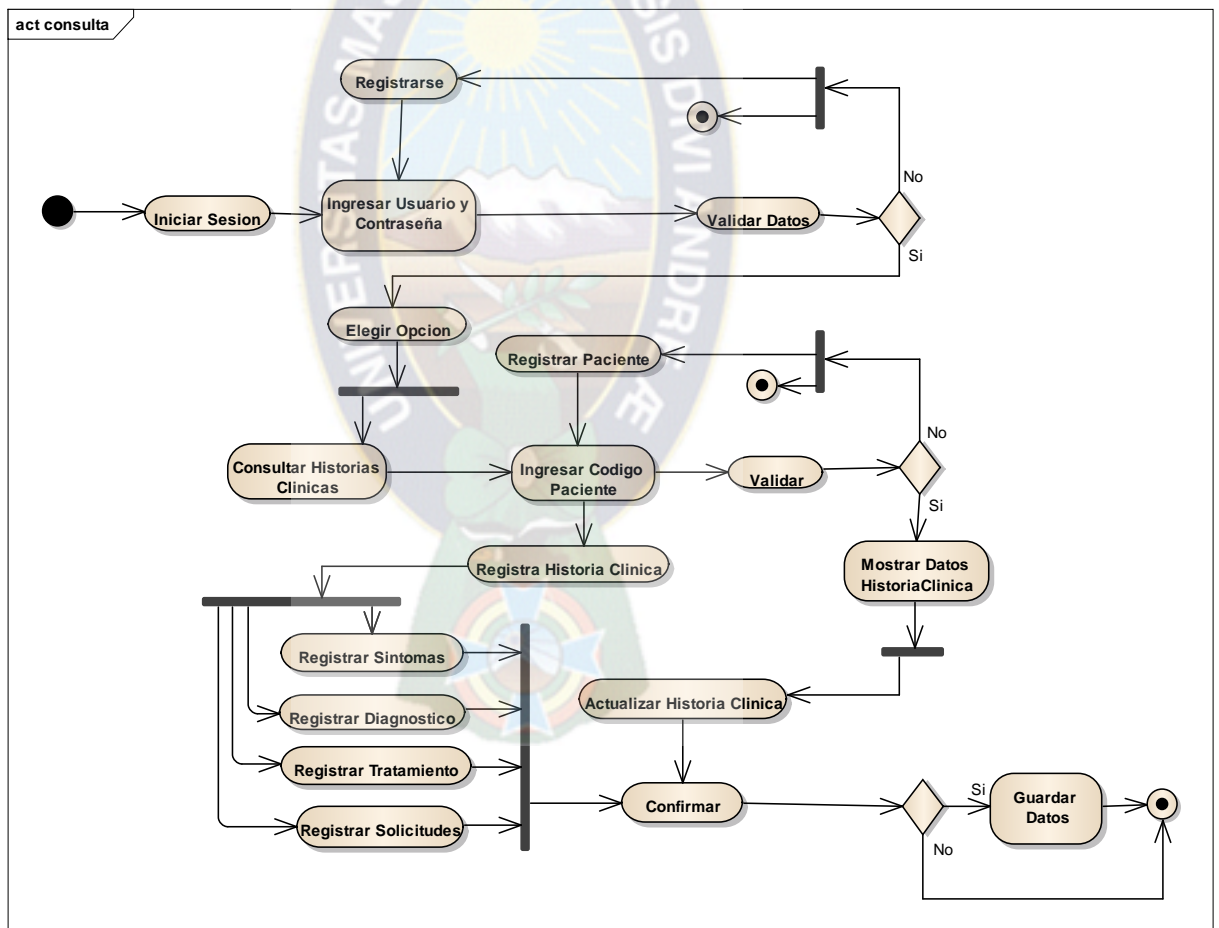


Figura 3.10 Diagrama de Actividad Consulta Detallada

Fuente (Elaboración Propia)

3.4.5.4 Diagrama de actividad Atención al paciente nuevo o re consulta

La figura 3.11 muestra el comportamiento que tiene el sistema cuando existe una re consulta y una nueva consulta para que exista un seguimiento de tratamiento si es que fuera el caso.

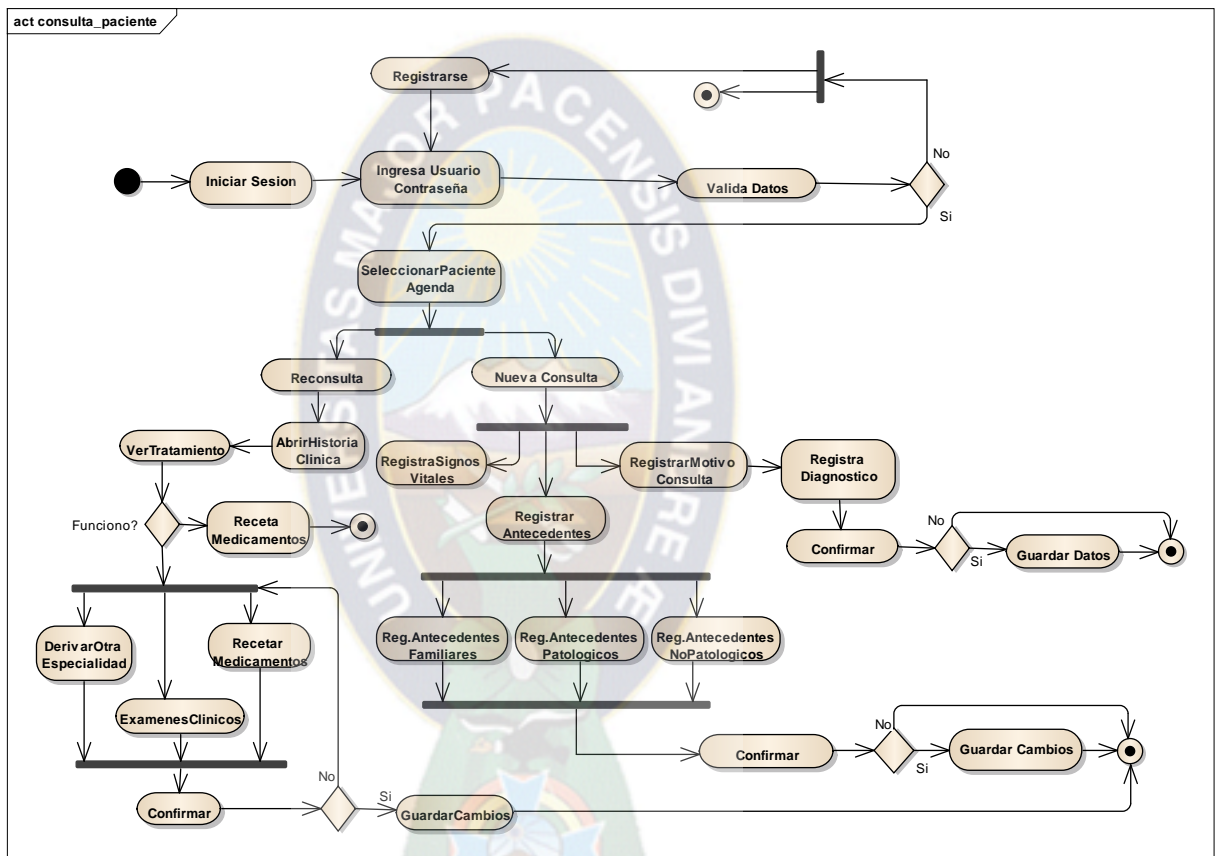


Figura 3.11 Diagrama de Actividad Atención paciente nuevo o re consulta

Fuente (Elaboración Propia)

3.4.6. Diagrama de Secuencia

3.4.6.1 Diagrama de Secuencia Inicio de sistema

La figura 3.12 describe los siguientes puntos

- Cuando el usuario del sistema se autentica correctamente se muestra la pantalla de bienvenida con su respectivo menú para la navegación.
- En este diagrama podemos visualizar que solo dos clases intervienen directamente con la pantalla del menú principal la clase de permisos y la clase de usuarios los cuales se pueden modificar la información que requiera el usuario a su conveniencia.
- Después de realizar cualquier modificación en las clases ya mencionadas se retorna automáticamente a la pantalla de bienvenida.

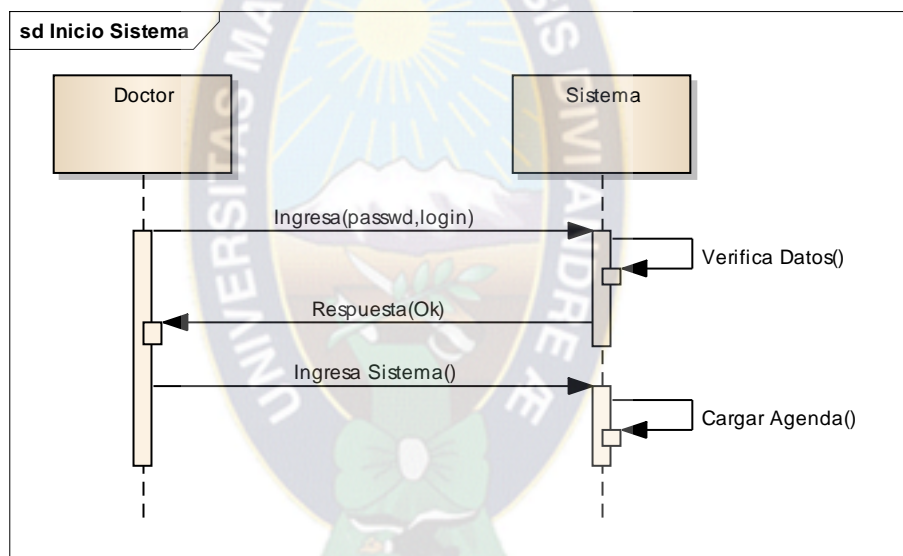


Figura 3.12 Diagrama de Secuencia Inicio Sistema

Fuente (Elaboración Propia)

3.4.6.2 Diagrama de Secuencia Revisión paciente

La figura 3.13 muestra la interacción que existe entre el paciente y el medico en los consultorios para el llenado de diagnósticos, actualización de historias clínicas elaboración de recetas y solicitud de exámenes complementarios para tener un diagnóstico certero.

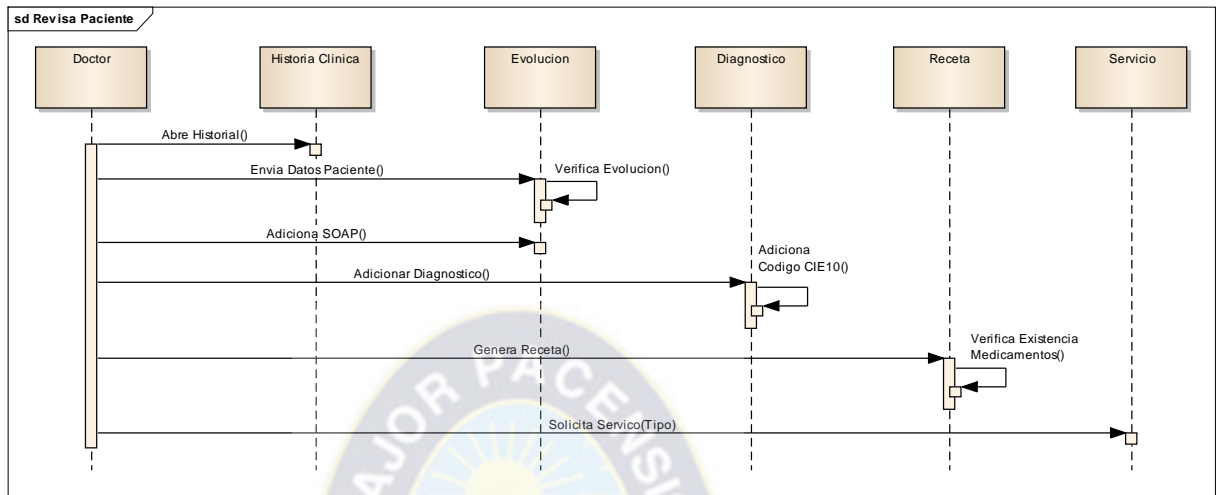


Figura 3.13 Diagrama de Secuencia Revisión Paciente

Fuente (Elaboración Propia)

3.4.6.3 Diagrama de Secuencia Privilegios administrador

La figura 3.14 describe los papeles importantes con las que cuenta el administrador para manipular el sistema de una manera coordinada con todos los módulos dando permisos a los usuarios de acuerdo al papel que desempeña en el hospital y las funciones que debe cumplir

Cuando el administrador va a ingresar al sistema, se verifica que los datos ingresados son los correctos para su acceso.

A continuación se presenta la interfaz de bienvenida al sistema con su respectivo menú para la navegación de este usuario.

Según la opción que seleccione del menú de usuario identificado se dirigirá a dicha pantalla.

En todas las pantallas de control de usuarios se encuentra la opción de búsqueda, dicha búsqueda es total por cada frase, línea, palabra o letra que se escriba en el buscador.

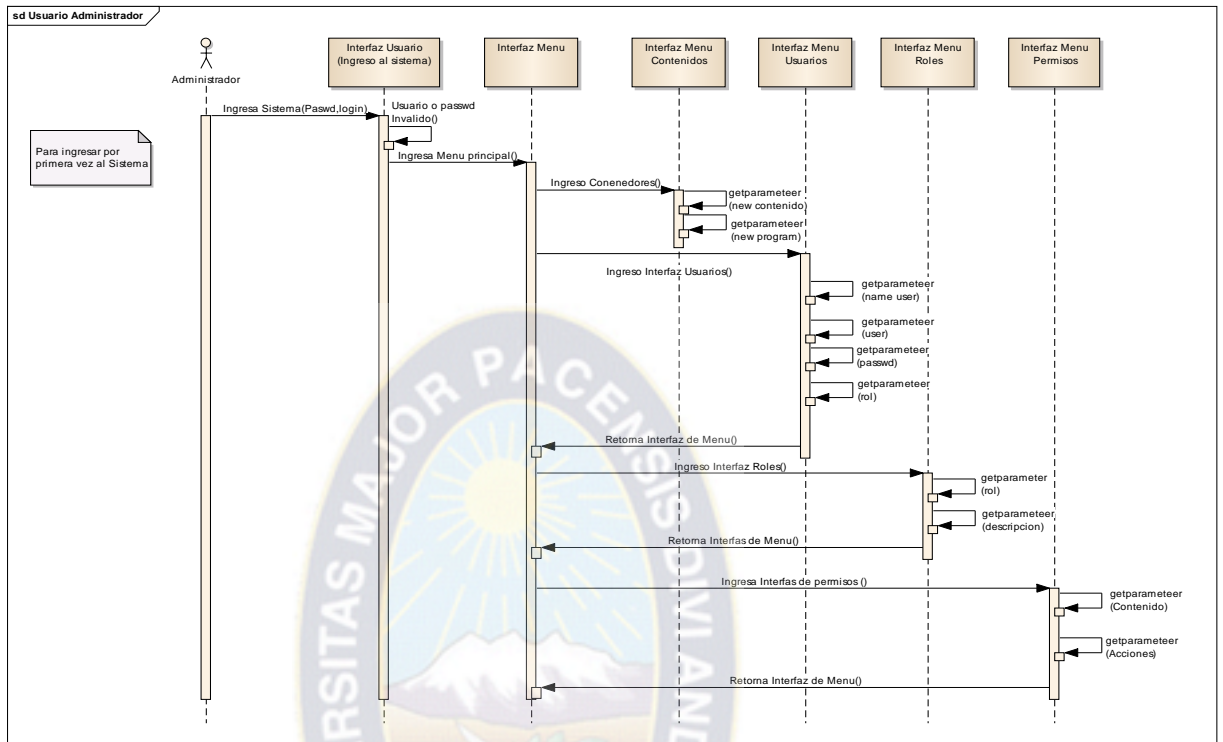


Figura 3.14 Diagrama de Secuencia Privilegios Administrador

Fuente (Elaboración Propia)

3.4.7. Diagrama de Clases

La figura 3.15 describe todos los diagramas de clases que se pudo identificar en el transcurso de la elaboración del sistema relacionándolo con los módulos de farmacia y admisión.

Las tablas identificadas en el modelado de consulta externa son:

- Paciente: en donde almacenamos todos los datos del paciente que fueron recabados en el modulo de afiliación para ser atendido en los consultorios de los médicos y cuenta con todos los beneficios que brinda el Seguro Social Universitario de La Paz como ser :
 - Farmacia
 - Imagenología

- Laboratorios
 - Trabajo social
 - Internación
 - Emergencia
 - Cirugías
- Doctor: es una clase muy importante que fue modelado en el modulo de admisión para modelar las agendas de los médicos y tengan un control de todo lo que sucede en cada consulta, el doctor se encarga de atender la consulta que solicito el paciente para ser diagnosticado.
 - Examen físico: el examen físico se la realiza a todos los pacientes que consultan las especialidades de:
 - Medicina general.
 - Medicina familiar.
 - Receta: La receta es elaborado por el médico que está a cargo de la consulta para proporcionar al paciente y de esta forma el paciente tenga una mejora en el transcurso de los días.
 - Antecedentes familiares: los antecedentes familiares son recabados por los pacientes en una entrevista con el médico tratante para llenar su historia clínica y tener siempre en cuenta cada antecedente para indagar en los diagnósticos.
 - Antecedentes patológicos: de igual manera que los antecedentes familiares estos datos que proporciona el paciente al médico son muy importantes que deben figurar en la historia clínica del paciente.
 - Antecedentes no patológicos: de igual manera que los antecedentes familiares se debe almacenar los datos que proporcione el paciente en la consulta en la historia clínica.
 - Especialidad: Al momento de sacar ficha médica para ser atendido es muy importante la especialidad que se escogió para que de esta manera se pueda generar un agenda para cada especialidad y no exista confusiones en los consultorios.

- CIE-10: La clasificación internacional de enfermedades (CIE-10) nos describe todas las enfermedades establecidas con la que los médicos diagnostican al paciente la anomalía que presenta y su posterior curación, este formulario es muy importante para la elaboración de diagnósticos médicos.
- Solicitudes complementarias: Las solicitudes complementarias son de vital importancia en cualquier consultorio médico para tratar al paciente y buscar su pronta mejora.

Las solicitudes complementarias con la que cuenta el seguro son:

- Farmacia: es una entidad muy importante donde se almacena la cantidad de medicamentos que fue recetado al paciente y de esa manera tener un control. Para esta parte se trabaja conjuntamente con el módulo de farmacia para ver el stock de medicamentos que están disponibles y de esta manera no exista errores como medicamento con bajo stock.
- Laboratorio se solicitan exámenes de laboratorio en caso de que el paciente requiera de algún examen para lo cual se trabaja con el módulo de laboratorios para solicitar dichos exámenes.
- Imagenología es una entidad para mandar a realizar a los pacientes para detectar la anomalía que está presente y buscar el tratamiento ideal para mejorar del paciente.
- Junta médica es una entidad muy importante en la que se guardan todas las juntas médicas para hablar de la situación del paciente en donde se requiere más de un médico para ver que sucede con el paciente en las próximas horas.
- Internación la internación del paciente se la autoriza viendo el estado en la que se encuentra y procede a trabajar con el módulo de Internación.
- Trabajo social es una entidad en donde se almacena los datos de los pacientes que requieran de un apoyo.
- Baja médica se la realiza después de un estudio que realiza el doctor y determina que el paciente necesita guardar reposo.

- Fisioterapia se la realiza a todos los pacientes que requieran para su mejora.
- Referencia es un punto muy importante , según el diagnostico del médico ve la necesidad de referenciar a otra especialidad para ser tratado de acuerdo a todos los exámenes solicitados para que el otro médico tenga el control del paciente

3.4.8 Modelo Entidad Relación

La figura 3.17 muestra en modelo entidad relación para observar cómo se construyó el armado de la base de datos para su posterior uso e integración con los demás módulos.



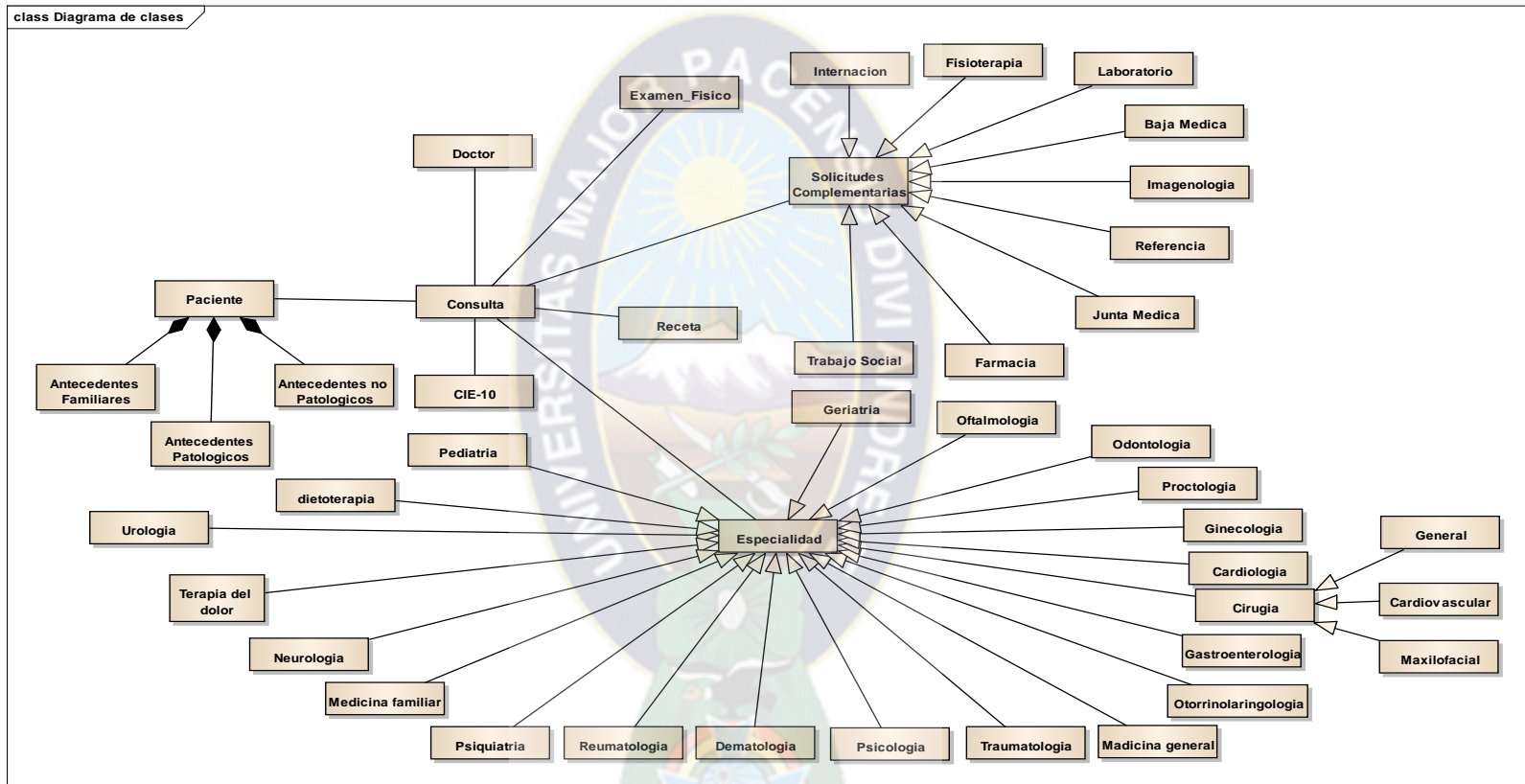


Figura 3.15 Diagrama de Clases

Fuente (Elaboración Propia)

La figura 3.15 visualiza todos los métodos que cuenta cada clase

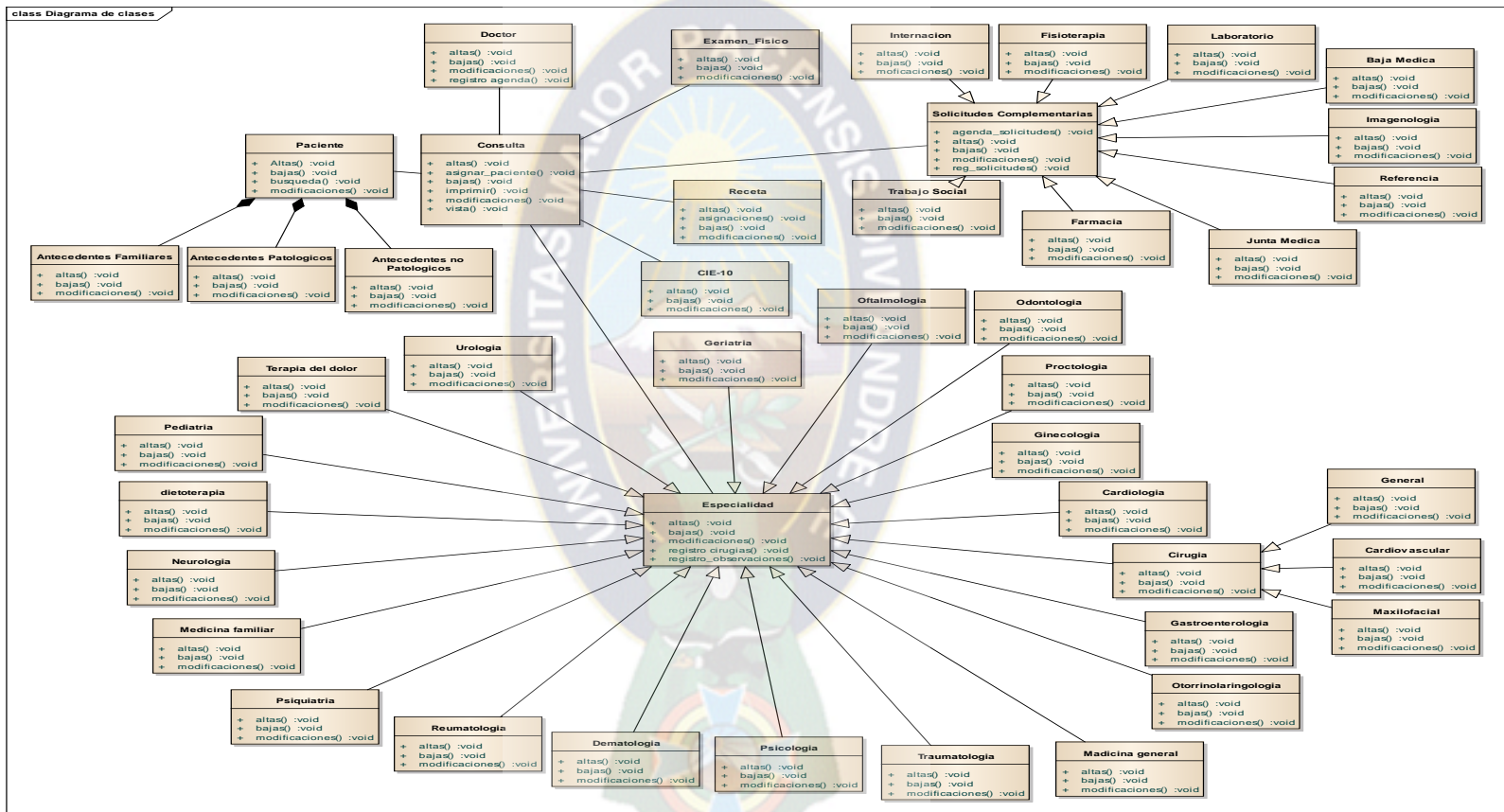


Figura 3.16 métodos de las clases

Fuente (Elaboración Propia)

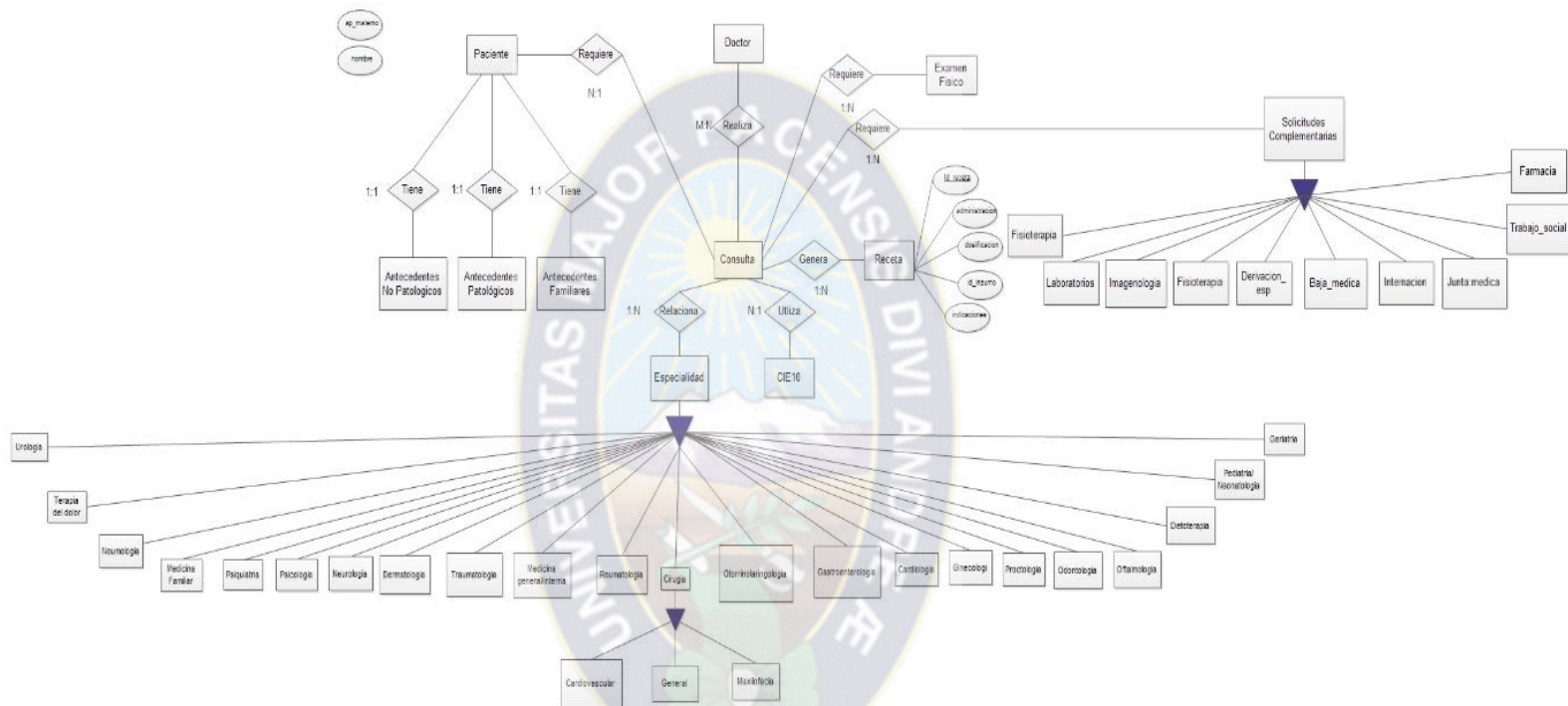


Figura 3.17 Modelo Entidad Relación

Fuente (Elaboración Propia)

3.4.9 Modelo Relacional

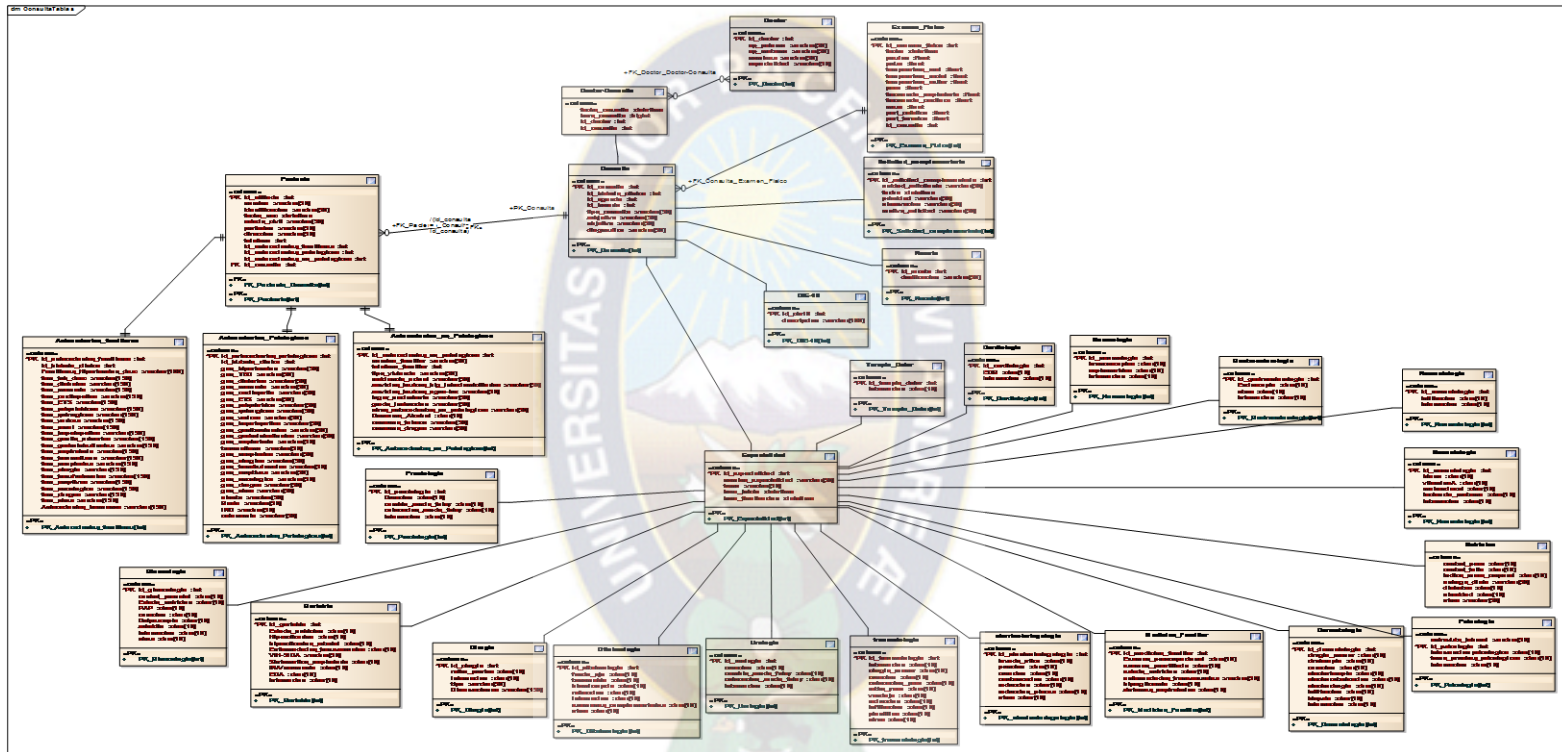


Figura 3.18 Modelo Relacional
Fuente (Elaboración Propia)

La figura 3.18 describe el modelo relacional que fue elaborado a partir del modelo entidad relación y la cardinalidad que existe entre las entidades para identificar las llaves primarias y foráneas de las base de datos y poder realizar consultas con el fin de generar un filtro de búsquedas un registro de los diagnósticos detectados a los pacientes y el control de los medicamentos que se recetan a los pacientes

3.4.10 Diagramas de Colaboración

- Cuando el usuario del sistema se autentifica correctamente se muestra la pantalla de bienvenida con su respectivo menú para la navegación
- En este diagrama podemos visualizar que solo dos clases intervienen directamente con la pantalla del menú principal la clase de permisos y la clase de usuarios, las cuales se pueden modificar la información que requiera el usuario a su conveniencia.
- Después de realizar cualquier modificación en las clases ya mencionadas se retorna automáticamente a la pantalla de bienvenida

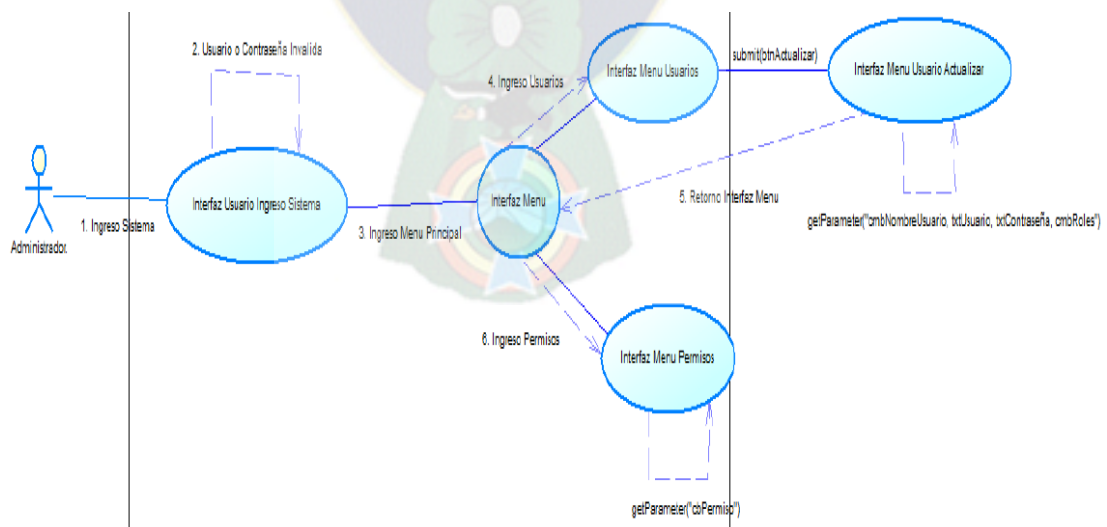


Figura 3.19 Diagrama de Colaboración Inicio Sesión

Fuente (Elaboración Propia)

3.4.11. Diseño de Seguridad

3.4.11.1. Diseño de niveles de acceso

Los niveles de acceso dentro de Gehma, se establecerán por medio de perfiles de usuarios, estos perfiles permitirán tener acceso a algunas o todas las funciones para el sistema. Los niveles a utilizar se pueden clasificar de la siguiente manera:

NIVEL 1. Administrador del Sistema:

Posee todos los privilegios, posee acceso a la creación y mantenimientos de usuarios del sistema, permite el acceso a todos los objetos de la base de datos, posee la capacidad de generación de todos los tipos de reportes, así como la modificación de cualquier tipo de registros, es el responsable de la seguridad del sistema, de las copias de respaldo y el óptimo funcionamiento del mismo.

NIVEL 2. Operador Medico.

Posee privilegios para actualizar y consultar cualquier expediente, realizar controles de seguimiento médico, rehabilitación y psicológico, además puede asignar altas a los paciente, no posee privilegios de eliminar o modificar ningún registro.

NIVEL 3. Operador Enfermera

Posee los privilegios para actualizar y consultar cualquier expediente, realizar control de asistencias a terapias, además de asignar altas a los pacientes, no posee privilegios de eliminar o modificar ningún registro.

3.4.11.2 Diseño de seguridad de datos, hardware y software

La seguridad del sistema comprende todos aquellos aspectos que le permitirán al sistema su funcionamiento de manera regular, estable y segura. Dichos aspectos se encuentran relacionados con elementos de confidencialidad, integridad de datos, autorizaciones, entre otros.

Seguridad de datos:

- Deberán realizarse copias de respaldo (backups) de la base de datos al menos una vez a la semana, esto debido al volumen de datos que se maneja y el grado de importancia de estos.
- El acceso y modificación de tablas de catálogos deberá ser realizado por el administrador de la base de datos.

Seguridad de Hardware:

- Los servidores están ubicados en un lugar de acceso restringido
- El lugar físico donde se ubicaran tanto medios de almacenamiento como servidores deberá estar libre de humedad y con la temperatura adecuada.
- Deben asignarse responsables sobre el manejo físico de los servidores y medios de almacenamiento tanto para su limpieza como para posibles cambios de ubicación física.
- Los cables de comunicaciones deberán ser colocados lejos de lámparas u otros objetos con campos electromagnéticos que puedan alterar la señal en transmisión.

Seguridad de Software:

- Se realizara una autenticación de usuarios para acceder a Gehma, mediante usuario y contraseña
- Las contraseñas de los usuarios deberán ser cambiadas por lo menos una vez cada dos meses para los operadores y una vez por mes para el administrador, lo anterior para garantizar la seguridad del acceso al sistema así como también prevenir la filtración de contraseñas en el lugar de trabajo, las cuales deberán incluir una combinación de letras mayúsculas, minúsculas, números y caracteres especiales además deberá de poseer una longitud mínima de 6 caracteres máximo 10.

- La asignación de perfiles para un usuario será realizada por el administrador del sistema.
- La bitácora de transacciones registrará los principales movimientos realizados dentro del Gehma, almacenando para ello lo siguiente: login de usuario, fecha, hora y tipo de transacción realizada.

3.5 Implementación

Diseño de Interfaz

La interfaz presenta una visión del modelo de usuario, proporcionando una situación que se elabora iterativamente para volverse un conjunto de clases de interfaz, el cual estará basado en dos reglas:

- Dar el control al usuario
- Construir una interfaz consecuente

Precisamente, uno de los requerimientos no funcionales es el control de acceso, en la figura 3.20 se visualiza la interfaz de autenticación, permitiéndole al usuario ingresar al sistema, digitando el nombre de usuario y la contraseña correspondiente que son únicos para cada usuario.



Figura 3.20 Inicio de sesión

Fuente: [Elaboración Propia]

La interfaz de usuarios nos muestra la opción para crear nuevos usuarios, eliminar, modificar y buscar como muestra la figura 3.21 para dar acceso al sistema y no exista ningún tipo de problemas.



Figura 3.21 Administración de Usuarios

Fuente: [Elaboración Propia]

La interfaz de roles nos permite crear roles para poder tener acceso al sistema teniendo limitaciones de acuerdo al rol que se le asignado y de acuerdo al rol se pueda limitar el sistema para no tener infiltración ajena como muestra la figura 3.22. Además la interfaz cuenta con la opción de búsqueda eliminación y modificación de roles.



Figura 3.22 Administración de Roles

Fuente: [Elaboración Propia]

La interfaz de permisos nos da la opción para limitar a ciertos usuarios el ingreso al sistema, además de dar ciertas acciones como actualizar, eliminar, ver e imprimir formularios que ya fueron llenados como muestra la figura 3.23.

Nombre y/o descripción	Tipo	Activo	Acciones habilitadas				
			Nuevo	Editar	Eliminar	Imprimir	Ver
convenios [convenios]	Contenido	<input type="checkbox"/>					
vigencia [vigencia]	Programa	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Utilitarios [Utilitarios del sistema]	Contenido	<input type="checkbox"/>					
Cambiar password [Cambiar contraseña]	Programa	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Administración [Administración general del sistema]	Contenido	<input type="checkbox"/>					
Roles [Administración de roles]	Programa	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Usuarios [Administración de usuarios]	Programa	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Contenidos [Administración de contenidos]	Programa	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Figura 3.23 Administración de permisos

Fuente: [Elaboración Propia]

SSU - SEGURO SOCIAL UNIVERSITARIO

USUARIO: [tarifa - Jorge Tarifa] ESPECIALIDAD: **Practología**

Consulta_Externa
S.O.A.P
Antecedentes
Exámenes_Complementarios
Examen_Fisico
Practología
Utilitarios
Cambiar password

Grabar ANTECEDENTES FAMILIARES

Antecedentes Familiares:

- Enf. Renal
- Epatopatías
- Enf. genitourinarias
- Enf. Gastrointestinales
- Enf. Respiratorias
- Traumatismos
- Neoplasias
- Alérgicos
- Transfusionales
- Chagas
- Otros
- Hipertensión
- TBC Pulmonar
- Diabetes
- Patología Mamaria
- Cardiopatías
- ETS
- Psiquiátricos
- Quirúrgicos
- Varices
- Oncológicos
- HipertensiónEruptivos

Figura 3.24 Antecedentes Familiares

Fuente: [Elaboración Propia]

© 2013 Seguro Social Universitario

Figura 3.25 Antecedentes no patológicos

Fuente: [Elaboración Propia]

© 2013 Seguro Social Universitario

Figura 3.26 Antecedentes Patológicos

Fuente: [Elaboración Propia]



Figura 3.27 Administración Contenidos

Fuente: [Elaboración Propia]



Figura 3.28 Inicio de Consulta

Fuente: [Elaboración Propia]

4. SEGURIDAD Y CALIDAD DE SOFTWARE

4.1 Políticas de Seguridad.

Para el Proyecto se propuso las siguientes recomendaciones de seguridad, basadas en lo requerimientos:

- El servidor de la Base de Datos debe estar de preferencia en un lugar físico seguro.
- Los usuarios debe cambias de contraseña de forma periódico para evitar suplantaciones.
- por semana se sugiere por el gran volumen de información que se está registrando constantemente.
- Es recomendable tener un antivirus debidamente actualizado pare evitar la intromisión de software malicioso.
- Control adecuado de la asignación de usuarios de acuerdo a su cargo.

4.2 Criterios de Seguridad

La seguridad consiste en métodos y procedimientos que contribuyan al resguardo de los datos, ya que la mayoría de los daños que pueden ocurrir se da sobre la información almacenada, por lo cual es importante tener muy en cuenta los criterios necesarios para la seguridad del sistema.

4.2.1 Seguridad en la red

El sistema desarrollado funciona en la intranet, por lo cual debemos considerar el Firewall (cortafuego), el cual trata de aplicaciones destinadas a prevenir la intromisión de elementos no deseados de la red a la máquina.

4.2.2 Seguridad de la aplicación

- **Control de usuario:** Es necesario tener un control de los usuarios que ingresan al sistema, para tener el sistema protegido de personas ajenas al seguro negándoles el acceso, para este fin el sistema tiene un proceso de identificación y autenticación..
- **Control de accesos por tipo de usuario:** no todos los usuarios tienen los mismos privilegios, y de acuerdo al tipo de usuario el sistema habilitara diferentes opciones.

4.2.3 Seguridad de la base de datos

- **Seguridad Lógica** Para mantener la seguridad lógica es necesario evaluar los controles de acceso de los usuarios a las plataformas de procesamiento y a los datos que estos manejan con el propósito de evitar irregularidades que afecten la confidencialidad, exactitud y disponibilidad de la información.
- **Seguridad Física** Para mantener la seguridad física de la información que se maneja se deben realizar copias de seguridad de la base de datos de manera periódica, para evitar una pérdida inesperada de la información.

4.3 Métricas de Calidad del Proyecto.

Los parámetros utilizados para medir la calidad del proyecto son:

4.3.1 Funcionalidad

Para medir la funcionalidad se utilizarán las métricas orientadas a la función.

Para calcular los puntos de función (PF) se utiliza la siguiente relación:

$$PF = Cuenta\ total = x[x + \min(y) \times Fi]$$

Dónde:

PF: Medida de Funcionalidad entregada a la aplicación como valor de normalización.

CT: La suma de los siguientes datos.

X: Nivel de confiabilidad del sistema.

Y: Nivel de significación de error

Fi: Valores de complejidad.

Se debe determinar cinco características de dominio de información las cuales son:

- Número de entradas de usuario.
- Número de salidas de usuario.
- Número de peticiones del usuario.
- Número de archivos.
- Numero de interfaces externas.

La determinación del valor de complejidad es subjetiva. A continuación se detalla los datos necesarios para ubicar los parámetros anteriormente mencionados.

Entrada de Usuario

- Panel de Registro de usuarios.
- Panel de Registro de Antecedentes Clínicos.
- Panel de Registro de Examen Físico.
- Panel de Registro de Signos Vitales.
- Panel de Registro del S.O.A.P.
- Panel de Registro de Nuevas Recetas

- Panel de Registro de Exámenes complementarios.
- Panel de Registro de Interconsultas

Salida de Usuario.

- Listado de pacientes atendidos por día.
- Reportes de los asegurados agendados atendidos según la especialidad.
- Reporte de los antecedentes de los pacientes.
- Listado de CIE10

Parámetros de medición	Número	Factor de Ponderación			
		Simple	Medio	Complejo	
Número de entradas de usuario	10	4	5	7	40
Número de salida de usuario	9	5	6	8	45
Número de peticiones de usuario	5	3	4	6	15
Número de archivos	20	10	13	18	200
Número de interfaces externas	4	5	7	10	20
Cuenta TOTAL					320

Tabla 4.1 Evaluación de funcionalidad

Fuente: [Elaboración Propia]

Los factores de ajuste responden en el rango de (0 - 5) de acuerdo a la siguiente tabla:

Escala	Sin Importancia	Incidental	Moderado	Medio	Significativo	Esencial
Factor (Fi)	0	1	2	3	4	5

Tabla 4.2 Factores de ajuste

Fuente: [Elaboración propia]

Nro.	Cuestionario	Factor
1	¿Requiere el sistema copias de seguridad y de recuperación fiable?	5
2	¿Se requiere comunicación de datos?	5
3	¿Existen funciones de procesamiento distribuidos?	4
4	¿Es crítico el rendimiento?	4
5	¿Se ejecutara el sistema en un entorno existente y fuertemente utilizado?	5
6	¿Requiere entrada de datos interactiva?	4
7	¿Requiere la entrada de datos interactiva que las transacciones de entrada se lleven a cabo sobre múltiples pantallas u operaciones?	3
8	¿Se actualizan los archivos maestros de forma interactiva?	4
9	¿Son complejas las entradas, las salidas, los archivos o las	5

	peticiones?	
10	¿Es complejo el procesamiento interno?	5
11	¿Se ha diseñado el código para ser reutilizable?	5
12	¿Están incluidas en el diseño la conversión y la instalación?	5
13	¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?	5
14	¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizada por el usuario?	5
	TOTAL	64

Tabla 4.3 Cuestionario de funcionalidad

Fuente: [Elaboración Propia]

Posteriormente reemplazamos en la ecuación PF y se tiene lo siguiente:

$$PF = 320 \times [0.65 + 0.01 \times 64]$$

El punto de función real obtenido es = 412.8

La función con nivel de confianza de 100% será:

$$PF_{esperado} = 320 \times [0.65 + 0.01 \times 70]$$

$$PF_{esperado} = 432$$

Calculando el porcentaje de funcionalidad, se tiene:

$$\%PF = \frac{PF_{real}}{PF_{esperado}} = 0.95$$

Para la medición de la calidad se realizaron dos pruebas y se ha notado que se obtuvo una mejora de 1.5% a la de la anterior prueba como se muestra en la tabla:

Nro.	Fecha	Valor
1	27-11-13	92.6 %
2	2-12-13	95 %

Tabla 4.4 Comparación de Pruebas

Fuente: [Elaboración Propia]

Al obtener un valor más alto en la segunda prueba se demuestra que se mejoró el sistema. Por lo tanto concluimos que el sistema de información tiene una funcionalidad del 95%.

4.3.2 Mantenibilidad

Entre las métricas diseñadas para el mantenimiento del software, está el estándar IEEE 982, el cual sugiere un índice de madurez del software que proporciona una estabilidad del producto.

Se determina la siguiente información.

MT: Número de módulos de la versión

FC: Número de módulos de la versión actual que se han cambiado.

Fa: Número de módulos de la versión actual que se han dañado.

Fd: Número de módulos de la versión que se han borrado con la versión actual.

Una vez que se cuenta con la información mencionada se calcula el índice de madurez del software de la siguiente manera:

$$IMS = \frac{[MT - (Fa + Fc + Fd)]}{MT}$$

Entonces tendremos

$$IMS = \frac{5 - (1 + 0 + 0)}{5}$$

$$IMS = 0.8 = 80\%$$

Por lo tanto se concluye que el sistema tiene un 80% de mantenibilidad.

4.3.3 Usabilidad

Para medir la usabilidad, existen diversos métodos, en nuestro caso utilizamos un cuestionario, al cual se le asignó un valor, y los resultados son los obtenidos en la siguiente tabla.

Nro.	Pregunta	Evaluación
1	¿Se ha satisfecho todos los requerimientos establecidos?	5
2	¿Le resulta fácil de recordar las órdenes y aprender las operaciones básicas?	4
3	¿Considera usted que es una herramienta útil?	5

4	¿Presenta la suficiente ayuda durante el tiempo que accede al sistema?	4
5	¿Cómo considera el formato de las salidas que genera el sistema?	5
6	¿El sistema tiene seguridad necesaria?	5
7	¿Los reportes que presenta son los suficientemente representativos?	5
8	¿Los reportes ayudan a la toma de decisiones?	5
9	¿Cómo le parece el tiempo de ejecución de tareas?	4
	Sum (xi) = Total	42

Tabla 4.5 Cuestionario en el aspecto de usabilidad

Fuente: Elaboración Propia

Ahora como ya tenemos la variable sum (xi) podemos usar la siguiente ecuación:

$$FU = \frac{\left[\left(\frac{42}{9}\right) \times 100\right]}{5}$$

$$FU = 93.33 \%$$

Se realizó dos pruebas y se ha notado que se mejoró un 2.23% como se muestra en la tabla

Nro.	Fecha	Valor
1	27-11-13	86.67 %

2	6-12-13	93.33 %
---	---------	---------

Tabla 4.6 Comparación de Pruebas

Fuente: Elaboración Propia

La mejora en el valor de las prueba muestra que el sistema presenta un porcentaje mejor de usabilidad, teniendo como resultado un 93.33%.

4.3.4 Portabilidad

La portabilidad de un sistema es un factor muy importante y necesario, debido a que el programa se instala en diferentes sistemas de hardware y software.

- **Portabilidad del lado del servidor:**

A nivel Sistema Operativo, es portable en: Microsoft Windows 2000, Windows XP, Windows 2003 Server o superiores.

A nivel del servidor de aplicaciones, es portable bajo Apache 2.2 o superiores-A nivel Servidor de Base de Datos es portable bajo SQL SERVER, MySQL, PostGress, Oracle, etc.

A nivel hardware es portable bajo las siguientes características mínimas del servidor Pentium III como mínimo, Memoria RAM de 1Gb, Procesador de 1Ghtz. Disco Duro de 10 GB.

- **Portabilidad del lado del cliente:**

A nivel de Sistema Operativo, el ordenador del cliente funciona bajo cualquiera de los siguientes sistemas operativos: Windows 2000, Windows XP, Windows 2003 Server, Windows Vista y Windows 7

A nivel Browser, es portable bajo los siguientes navegadores: Internet Explorer, Mozilla Firefox, Chrome, etc.

A nivel hardware es portable bajo el equipo del cliente que cuenta con las siguientes características: Pentium III, Memoria RAM de 128 Mb, Procesador de 1.8 Ghtz, Disco duro de 1 Gb o superior.

4.3.5 Confiabilidad

Al momento de realizar las pruebas del proyecto se detectaron diversas fallas. A continuación se muestra una tabla con los resultados obtenidos.

Tiempo de Servicio	Número de Peticiones	Fallos Encontrados	Probabilidad de Fallo	Tiempo Medio entre Fallos
8 horas	25	1	0.04	8
16 horas	50	2	0.04	8
32 horas	80	4	0.05	11.3
TOTAL			0.13	27.3

Tabla 4.7 Confiabilidad

Fuente: Elaboración Propia

Por tanto el valor promedio de fallas producidas en un tiempo de servicio (PFTS) es la siguiente:

$$PFTS = \frac{0.04 + 0.04 + 0.05}{3} = \frac{0.13}{3} = 0.043$$

Lo que indica que el sistema en promedio puede presentar 43 fallas cada mil peticiones.

El tiempo promedio de presentarse estos fallos es (TMF):

$$TMF = \frac{8 + 8 + 11.3}{3} = \frac{27.3}{3} = 9.1$$

Lo que indica que el sistema presenta una falla en promedio cada 9.1 en que se hace uso del mismo.

Entonces se tiene:

$$(1 - PFTS)\% = (1 - 0.043)\% = 0.957 = 95.7 \%$$

Por tanto sabemos que el presente sistema tiene la capacidad de ser utilizado libre de errores con un 95.7 % y con una probabilidad de que suceda una falla cada 23 peticiones.

Característica	Valor	Observación	Lo que se espera
Confiabilidad	95.7 %	Resultado obtenido durante la etapa de las observaciones	Se tiene en cuenta que con el uso continuo del sistema se puedan realizar modificaciones para así mejorar la confiabilidad del sistema.

Tabla 4.8 Análisis de la Confiabilidad

Fuente: Elaboración Propia

4.4 Análisis Costo – Beneficio

Para calcular el costo del software se utiliza el modelo COCOMO II, este tiene una jerarquía de modelos: básico, intermedio, avanzado y se aplican a tres diferentes tipos de software

- **Orgánico.** Proyecto relativamente sencillos, menores a 50000 líneas de código.
- **Semiacoplado.** Proyectos intermedios en complejidad y tamaño. La experiencia de este tipo de proyectos es variable y las restricciones intermedias.
- **Empotrado.** Proyectos bastantes complejos, en los que apenas se tiene experiencia y en un entorno de gran innovación técnica.

En este caso se aplica el tipo orgánico ya que se encuentra es este rango. La ecuación de COCOMO básico tiene la siguiente fórmula:

$$E = \text{Esfuerzo} = a \times KLDC^b$$

Donde, KLDC es el número de líneas de código distribuido en millares para el proyecto. La ecuación para calcular el tiempo de desarrollo es la siguiente:

$$T = \text{Tiempo de duración del desarrollo} = c \times \text{Esfuerzo}^d$$

Los valores a, b, c y d vienen dados en la siguiente tabla:

Tipo de software	A	B	C	D
Orgánico	2.4	1.05	2.5	0.38
Semiacoplado	3.0	1.12	2.5	0.35

Empotrado	3.6	1.20	2.5	0.32
-----------	-----	------	-----	------

Tabla 4.9 Coeficientes COCOMO

Fuente: [Norman, Fenton, 1991]

Los valores que corresponden a nuestro software son: $a = 2.4$, $b = 1.05$, $c = 2.5$, $d = 0.38$.

El LDC es de 422.84.

Calculando el esfuerzo tenemos:

$$E = a \times LDC^b = 2.4 \times 4.2^{1.08} = 10.83 \text{ personas/mes}$$

Ahora se calcula el tiempo de duración del proyecto:

$$T = c \times E^d = 2.5 \times 10.83^{0.38} = 6.18 \text{ meses}$$

Entonces, el número de personas para desarrollar el proyecto es:

$$Nro. \text{ Personas} = \frac{E}{T} = \frac{10.83}{6.18} = 1.75 = 2 \text{ personas}$$

Teniendo en cuenta que el valor de fuerza laboral por persona es de 1000 \$us/por módulo y la aplicación cuenta con tres módulos, se tiene:

$$3 \text{ módulos} \times 1000 \frac{\$us}{\text{módulos}} = 3000 \text{ $us}$$

Por tanto se concluye que el costo del proyecto es de 3000 \$us.

5. CONCLUSIONES Y RECOMENDACIONES

Habiendo finalizado el desarrollo del proyecto, y logrando un sistema de información que cumple con los requisitos planteados por los clientes, están dadas las condiciones para evaluar el cumplimiento de los objetivos realizados, las conclusiones a las que se llegó y las recomendaciones para su uso del mismo

5.1 Conclusiones

Con el desarrollo e implementación del presente proyecto se ha logrado alcanzar los **objetivos planteados.**

Implementar un sistema de Consulta Externa para el Seguro Social Universitario de La Paz para brindar al médico la sistematización en la redacción de diagnósticos y tratamientos a los pacientes, minimizando el tiempo en el proceso del manejo de información de los asegurados.

Con respecto a los objetivos trazados en la etapa inicial, han sido cumplidos de manera satisfactoria, cumpliendo la creación de procedimientos mencionados en los objetivos específicos logrando que se cumpla el objetivo principal.

Se ha realizado el análisis, diseño y construcción dentro del marco de requerimientos del usuario. Logrando establecer el sistema que ayuda de gran manera al proceso de Consulta Externa, contando con la base de datos para el almacenamiento sistematizado de la información, donde el acceso a la información es inmediata, integra, confiable y funcional.

Entre las facilidades que tiene el sistema está la flexibilidad que permitirá a la unidad de Consulta Externa modificar datos en los expedientes clínicos, añadir nuevas consultas y dar la facilidad de solicitar a otras unidades servicios como: Farmacia, Imagenología, Fisioterapia, Laboratorio, etc.

Para cada una de las tareas realizadas en el desarrollo del sistema, se diseñó interfaces graficas sencillas que permitan a los usuarios trabajar cómodamente y evitar confusiones. La elaboración de interfaces similares a formularios ya utilizados en el seguro y hacer partícipe al usuario del diseño de las mismas, contribuye a realizar un software de fácil uso y entendimiento.

Por tanto la implementación del Sistema de Gestión Hospitalario Modulo Consulta Externa Caso: Seguro Social Universitario, mejora el trabajo en los procesos técnicos de la unidad de Consulta Externa, cumpliendo todos los requerimientos establecidos y cubriendo satisfactoriamente las necesidades del usuario.

Se logró una comunicación entre farmacia y los laboratorios, de esta manera evitar la pérdida de documentación en el trayecto

5.2 Recomendaciones

Al finalizar el presente proyecto se recomienda:

- Desarrollar en un mismo lenguaje los demás módulos del sistema de gestión hospitalario GEHMA v3.0 como Hospitalización, Farmacia, Laboratorios y otros. Para un buen funcionamiento y una mejor integración y de esta manera poder tener un mejor tratamiento de la información, para el mejoramiento y fortalecimiento de la toma de decisiones en el SSU.
- A futuro se migre la base de datos al SQL SERVER 2008 debido a que tiene muchas mejoras.

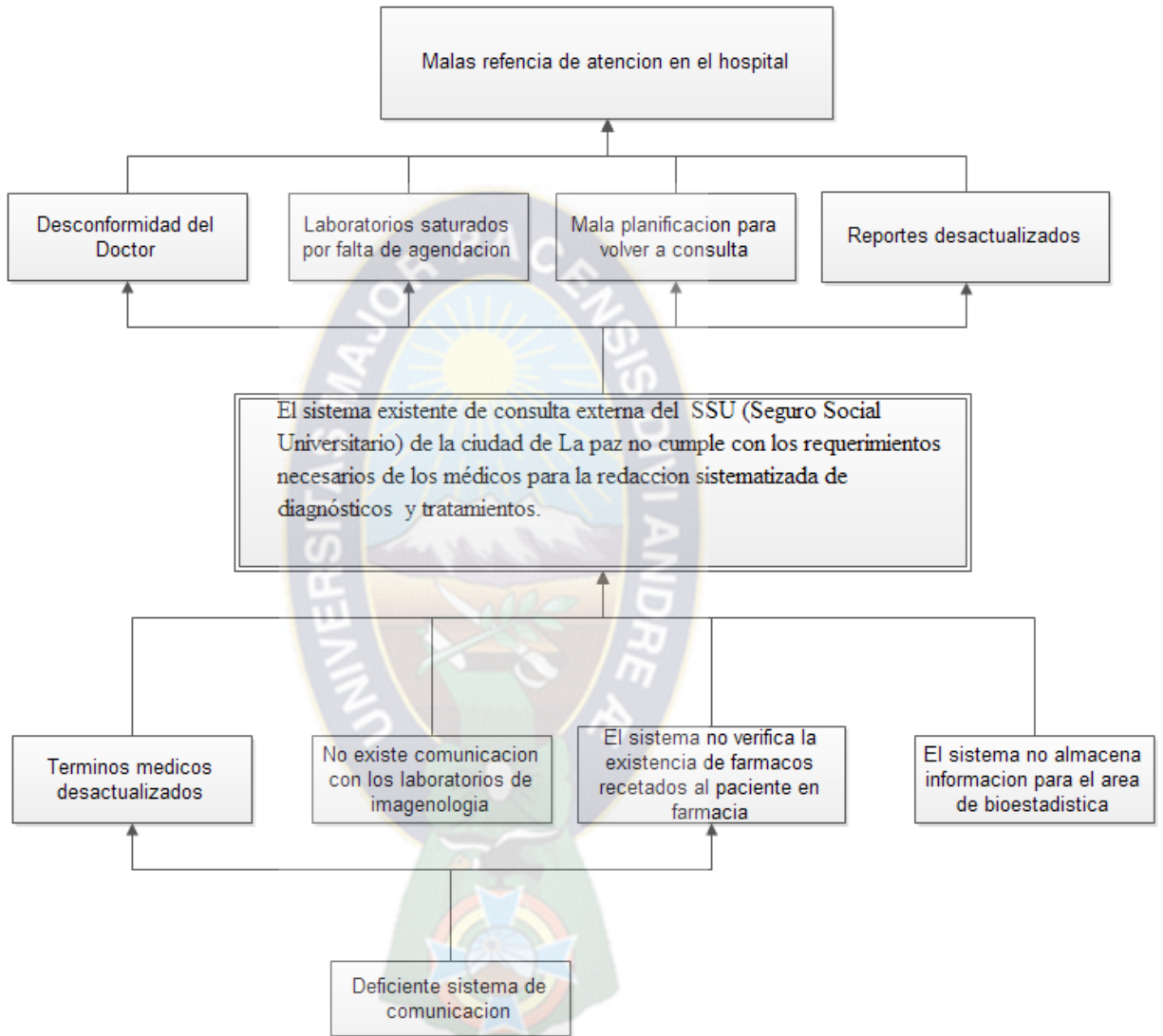
- Realizar el cambio constante de claves de acceso al sistema, para evitar fugas de información que afecten al Seguro Social Universitario.



BIBLIOGRAFÍA

- [Jacobson Ivar, Grady Booch, James Rumbaugh] EL PROCESO UNIFICADO DE DESARROLLO DE SOFTWARE, 1ra Edición Addison Wesley 1999. Iberoamericano
- [FOWLER M., 1999] UML gota a gota, 1ra ed., Addison Wesley Logman, México. (003_T-1798)
- [Pressman R. 2002] *Ingeniería de Software, Quinta edición*, Editorial Mc GrawHill, Madrid.
- [Silvio Martínez 2003], COMO REALIZAR UNA TESIS, 5ta edición
- [IVAR JACOBSON, GRADY BOOCH, JAMES RUMBAUGH, 2000] El Proceso Unificado de desarrollo de software. 1ra. Edición en español: Pearson Education S. Introducción al Proceso Unificado de Desarrollo (RUP) – Patricio Letelier Torres, Universidad Politécnica de Valencia, España
- [Escribano, 2002] Escribano F. G. *Ingeniería del Software II*. (2002).
- [Hammer, 1995] Hammer, M. (1995). *Reingeniería*. Colombia: Editoria Norma Bogotá
- [Jaimes, 2005] Jaimes, X. R. (2005), *Sistema Integrado de Administración Hospitalaria para la Caja Petrolera de Salud* (Proyecto de Grado).
- [Lewis, 1994] Lewis G. (1994). . *What is Software Engineering*. Inglad: Editorial DataPro.
- [Pastén, 2011] Pastén, E. (2011). *Seguro Social Universitario*, La Paz- Bolivia. 2007-2011.
- [Pressman, 2005] Pressman R. S. (2005). *Ingeniería Del Software*. México: Editorial McGraw Hill 5ta edición.
- [Pressman, 2001] Pressman R. S. (2001). *Ingeniería de Software enfoque práctico*. México. Ed McGraw Hill.
- [Pressman, 1998] Pressman R. S. (1998). *Ingeniería de Software enfoque práctico*. México: Ed. McGraw Hill 5ta edición.

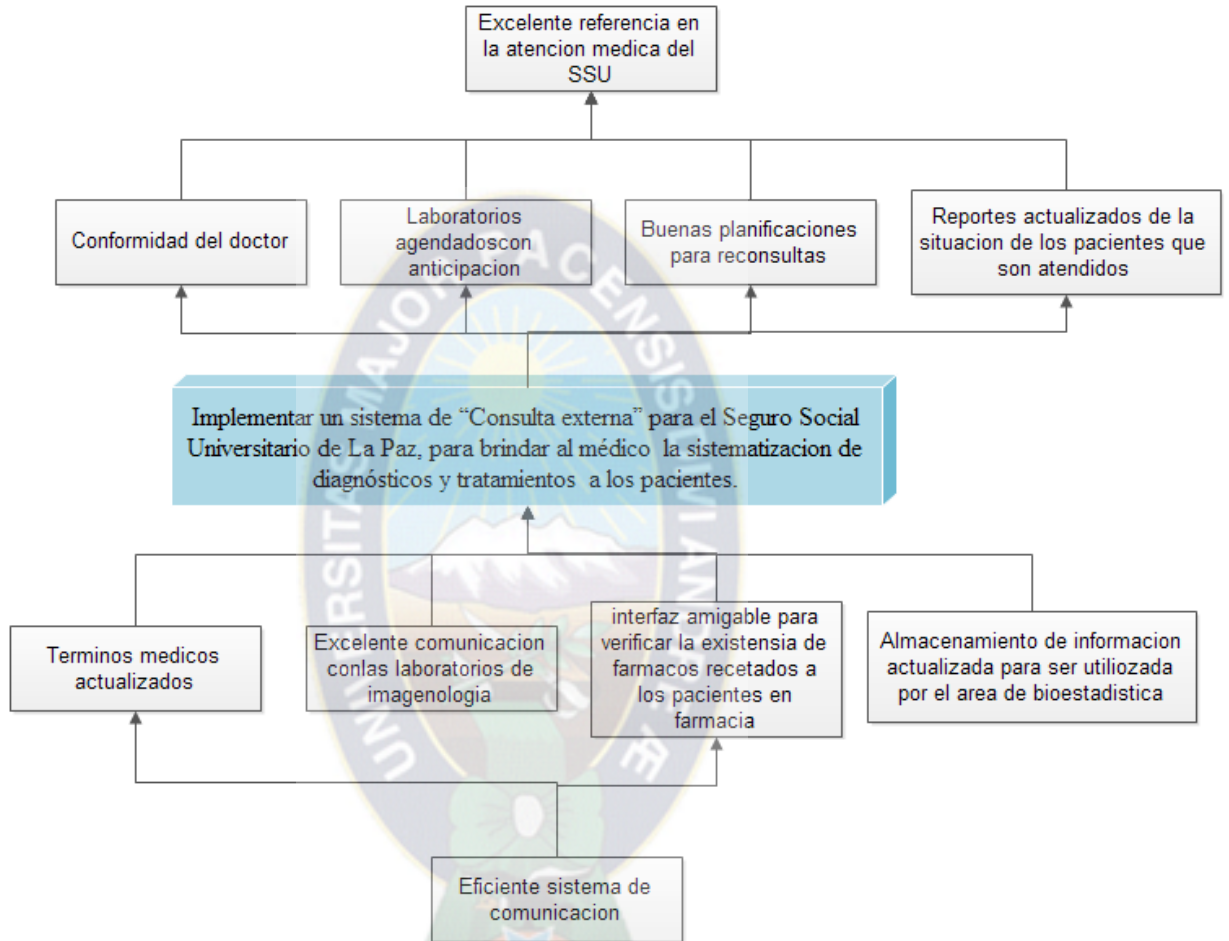
Anexo A



Árbol de problemas

Fuente (Elaboración Propia)

Anexo B



Árbol de objetivos

Fuente (Elaboración Propia)