

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



PROYECTO DE GRADO
“CONTROL Y SEGUIMIENTO DE ESTABLECIMIENTOS DE
SALUD DEL SERVICIO DEPARTAMENTAL DE SALUD”
SEDES LA PAZ

PARA OPTAR AL TÍTULO DE LICENCIATURA EN INFORMÁTICA
MENCIÓN: INGENIERÍA DE SISTEMAS INFORMÁTICOS

POSTULANTE: Nelly Irma Nogales Quispe

TUTOR: MSc. Franz Cuevas Quiroz

REVISOR: Lic. Efraín Silva Sánchez

La Paz – Bolivia

2009

DEDICATORIA DEDICATORIA

A ti mi Dios por darme la vida y darme la oportunidad de cumplir un objetivo en mi vida.

A mi papá y mamá por su amor y comprensión constante.

A mis hermanos y hermanas, en a especial Jenny por, el ejemplo, y los consejos que siempre me manifestaron.

AGRADECIMIENTOS

A mi Tutor MSc. Franz Cuevas Quiroz, excelente catedrático, mentor y amigo, por todos los valiosos conocimientos transmitidos de manera desinteresada.

A mi Revisor Lic. Efraín Silva Sánchez, por todo el apoyo y la orientación profesional en la culminación del presente documento.

A la Unidad Planificación Y Unidad De Acreditación y Certificación del SEDES LA PAZ por permitirme implementar el presente proyecto, y de manera especial al Sr. Erick Troche por brindarme su colaboración en el desarrollo del presente proyecto.

RESUMEN

El ser humano, por naturaleza, siempre ha sido ávido de conocimiento. Con la aparición de los sistemas Web, las puertas del conocimiento se abrieron, dando paso a una nueva forma de concepción, donde el tiempo y la distancia desaparecen.

El presente documento pretende reflejar, el trabajo realizado para la Unidad de Acreditación y Certificación del Servicio Departamental de Salud de La Paz en especial en la unidad de Control de Establecimientos de Salud.

El Sistema denominado Control y Seguimiento de establecimientos de Salud Del Servicio Departamental de Salud SEDES LA PAZ, fue diseñado para responder a las necesidades específicas de la institución.

El sistema provee información oportuna concerniente a establecimientos de Salud (como ser de servicios: Médicos, Odontológicos, Farmacéuticos Laboratorios, Ópticos y otros) controlando a través del registro de trámites la duplicidad de Registros, además realiza el seguimiento de infracciones y multas.

Para los establecimientos de Salud Pública se diseñó una interfaz que les permita registrar el movimiento mensual de los medicamentos que administran, con el fin de programar de forma exacta sus nuevas dotaciones.

Se combinó las metodologías RATIONAL UNIFIED PROCESS (RUP) y MODELO DE PROCESO DE INGENIERÍA debido a las características del grupo de trabajo que desarrolló el sistema, haciendo una selección de artefacto de manera que se integren y complementen, esto permitió facilitar el diseño y la implementación.

La implementación permitió dotar a la Unidad de Acreditación y Certificación del Servicio Departamental de Salud de un software especializado que erradique el problema de pérdida de información y agilice los trámites de licencias de funcionamiento.

INDICE

1. Marco Referencial	1
1.1. INTRODUCCIÓN	1
1.2. ANTECEDENTES	2
1.2.1. ANTECEDENTES DE LA INSTITUCIÓN	2
1.2.3. ANTECEDENTES DE TRABAJOS REALIZADOS	5
1.3. ANÁLISIS Y PLANTEAMIENTO DEL PROBLEMA	7
1.3.1. DEFINICIÓN DEL PROBLEMA	7
1.4. OBJETIVOS	7
1.4.1. Objetivo General	7
1.4.2. Objetivos Específicos	8
1.5. JUSTIFICACIÓN	8
1.5.1. Justificación Técnica	8
1.5.2. Justificación Económica	8
1.5.3. Justificación Social	9
1.6. METODOLOGIA	9
1.6.1. Métodos Y Medios De Investigación Científica	9
1.6.2. METODOS Y MEDIOS DE INVESTIGACIÓN INFORMÁTICA	10
1.7. ALCANCES Y APORTES	10
2. Marco Teórico	14
2.1. MARCO INSTITUCIONAL	15
2.1.1. UNIDAD DE ACREDITACION Y CERTIFICACION	15
2.1.1.1. ESTABLESIMIENTOS DE SALUD	15
2.2. MARCO CONCEPTUAL	16
2.2.1. METODOLOGIA	20
2.2.2. RATIONAL UNIFIED PROCESS (RUP)	20
2.2.2.1. Características de la metodología RUP	21
2.2.2.2. Dirigido por casos de uso	21
2.2.2.3. Centrado en la arquitectura	22
2.2.2.4. Iterativo e incremental	22
2.2.3. CICLOS DE VIDA Y FACES DEL PROCESO UNIFICADO	22

2.2.3. FASES EN RUP	23
2.2.3.1. Fase De Inicio	24
2.2.3.2. Fase De Elaboración.	24
2.2.3.3. Fase De Construcción.	24
2.2.3.4. Fase De Transición.	24
2.2.4. FASES E ITERACIONES DE LA METODOLOGÍA RUP	25
2.2.4.1. Flujo De Trabajo De Procesos	25
2.2.4.2. Flujo de trabajo de soporte	25
2.2.5. BLOQUES DE CONSTRUCCIÓN	26
2.2.6. FASE DE TRANSICION	26
2.2.6.1. Mapa Navegacional	26
2.2.6.2. Elementos de la metodología	38
2.3. METRICAS DE CALIDAD	38
2.3.1. Modelo de Calidad Establecido por el estándar ISO 9126	44
2.3.2. Características Propuestas por ISO-9126	45
1. Funcionalidad	46
2. Confiabilidad	47
3. Usabilidad	47
4. Eficiencia	48
5. Mantenibilidad	49
6. Portatilidad	49
3. Diseño del Sistema	50
3.1. Analisis, desarrollo e implementación	52
3.2.3.2. Fases de la metodología RUP	52
3.1.3.2.1 Modelado del Negocio	52
3.2.1.1. Diagrama de casos de uso	53
3.2.1.2 Modelo de Objetos	54
3.2.2 Diagramas de Actividad	55
3.2.3 Modelado de Análisis	58
3.2.3.1 Diagramas de Secuencia	59
3.2.4 Modelado de Diseño	60
3.2.5 Modelo Relacional	60
	62

3.2.5 Modelado de Implementación	64
3.2.5.1 Diagrama de Componentes	65
3.2.5.2 Diagrama de Despliegue	65
3.2.5.3 Pantallas del Sistema	66
3.2.5 Modelado de Pruebas	66
4. Evaluación Del Sistema	70
4.1. Calidad de software	74
4.1.1. Funcionalidad	74
4.1.2. Confiabilidad	75
4.1.3.Facilidad de mantenimiento	76
4.1.4. Portabilidad	78
5. Conclusiones y recomendaciones	79
5.1 Conclusiones	83
5.2 Recomendaciones	83
Bibliografía	84





CAPITULO I ANTECEDENTES Y MARCO REFERENCIAL

Antecedentes y Marco Referencial

1.3. INTRODUCCIÓN

La tecnología en el mundo ha revolucionado la forma en que la información es administrada, uno de estos avances es la comunicación a través de Internet que nos permite romper las barreras de distancia.

La Web se ha convertido en un elemento clave, tanto en el desarrollo de las empresas como de las instituciones, ofreciendo información y una amplia gama de servicios a través de la misma. En Bolivia este hecho no es desconocido del todo, efectivamente podemos observar un índice creciente de portales institucionales privados y públicos, que ofrecen información, servicios, productos y otros; pero pocos sistemas orientados a apoyar el trabajo que se realiza dentro de las instituciones públicas.

En el presente proyecto de grado, se dan bases de la implementación del software que se desarrollará en la Unidad de Acreditación y certificación del Servicio Departamental de La Paz (SEDES-La Paz), se pretende hacer uso de Internet y de herramientas modernas y probadas para solucionar la problemática de las funciones, además de brindar apoyo pertinente al área de Salud, la cual demanda información de forma eficaz y eficiente además de oportuna que permita la toma de decisión, para que los establecimientos de Salud (Farmacias, Laboratorios, Odontológicos y Servicios de Salud), cumplan con requisitos mínimos indispensables bajo ley vigente para su buen funcionamiento.

1.4. ANTECEDENTES

1.4.1. ANTECEDENTES DE LA INSTITUCIÓN

El Servicio Departamental de Salud desarrolla sus actividades en el marco del ordenamiento legal que rige el Sistema Nacional de Salud, bajo el marco regulatorio de las leyes 1178 de Administración y Control Gubernamental, 1654 de Descentralización Administrativa, 1551 de Participación Popular, D.S. 2028 Ley de Municipalidades, Ley 2426 del Seguro Universal Materno Infantil (SUMI), D.S. 26873 Sistema Nacional Único de Suministros, Medicamentos e Insumos (SNUS), D.S. 26874 Reglamento de Prestaciones y Gestión del SUMI, DS. 26875 Modelo de Gestión y Directorio Local de Salud (DILOS).

De acuerdo a la decisión política del Servicio Departamental de Salud de La Paz (SEDES), manifestada en la resolución de aprobación de su estructura organizativa RA DIR. SEDES- N°001/2004, validada por la Prefectura del Departamento mediante resolución RAP N° 082 del 17 de febrero de 2005, en la cual se aprueba la estructura organizacional del SEDES, y cuya misión es la facultada de generar, recaudar e invertir recursos, e implementar la gestión epidemiológica, sanitaria y pública para el desarrollo del sector salud, y cuya Visión garantizar un alto nivel de gestión pública y sanitaria en la red municipal de Salud, coadyuvando en la gestión compartida y concurrente en salud, garantizando un acceso universal y permanente a servicios de salud con calidad, enfoque de género, interculturalidad y equidad para cumplir su misión.

El Servicio Departamental de Salud es el articulador de las políticas nacionales y de la gestión municipal, coordina y supervisa la gestión de salud en el departamento y es el encargado de cumplir y hacer cumplir la política de salud y normas de orden público en su jurisdicción territorial en el sector público, el Seguro Social de corto plazo, Iglesias, Privados con y sin fines de lucro y la Medicina Tradicional.

Misión

El Servicio Departamental de Salud es la máxima instancia técnica y normativa en el área, responsable de aplicar las políticas nacionales y formular las políticas regionales, a través de las redes de servicios de salud, redes sociales y DILOS, interactuando descentralizadamente con sectores y actores públicos, privados, municipales y comunidades indígenas y originarias, con una profunda vocación de servicio enmarcados en los Códigos de Ética Profesional, los mandatos de las Leyes y la Política Nacional de Salud, hasta conseguir un estado de bienestar en la población, con elevados índices de desarrollo humano en el Departamento, optimizar los recursos.

Visión

Un Servicio Departamental de Salud de La Paz, rector del sector salud, con alto nivel de gestión institucional compartida, de corresponsabilidad con actores públicos, privados, municipales y comunidades indígenas y originarias, que garantiza el acceso universal y permanente a servicios de salud integrales con calidad, enfoque de género, interculturalidad y equidad, generando espacios y conductas saludables que contribuyen al desarrollo humano sostenible.

Funciones

El Servicio Departamental de Salud es el articulador de las políticas nacionales y de la gestión municipal, coordina y supervisa la gestión de salud en el departamento y es el encargado de cumplir y hacer cumplir la política de salud y normas de orden público en su jurisdicción territorial en el sector público, el Seguro Social de corto plazo, Iglesias, Privados con y sin fines de lucro y la Medicina Tradicional.

La Unidad de Acreditación y Certificación del Servicio Departamental de Salud de La Paz SEDES LA PAZ, depende de la Prefectura del Departamento de La Paz de manera DESCONCENTRADA. El responsable de Unidad es

designado por el Director de SEDES LA PAZ, en lo administrativo, en lo que se refiere a dotación de insumos y materiales, depende de la Jefatura Administrativa Financiera de SEDES LA PAZ.

La Unidad de Acreditación y Certificación, se organiza de acuerdo a las características que mencionaremos de la siguiente manera:

ESTABLESIMIENTOS DE SALUD

SERVICIOS GENERALES

NUTRICIÓN

CAJA

1.3.3. ANTECEDENTES DE TRABAJOS REALIZADOS

Para el desarrollo del proyecto se analizaron los trabajos siguientes llevados a cabo en la misma institución:

SISTEMA DE INFORMACION AUTOMATIZADO PARA CERTIFICACION Y ACREDITACION DE SERVICIOS –SEDES LP

Trabajo realizado por el Lic. Miguel Ángel Foronda Arias, sistema que automatiza gran parte de los procesos que se lleva a cabo en el área de Certificación y Acreditación de Servicios, automatizando la liquidación de que hacen las áreas de nutrición y servicios generales contribuyendo a mejorar el servicio a la sociedad.

SISTEMA DE SEGUIMIENTO DE MODELO DE INFORMACION BASICA PARA EL SISTEMA NACIONAL DE INFORMACION EN SALUD SEDES.

Trabajo realizado por la Lic. Jovanna Isabel Miranda Mollinedo, sistema que contribuye a brindar información sobre el área de trabajo y el entorno socio económico y cultural de los Establecimientos de Salud del Departamento de La Paz.

*SISTEMA DE INFORMACION INTEGRADO PARA LA UNIDAD DE
FARMACIA Y LABORATORIO DEL SERVICIO DEPARTAMENTAL DE
SALUD SEDES LP*

Trabajo realizado por la Lic. Norma Jenny Mérida Céspedes, sistema que contribuye a brindar información en la administración de suministros de los medicamentos e insumos de cada establecimiento dependiente del sistema público; facilitando el trabajo de la Jefatura de Farmacia y Laboratorio.

1.4. ANÁLISIS Y PLANTEAMIENTO DEL PROBLEMA

Los problemas que enfrenta la Unidad de Acreditación y Certificación del Servicio Departamental de Salud de La Paz es: el manejo de información, reporte e informes que se realiza en forma manual por lo que existe cierta demora en la atención tanto de los encargados como de los usuarios que solicitan los trámites. Se encontró las siguientes dificultades:

- Cuenta con una administración manual en el control y seguimiento del manejo de la información, en la generación y elaboración de reportes, lo que ocasiona demora y gastos en los procesos.
- Las inspecciones realizadas a los Establecimientos de Salud, no cuentan con un seguimiento histórico de infracciones ni de inspecciones, que determine si dicho establecimiento cometió o no dicha infracción anteriormente.
- Existe algunos reportes estadísticos, lo que no satisface la información requerida.
- Demora en el llenado de libros y formularios por ser de operación manual.
- Debido al manejo de archivos en forma manual, ocasiona posibles extravíos de documentos.

- La falta de tecnología adecuada, dificulta la obtención de información del seguimiento de resultados de trámites, lo que significa una búsqueda manual para su actualización.

2.3.1. DEFINICIÓN DEL PROBLEMA

La Unidad de Acreditación y Certificación (Establecimientos de Salud) del Servicio Departamental de Salud de La Paz (SEDES) atraviesa ciertas dificultades debido a la falta de tecnología adecuada en el almacenamiento de la información el cual está propenso a errores y demora en la obtención de registros de búsqueda y generación de reportes.

Es por ello que se ha identificado como problema principal:

¿El Sistema de la Unidad de Acreditación y Certificación (Establecimientos de Salud) Servicio Departamental de Salud de La Paz (SEDES), permitirá mejorar la calidad en el manejo de información, para la generación y elaboración de reportes e informes?

2.4. OBJETIVOS

2.4.1. Objetivo General

Desarrollar e implementar un sistema de información para el control y seguimiento de la Unidad de Acreditación y Certificación (Servicio de Salud) del Servicio Departamental de Salud de La Paz (SEDES), para la prestación de servicio de manera oportuna y confiable.

2.4.2. Objetivos Específicos

- Elaborar una base de datos para registrar de manera eficiente, protegida y oportuna de datos de los Establecimientos de Salud y controlar el seguimiento de los mismos.
- Determinar flujos de registros del seguimiento de los trámites y generar informes de los resultados obtenidos.
- Diseñar un módulo para el control de las citaciones de los inspectores del área de Establecimientos de Salud.
- Diseñar un módulo para el control de las sanciones a los Establecimientos de Salud previo procedimiento administrativo.

2.5. JUSTIFICACIÓN

2.5.1. Justificación Técnica

La unidad de Acreditación y certificación, Actualmente cuenta con equipos de computación adecuados para la implantación del sistema propuesto, los que permita mejorar los resultados organizacionales.

Con el sistema a ser realizado se podría mejorar el manejo de la información de manera oportuna y uniforme, y que la misma sea un instrumento para la toma de decisiones con respecto a los datos emitidos. El sistema se diseñará de acuerdo a las necesidades de la institución y de acuerdo a las normas establecidas por la tecnología.

2.5.2. Justificación Económica

Ya que la Unidad de Acreditación y certificación cuenta con hardware disponible y con la posibilidad de acceder a un software acorde al sistema, este reducirá el gasto de:

- Los costos de implementación del sistema ya que cuenta con personal adecuado para su mantenimiento.
- Se logra reducir los costos de control administrativo (formularios, papeletas, registros) y administrar de esta manera los recursos de la unidad.

2.5.3. Justificación Social

El proveer un sistema con las características mencionadas es ya una justificación social, con la utilización de una tecnología adecuada se brinda un mejor servicio a la población, con mayor seguridad y en el menor tiempo posible.

2.6. METODOLOGIA

2.6.1. Métodos Y Medios De Investigación Científica

Durante el desarrollo de modelo se empleará el método científico que consta de la siguiente forma:

- Recolección de datos
- Formulación de problemas

- Árboles analíticos
- Marco lógico
- Método Científico

Los métodos y las técnicas son el camino o la iba por donde transita el proceso del proyecto planteado, y permite transformar el objeto de estudio desde su estado no deseado hasta el estado propuesto.

El método de investigación a utilizar será fáctico encargándose de estudiar hechos auxiliándose de la observación, medición y experimentación

2.6.2. METODOS Y MEDIOS DE INVESTIGACIÓN INFORMÁTICA

Para poder alcanzar los objetivos propuestos en el presente proyecto se recurrirá a los siguientes métodos.

Para el análisis y desarrollo de sistema:

- El proceso unificado de desarrollo de software (RUP)
- El Lenguaje de Modelamiento Unificado (UML)

Para la programación:

Las herramientas para el desarrollo que se utilizaran de acuerdo a los requerimientos y especificaciones de la unidad de Acreditación y Certificación (Servicio de Salud) son las siguientes

- Un administrador y gestor de base de datos (SGDB) MySQL, muy utilizado dentro de los entornos de desarrollo de sistemas Web con PHP

- Un servidor Web Apache, que es un programa que permite que otros ordenadores vean el sistema Web mediante el navegador.
- Plataforma Windows XP o LINUX

2.7. ALCANCES Y APORTES

El presente proyecto realiza una investigación aplicada con las siguientes características.

- Módulo de registro de Establecimientos de Salud, propietarios y empleados.
- Módulo de Control de inspecciones a los establecimientos de salud.
- Módulo de Control de citaciones y sus respectivas sanciones.
- Módulo de Reporte e informes.

El presente proyecto cumplirá con los requerimientos establecidos por los usuarios de la unidad de Acreditación y Certificación (Establecimientos de Salud), y permitirá que tenga una administración más eficiente, eficaz y sencilla de toda la información que se genera.

También proporciona una herramienta que brinda información oportuna y confiable como soporte para la toma de decisiones.

Por tanto el alcance del presente proyecto abarcará lo siguiente:

AREA URBANA / RURAL

Servicios de salud

- Hospital

- Clínica
- Clínica Especializada
- Centro Médico y / o Quirúrgico Especializado
- Consultorio Médico General
- Consultorio Médico Especialista
- Poli consultorio dos o más profesionales (por profesional)
- Servicio de Atención Medica Pre- Hospitalaria y Transporte
- Servicio de Atención Medica de Aseguradora SOAT
- Consultorio de Medicina con Gabinete (Eco. RX - Gamma-Tomo-Scanner. Etc.)
- Gabinete de Fisioterapia y Rehabilitación
- Gabinete Óptico
- Gabinete Radiológico
- Gabinete Mecánica Dental
- Consultorio de Psicología o Nutrición
- Consultorio Naturista, Callahuaya etc.
- Consultorio de Acupuntura Nacional
- Consultorio Naturista o Acupuntura Extranjero
- Servicio de Tatuajes por Acupuntura

Servicios Odontológicos

- Consultorio Odontológico General
- Consultorio Odontológico Especialista
- Consultorio Odontológico con Gabinete

Servicios De Farmacia Y Laboratorio

- Farmacias publicas
- Farmacias privadas
- Farmacias de clínicas
- Farmacias hospitales
- Laboratorios farmacéuticos

GABINETE URBANO / RURAL

- Óptica
- Fisioterapia y rehabilitación
- Radiológico
- Mecánico Dental
- Consultório de medicina tradicional o alternativa
- Servicio de tatuajes por acupuntura
- Centro medico y/ o quirúrgico especializado
- Clínicas
- Clínica especializada
- Hospital
- Institutos técnicos de enseñanza superior
- Servicio de Tatuajes por Acupuntura
- Guarderías infantiles u hogares de niños
- Hogares de ancianos o casas de reposo
- Institutos estéticos y de adelgazamiento
- Concesionaria de hospitales
- Institutos estéticos y de adelgazamiento



CAPITULO II

MARCO TEÓRICO

2. MARCO TEÓRICO

El presente capítulo proporciona el fundamento teórico que permite elaborar el sistema de Control y Seguimiento de Establecimientos de Salud del Servicio Departamental de Salud SEDES LA PAZ.

2.1. MARCO INSTITUCIONAL

SEDES LA PAZ, tiene como objetivo fundamental mejorar las condiciones higiénicas y la calidad de vida en los Servicios Generales, como los servicios de agua potable, disposición sanitaria de excretas y control de residuos sólidos.

2.1.1. UNIDAD DE ACREDITACION Y CERTIFICACION

La Unidad De Acreditación y Certificación dependiente del SEDES La Paz, garantiza la protección sanitaria, haciendo cumplir las políticas de salud mediante la certificación, habilitación, acreditación, monitoreo y evaluación continua de los servicios de salud y de los servicios generales; para la prestación eficiente, oportuna y sostenida a la población del Departamento de La Paz, mediante el Sistema Único Departamental de Salud.

Por lo que el órgano competente y ejecutor de las normas y reglamentos es el Director Técnico del SEDES La Paz a través de la Unidad de Acreditación y Certificación, facultada para dar cumplimiento a las disposiciones sanitarias en actual vigencia, a través del control y vigilancia de los establecimientos y/o locales privados, recreativos, de tolerancia y similares, en sus diferentes actividades de acuerdo a las siguientes competencias:

- Certificar y acreditar, a nombre del Estado a los establecimientos que ofertan bienes y servicios relacionados a la salud pública. Previa información y/o educación.
- Regular mediante normas vigentes el comportamiento humano, relacionado a las buenas prácticas de higiene, seguridad industrial, laboral y de consumo de bienes y servicios de consumo, así como las acciones en caso de transgresiones.
- Otorgar documento legal válido (certificación), previo cumplimiento de requisitos formales a los establecimientos industriales, de producción, de transporte y de consumo. Así como a los establecimientos de salud públicos, privados o de otra naturaleza.

La Unidad de Acreditación y Certificación, se organiza de acuerdo a las siguientes características:

- Establecimientos de salud
- Servicios generales
- Servicio de nutrición (alimentos y bebidas)
- Caja

2.1.1.1. ESTABLECIMIENTOS DE SALUD

De acuerdo al Código de Salud, El reglamento de Hospitales, Ley del Medicamento. La UAC tiene competencia exclusiva, en el control de los establecimientos públicos, privados, de convenio o de otra índole, en control y seguimiento de los Establecimientos Que Ofertan Servicios Médicos.

En el área de establecimientos de salud cuenta con los siguientes sub áreas:

SERVICIOS MEDICOS

Tiene competencia en el control de establecimientos que ofertan servicios de atención a pacientes, en cuanto a las patologías humanas en la Ciudad. Referente a la otorgación de licencias y renovaciones de funcionamiento.

En la unidad de Acreditación y Certificación el Área de Servicios Médicos viene a cumplir las siguientes funciones:

1. Planificar dentro del área de adscripción actividades de Evaluación para apertura o renovación de Resoluciones Administrativas de Establecimientos Prestadores de Servicios de Salud.
2. Coordinar dentro del área de adscripción actividades de Evaluación de forma multidisciplinaria en Establecimientos Prestadores de Servicios de Salud que requieran evaluación técnica
3. Supervisar dentro del área de adscripción el funcionamiento, acorde a normas vigentes, de Establecimientos Prestadores de Servicios de Salud.
4. Evaluar a través de Instrumentos validados técnicamente Establecimientos Prestadores de Servicios de Salud.
5. Implementar instrumentos validados técnicamente para Evaluación, Supervisión y Seguimiento de Establecimientos Prestadores de Servicios de Salud.
6. Organizar un sistema de vigilancia y seguimiento que sea oportuno, eficiente y preciso

En el marco del Código de Salud, el reglamento de hospitales, y la ley del ejercicio de la medicina, corresponde a los siguientes rubros:

- Medicina natural y/o tradicional

- Clínicas privadas
- Hospitales privados
- Consultorios médicos
- Gabinetes de Rayos X
- Servicios de Ecografía
- Servicios de Imaginología
- Poli consultorios
- Consultorio de psicología o nutricionistas
- Óptica
- Fisioterapia y rehabilitación
- Radiológico
- Mecánico Dental
- Consultorio de medicina tradicional o alternativa
- Servicio de tatuajes por acupuntura
- Otros.

SERVICIOS DE FARMACIAS Y LABORATORIOS

FARMACIAS.- Encargado en el control de Farmacias, tanto públicas y privadas, productos farmacéuticos naturales y otros, en lo referente a las licencias e infracciones, en cumplimiento a los procedimientos administrativos, incluyendo clausuras. Esta actuación, se lo realiza en directa coordinación con jefatura de farmacias de SEDES LA PAZ.

Las funciones que cumplen esta área son:

- Establecer una base de datos documental de farmacias y regentes farmacéuticos
- capacitar sobre la ley del medicamento y decretos reglamentarios

- Implementar la conformación de comités de farmacia y terapéutica por red
- Supervisión y seguimiento periódico de buenas prácticas de farmacias
- Implementación de un sistema de información de medicamentos en servicios de salud
- Coordinación de aperturas, cierres, depósitos y traslados de farmacias privadas con jefatura departamental
- Análisis y validación de datos snus 3-04
- Vigilancia al cumplimiento de indicadores de gestión logística
- Validación de datos emitidos por el SIAF en hospitales del departamento.
- Coordinar inspecciones con unidad de certificación
- Coordinar con unidad de planificación realización de convenios con universidades para conformación de pasantías e inspecciones de brigadas universitarias

LABORATORIOS.- Encargado en el control de los laboratorios, en lo referente a la otorgación de licencias, aplicación de sanciones, seguimiento.

Objetivo general de esta área es el estricto cumplimiento y aplicación de las normas legales sobre las autorizaciones de instalación, funcionamiento, condiciones y requisitos técnicos que deben cumplir los laboratorios clínicos tanto públicos como privados, ubicados en la ciudad como en el área rural, previas a su apertura, así como las normas reguladoras de las determinaciones clínicas que en ellos se realizan.

SERVICIOS DE ODONTOLOGIA

Encargado en el control de establecimientos odontológicos, laboratorios de Odontología, en lo referente a la otorgación de licencias, aplicación de sanciones, seguimiento.

2.2. MARCO CONCEPTUAL

2.2.1. METODOLOGIA

El Proceso Unificado es un proceso de desarrollo de *software*. Un proceso de desarrollo de software es el conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema de software.

El proceso Unificado utiliza el Lenguaje Unificado de Modelado (UML) para preparar todos los esquemas de un sistema, de hecho UML es una parte esencial del Proceso Unificado – sus desarrollos fueron paralelos.

Los aspectos definitorios del Proceso Unificado se resumen en tres frases claves

- Dirigidos por casos de uso
- Centrado en la arquitectura
- Iterativo e incremental.

2.2.2. RATIONAL UNIFIED PROCESS (RUP)

En primer lugar, el Proceso Unificado es un proceso de desarrollo de *software*. Un proceso de desarrollo de software es el conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema de software (Figura 2.1). Sin embargo, el Proceso Unificado es más que un proceso; es un marco de trabajo genérico que puedes especializarse par una gran variedad de sistemas software, para diferentes áreas de aplicación deferentes tipos de organización, diferentes niveles de aptitud y diferentes tamaños de proyectos.

El proceso Unificado utiliza el Lenguaje Unificado de Modelado (UML) para preparar todos los esquemas de un sistema, de hecho UML es una parte esencial del Proceso Unificado – sus desarrollos fueron paralelos.

Los aspectos definitorios del Proceso Unificado se resumen en tres frases claves – dirigidos por casos de uso, centrado en la arquitectura, e iterativo e incremental.

Esto es lo que hace único al Proceso Unificado (Figura 2.1)



Figura. 2. 1. Un proceso de desarrollo de software

Fuente: Proceso unificado de Desarrollo de Software, 1999

2.2.2.1. Características de la metodología RUP

Los aspectos definitorios del Proceso Unificado se resumen en tres frases claves

- Dirigidos por casos de uso

- Centrado en la arquitectura
- Iterativo e incremental.

2.2.2.2. Dirigido por casos de uso

Utiliza casos de uso que proporcionan los requisitos funcionales del sistema. El modelo de casos de uso que describe toda la funcionalidad del sistema. La filosofía de desarrollo que involucra los modelos de: casos de uso, análisis, diseño e implementación. La trazabilidad del sistema que implica cambios en requisitos de un caso de uso fácil de detectar las clases y componentes que afectan [Jacobson, 2001].

2.2.2.3. Centrado en la arquitectura

Involucra que con la utilización de los casos de uso, se describa la funcionalidad del sistema, la arquitectura define la forma del sistema y se describe mediante vistas.

2.2.2.4. Iterativo e incremental

El ser iterativo e incremental da beneficio de obtener un sistema robusto, reducir el riesgo de tener un mal producto, reducir el riesgo de no obtener el producto en el tiempo previsto y permite atacar problemas con requisitos incompletos.

2.2.3. CICLOS DE VIDA Y FACES DEL PROCESO UNIFICADO

Cada una de las etapas se desarrolla mediante el ciclo de iteraciones, la cual consiste en reproducir el ciclo de vida en cascada a menor escala. Los Objetivos de una iteración se establecen en función de la evaluación de las iteraciones (Figura 2.2)

Por lo que en Rational Unified Process se puede describir el proceso en dos dimensiones como ser:

- El eje horizontal representa *tiempo* y muestra el aspecto dinámico del proceso, expresado en términos de *ciclos, fases, iteraciones, y metas*.
- El eje vertical representa el aspecto estático del proceso; como está descrito en términos de *actividades, artefactos, trabajadores y flujos de trabajo*.

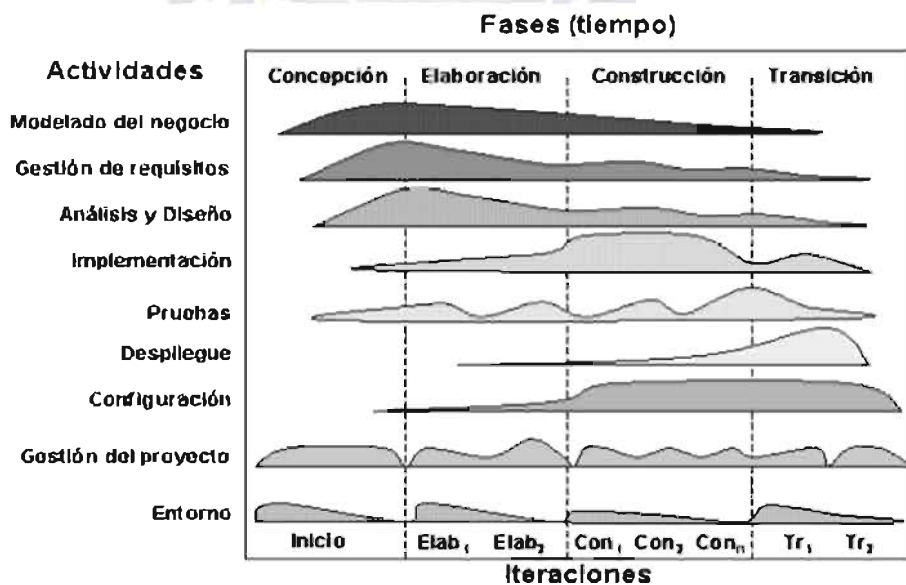


Figura 2.2.: Ciclo de vida del Proceso Unificado [Booch,2000]

2.3.3. FASES EN RUP

La metodología RUP se divide en 4 fases el desarrollo del software que son:

- Fase de Inicio (concepción)
- Fase de Elaboración
- Fase de Construcción
- Fase de Transición

2.3.3.1. Fase De Inicio

Dúrate esta fase se desarrolla una descripción del producto final a partir de una buena idea, y se presenta el análisis de negocio para el producto. A lo largo de la misma se deben establecer las principales funciones del sistema para los usuarios más importantes, la arquitectura del sistema a grandes rasgos y el plan de proyecto con una aproximación del coste del producto.

2.3.3.2. Fase De Elaboración.

Durante esta fase se especifica en detalle la mayoría de los casos de uso del producto y se diseña la arquitectura del sistema. Al final de esta fase el responsable del proyecto estará en condiciones de planificar las actividades y estimar los recursos necesarios para llevar a cabo el proyecto

2.3.3.3. Fase De Construcción.

En esta fase la línea base de la arquitectura obtenida como producto durante la fase anterior crece hasta convertirse en el sistema completo

2.3.3.4. Fase De Transición.

Cubre el periodo durante el cual el producto se convierte en versión beta. Conlleva actividades como la formación al cliente, la asistencia, resolución de incidencia y clasificación de aquella que justifica una versión del producto.

2.3.4. FASES E ITERACIONES DE LA METODOLOGÍA RUP

Vale mencionar que el ciclo de vida que se desarrolla por cada iteración, es llevada bajo dos flujos de trabajo:

- Flujo de Trabajo de Procesos
- Flujo de Trabajo de Soporte

2.3.4.1. Flujo De Trabajo De Procesos

Modelado de Negocios: Entendiendo las necesidades del negocio.

Gestión de Requerimientos: Traslado de las necesidades del negocio a un sistema automatizado.

Análisis y Diseño: Traslado de los requerimientos dentro de la arquitectura de software.

Implementación: Creando software que se ajuste a la arquitectura y que tenga el comportamiento deseado.

Pruebas: Asegurándose que el comportamiento requerido es el correcto y que todo lo solicitado está presente.

2.3.4.2. Flujo de trabajo de soporte

Configuración: Guardando todas las versiones del proyecto.

Administración del proyecto: Administrando horarios y recursos.

Ambiente: Administrando el ambiente de desarrollo.

Distribución: Hacer todo lo necesario para la salida del proyecto

2.3.5. BLOQUES DE CONSTRUCCIÓN

Los bloques de construcción se dividen en tres partes:

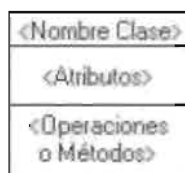
a. Elementos.

• Elementos Estructurales

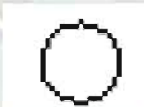
Los elementos estructurales en UML, es su mayoría, son las partes estáticas del modelo y representan cosas que son conceptuales o materiales.

Clases

Una clase es una descripción de un conjunto de objetos que comparten los mismos atributos, operaciones, métodos, relaciones y semántica. Una clase implementa una o más interfaces. Gráficamente se representa como un rectángulo que incluye su nombre, sus atributos y sus operaciones.



Interfaz

Una interfaz es una colección de operaciones que especifican un servicio de una determinada clase o componente. Una interfaz describe el comportamiento visible externamente de ese elemento, puede mostrar el comportamiento completo o sólo una parte del mismo. Una interfaz describe un conjunto de especificaciones de operaciones (o sea su signatura) pero nunca su implementación. Se representa con un círculo, , y rara vez se encuentra aislada sino que más bien conectada a la clase o componente que realiza.

Es una agrupación de métodos u operaciones que especifican un servicio de una clase o componente, describiendo su comportamiento, completo o parcial, externamente visible. UML permite emplear un círculo para representar las interfaces, aunque lo más normal es emplear la clase con el nombre en cursiva.

Colaboración

Define una interacción y es una sociedad de roles y otros elementos que colaboran para proporcionar un comportamiento cooperativo mayor que la suma de los comportamientos de sus elementos. Las colaboraciones tienen una dimensión tanto estructural como de comportamiento. Una misma clase puede participar en diferentes colaboraciones. Las colaboraciones representan la implementación de

patrones que forman un sistema. Se representa mediante una elipse con borde discontinuo.



Casos de Uso

Un caso de uso es la descripción de un conjunto de secuencias de acciones que un sistema ejecuta y que produce un determinado resultado que es de interés para un actor particular. Un caso de uso se utiliza para organizar los aspectos del comportamiento en un modelo. Un caso de uso es realizado por una colaboración. Se representa como una elipse con borde continuo.



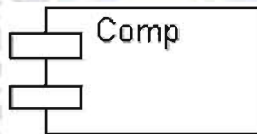
Clase Activa

Es una clase cuyos objetos tienen uno o más procesos o hilos de ejecución por lo tanto pueden dar lugar a actividades de control. Una clase activa es igual que una clase, excepto que sus objetos representan elementos cuyo comportamiento es concurrente con otros elementos. Se representa igual que una clase, pero con líneas del contorno son más gruesas que en la clase "normal"



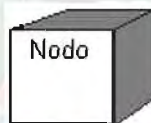
Componentes

Un componente es una parte física y por tanto reemplazable de un modelo, que agrupa un conjunto de interfaces, archivos de código fuente, clases, colaboraciones y proporciona la implementación de dichos elementos.. Un componente representa típicamente el empaquetamiento físico de diferentes elementos lógicos, como clases, interfaces y colaboraciones.



Nodos

Un nodo es un elemento físico que existe en tiempo de ejecución y representa un recurso computacional que, por lo general, dispone de algo de memoria y, con frecuencia, de capacidad de procesamiento. Un conjunto de componentes puede residir en un nodo.



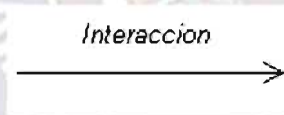
Estos siete elementos vistos son los elementos estructurales básicos que se pueden incluir en un modelo UML. Existen variaciones sobre estos elementos básicos, tales como actores, señales, utilidades (tipos de clases), procesos e hilos (tipos de clases activas) y aplicaciones, documentos, archivos, bibliotecas, páginas y tablas (tipos de componentes).

- **Elementos de comportamiento**

Los elementos de comportamiento son las partes dinámicas de un modelo. Se podría decir que son los verbos de un modelo y representan el comportamiento en el tiempo y en el espacio. Los principales elementos son los dos que siguen.

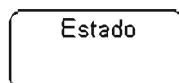
Interacción

Es un comportamiento que comprende un conjunto de mensajes intercambiados entre un conjunto de objetos, dentro de un contexto particular para conseguir un propósito específico. Una interacción involucra otros muchos elementos, incluyendo mensajes, secuencias de acción (comportamiento invocado por un objeto) y enlaces (conexiones entre objetos). La representación de un mensaje es una flecha dirigida que normalmente con el nombre de la operación.



Maquinas de estados

Es un comportamiento que especifica las secuencias de estados por las que van pasando los objetos o las interacciones durante su vida en respuesta a eventos, junto con las respuestas a esos eventos. Una maquina de estados involucra otros elementos como son estados, transiciones (flujo de un estado a otro), eventos (que disparan una transición) y actividades (respuesta de una transición)



- **Elementos de agrupación**

Forman la parte organizativa de los modelos UML. El principal elemento de agrupación es el **paquete**, que es un mecanismo de propósito general para organizar elementos en grupos. Los elementos estructurales, los elementos de comportamiento, incluso los propios elementos de agrupación se pueden incluir en un paquete.

Paquete

Un paquete es puramente conceptual (sólo existe en tiempo de desarrollo). Gráficamente se representa como una carpeta conteniendo normalmente su nombre y, a veces, su contenido. Se emplea para organizar otros elementos en grupos

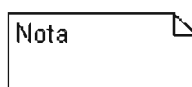


- **Elementos de anotación**

Los elementos de anotación son las partes explicativas de los modelos UML. Son comentarios que se pueden aplicar para describir, clasificar y hacer observaciones sobre cualquier elemento de un modelo.

Nota

El tipo principal de anotación es la **nota** que simplemente es un símbolo para mostrar restricciones y comentarios junto a un elemento o un conjunto de elementos.



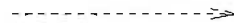
b. Relaciones

Existen cuatro tipos de relaciones entre los elementos de un modelo UML. *Dependencia, asociación, generalización y realización*, estas se describen a continuación:

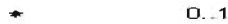
Dependencia

Es una relación semántica entre dos elementos en la cual un cambio a un elemento (el elemento independiente) puede afectar a la semántica del otro elemento (elemento dependiente). Se representa como una línea discontinua, posiblemente dirigida, que a veces incluye una etiqueta.

Asociación



Es una relación estructural que describe un conjunto de enlaces, los cuales son conexiones entre objetos. La agregación es un tipo especial de asociación y representa una relación estructural entre un todo y sus partes. La asociación se representa con una línea continua, posiblemente dirigida, que a veces incluye una etiqueta. A menudo se incluyen otros adornos para indicar la multiplicidad y roles de los objetos involucrados.



Generalización

Es una relación de especialización / generalización en la cual los objetos del elemento especializado (el hijo) pueden sustituir a los objetos del elemento general (el padre). De esta forma, el hijo comparte la estructura y

el comportamiento del padre. Gráficamente, la generalización se representa con una línea con punta de flecha vacía.



Realización

Es una relación semántica entre clasificadores, donde un clasificador especifica un contrato que otro clasificador garantiza que cumplirá. Se pueden encontrar relaciones de realización en dos sitios: entre interfaces y las clases y componentes que las realizan, y entre los casos de uso y las colaboraciones que los realizan. La realización se representa como una mezcla entre la generalización y la dependencia, esto es, una línea discontinua con una punta de flecha vacía.

Es una relación que implica que la parte realizante cumple con una serie de especificaciones propuestas por la clase realizada (interfaces).



c. Diagramas

Los diagramas se utilizan para representar diferentes perspectivas de un sistema de forma que un diagrama es una proyección del mismo. UML proporciona un amplio conjunto de diagramas que normalmente se usan en pequeños subconjuntos para poder representar las cinco vistas principales de la arquitectura de un sistema.

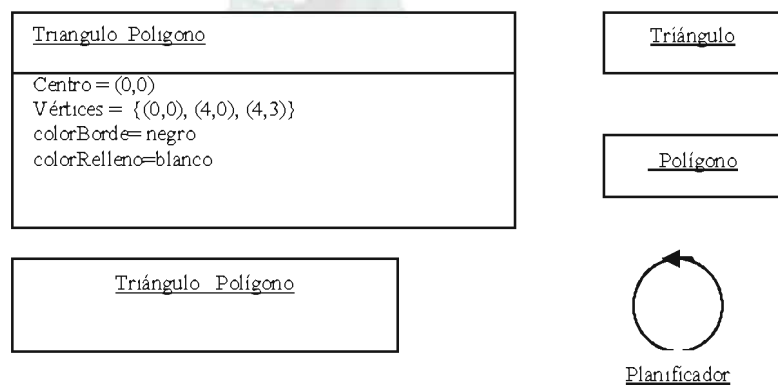
Diagramas de Clases

Muestran un conjunto de clases, interfaces y colaboraciones, así como sus relaciones. Estos diagramas son los más comunes en el modelado de sistemas orientados a objetos y cubren la vista de diseño estática o la vista de procesos estática (sí incluyen clases activas).

Diagramas de Objetos

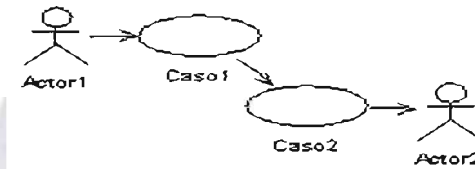
Muestran un conjunto de objetos y sus relaciones, son como fotos instantáneas de los diagramas de clases y cubren la vista de diseño estática o la vista de procesos estática desde la perspectiva de casos reales o prototípicos.

Análogo al diagrama de clases, muestra un conjunto de objetos y sus relaciones, pero a modo de vista instantánea de instancias de una clase en el tiempo.



Diagramas de Casos de Usos

Muestran un conjunto de casos de uso y actores (tipo especial de clases) implicados y sus relaciones. Cubren la vista estática de los casos de uso y son especialmente importantes para el modelado y organización del comportamiento. Son diagramas fundamentales en el modelado y organización del sistema



Diagramas de Secuencia y de Colaboración

Tanto los diagramas de secuencia como los diagramas de colaboración son un tipo de diagramas de interacción. Constan de un conjunto de objetos y sus relaciones, incluyendo los mensajes que se pueden enviar unos objetos a otros. Cubren la vista dinámica del sistema.

Diagrama de Secuencia

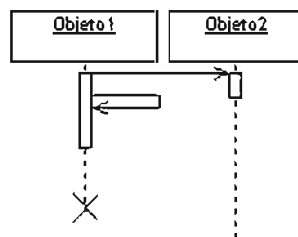
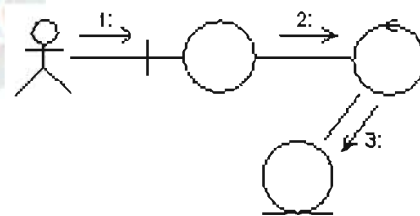


Diagrama de Colaboración

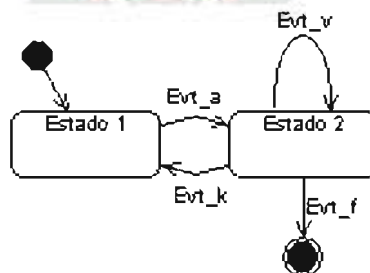


Los diagramas de secuencia enfatizan el ordenamiento temporal de los mensajes mientras que los diagramas de colaboración muestran la organización estructural de los objetos que envían y reciben mensajes. Los diagramas de secuencia se pueden convertir en diagramas de colaboración sin pérdida de información, lo mismo ocurren en sentido opuesto.

El diagrama de secuencia resalta la ordenación temporal de los mensajes, mientras que el de colaboración resalta la organización estructural de los objetos, ambos siendo equivalentes o isomorfos. En el diagrama de colaboración, se puede ver que los elementos gráficos no son cajas rectangulares, como cabría esperar, y en su lugar encontramos sus versiones adornadas. Estas versiones tienen como finalidad evidenciar un rol específico del objeto siendo modelado. En la figura encontramos de izquierda a derecha y de arriba abajo un Actor, una Interfaz, un Control (modela un comportamiento) y una Instancia (modela un objeto de dato)

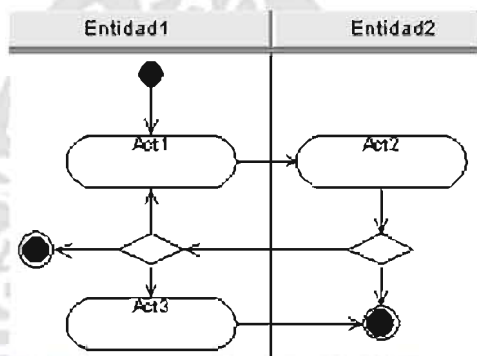
Diagramas de Estados

Muestran una máquina de estados compuesta por estados, transiciones, eventos y actividades. Estos diagramas cubren la vista dinámica de un sistema y son muy importantes a la hora de modelar el comportamiento de una interfaz, clase o colaboración, en base a eventos



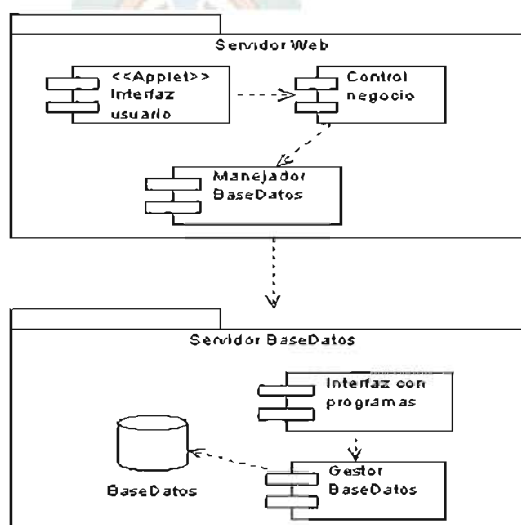
Diagramas de Actividades

Son un tipo especial de diagramas de estados que se centra en mostrar el flujo de actividades dentro de un sistema. Los diagramas de actividades cubren la parte dinámica de un sistema y se utilizan para modelar el funcionamiento de un sistema resaltando el flujo de control entre objetos.



Diagramas de Componentes

Muestra la organización y las dependencias entre un conjunto de componentes. Cubren la vista de la implementación estática y se relacionan con los diagramas de clases ya que en un componente suele tener una o más clases, interfaces o colaboraciones.



Diagramas de Despliegue

Representan la configuración de los nodos de procesamiento en tiempo de ejecución y los componentes que residen en ellos. Muestran la vista de despliegue estática de una arquitectura y se relacionan con los componentes ya que, por lo común, los nodos contienen uno o más componentes.

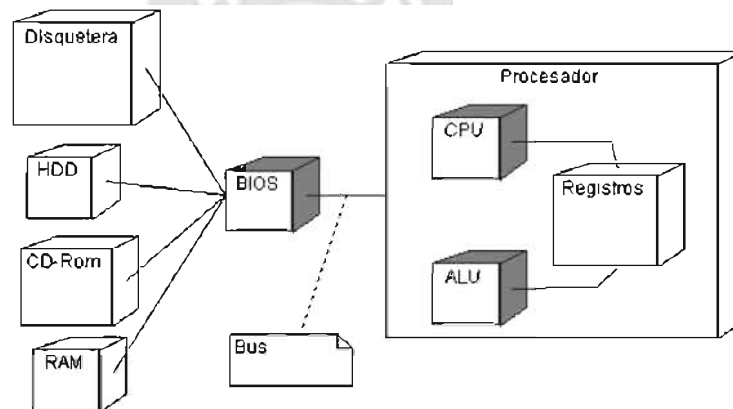


Diagrama de Despliegue

2.3.6. FASE DE TRANSICION

2.3.6.1. Mapa Navegacional

Como necesidad inmediata ante el desarrollo de una aplicación hipertextual, está la utilización de una metodología adecuada para diseñar y documentar el producto software. Surge así el estudio de las metodologías RMM y ADV chart con el fin de utilizarlas como método de diseño y documentación. Sin embargo, el estudio de estas metodologías no fue completamente satisfactorio. Veamos algunas de sus falencias.

1) La metodología RMM [ISBA95] exige que se obtenga el esquema E/R que gobierna la lógica de la aplicación, pero muchos de los módulos pertenecientes a la aplicación no quedan documentados bajo este esquema, como por ejemplo, todos los procesos gatillados por la ocurrencia de un evento en particular. Además, esta metodología es de poca utilidad cuando la volatilidad de la información es baja o de características estables, que es la situación de la información involucrada en el presente proyecto.

2) La utilización de ADV chart [MATU97] es bastante completa en la documentación de interfaces, maneja el concepto de evento, realiza una buena descripción de los elementos involucrados en cada pantalla y en cierta medida de la navegación a través de la aplicación, pero no involucra la naturaleza de los elementos insertos en cada despliegue o pantalla que compone la interfaz (texto, bitmap, vídeo, etc.).

Por estos motivos, se propone una nueva metodología de diseño que pretende Documentar el esquema de navegación de la aplicación y el diseño de la interfaz. Esta metodología nace inspirada en el diseño navegacional de RMM y la claridad para documentar interfaces de ADV chart.

Restricciones:

La metodología propuesta posee dos restricciones esenciales:

- a) La información debe ser estable o muy poco volátil.
- b) La información debe ser altamente estructurada y jerarquizada.

2.3.6.2. Elementos de la metodología

Esta metodología se aplica en tres niveles: los que corresponden al diseño de navegación de la aplicación, al diseño de la interfaz y documentación de procesos.

A) Navegación:

Para el diseño de la navegación, se definen una serie de elementos que resultan de la comprensión de la información involucrada, como si ésta fuera compuesta por entidades de contenido y por entidades generadoras de un cambio del contenido expuesto. Se definen dos elementos básicos:

Contenidos: Corresponden a las unidades o núcleos de información, tales como explicaciones, descripciones, diagramas, relatos, etc., que serán presentados en el sistema.

Representación:



Nombre Contenido

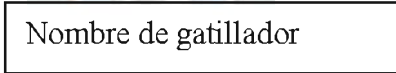
Si un contenido en particular, se repite dentro de la navegación, éste se representará como sigue: para evidenciar de que se trata del mismo elemento de contenido.



Nombre Contenido

Gatilladores: Corresponden a las unidades o elementos que desencadenan la aparición o visualización de un contenido.

Representación:



Nombre de gatillador

Si un Gatillador en particular, se repite dentro de la navegación, éste se representará como sigue:

Nombre de gatillador	
----------------------	--

Menú: Corresponde a una agrupación de gatilladores.

Nombre Gatillador
Nombre Gatillador
:
:
Nombre Gatillador

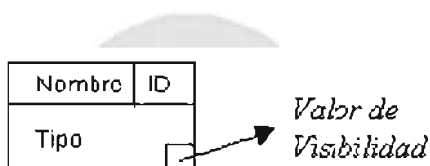
B) Interfaz:

El diseño de la interfaz está directamente relacionado con los elementos presentes en el diagrama resultante del diseño de navegación, por lo que la relación es prácticamente uno a uno entre cada diseño de interfaz (pantalla o despliegue) y los elementos de contenido. No se pretende con esta metodología desligar la ocurrencia de cada despliegue con la ejecución de ciertos procedimientos, ya que son muchas veces ellos, en aplicaciones de tipo multimedial, los que determinan la ocurrencia de las visualizaciones o despliegues.

i) Se definen los siguientes elementos:

Elementos de Contenido: Corresponden a los entes de información que participan en el despliegue.

Representación



Nombre: Nombre del contenido.

ID: Campo opcional que asocia un número al elemento, es útil para tener una apreciación de los elementos de media involucrados. Se debe generar, si corresponde, una biblioteca con los elementos de media y sus respectivos números o códigos.

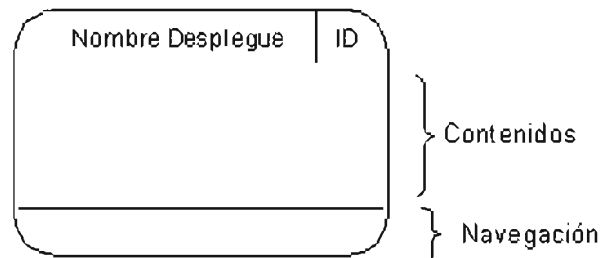
Tipo: Describe la naturaleza del contenido desplegado:

	Tipo	Descripción
BMP	Bitmap	Imágenes o mapas de bits.
FLM	Film	Animaciones, sucesión de mapas de bits o vídeos digitales.
AUD	Audio	Señales audibles.
TXT	Texto	Textos de contenido estático.
FLD	Campo	Textos de contenido variable.

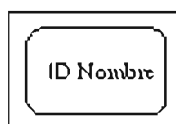
Valor de visibilidad: Determina la propiedad de visibilidad del elemento. Un elemento puede estar presente en el despliegue pero no ser visualizado. Se representa de la siguiente manera:

- : No visible.
- : Visible.
- : Visibilidad variable, estado inicial visible. La flecha indica al evento que cambia su estado.
- : Visibilidad variable, estado inicial invisible. La flecha indica al evento que cambia su estado.

Despliegue: Corresponde a la representación de un despliegue. Se representa:



En la zona correspondiente a los contenidos se ubican los elementos de contenido que despliegan la información. En la zona de Navegación se ubican los elementos de contenido que permiten la navegación a través de los distintos despliegues.



Un despliegue puede ser incluido dentro de otro. Se abrevia como sigue:

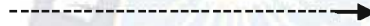
Vínculo: relaciona dos elementos en el sentido de la navegación.

Representación:



Vínculo de dependencia: relacionan un contenido con un gatillador, únicamente cuando el gatillador depende de la información de un contenido.

Representación:



Los flujos de retorno dentro de la navegación se obvian, ya que como es habitual en cualquier hipertexto, siempre es posible retroceder a los despliegues de origen.

2.4. METRICAS DE CALIDAD

Hablar de calidad del software implica la necesidad de contar con parámetros que permitan establecer los niveles mínimos que un producto de este tipo debe alcanzar para que se considere de calidad.

El problema es que la mayoría de las características que definen al software no se pueden cuantificar fácilmente; generalmente, se establecen de forma cualitativa, lo que dificulta su medición, ya que se requiere establecer métricas que permitan evaluar cuantitativamente cada característica dependiendo del tipo de software que se pretende calificar. En este sentido se han realizado muchos trabajos que establecen propuestas para el establecimiento de los factores cualitativos que afectan la calidad del software. Entre los principales

están los factores de calidad de McCall y aquellos propuestos por Hewlett-Packard.

En este trabajo se menciona un esquema general del estándar ISO 9126, con el fin de mostrar los elementos que deben considerarse en la evaluación de calidad de los productos de software de acuerdo a este estándar, de modo que todo aquél que se interese en aplicar modelos de calidad en la producción de software pueda generar sus propias métricas bajo la guía de este estándar.

2.3.1. Modelo de Calidad Establecido por el estándar ISO 9126

La ISO, bajo la norma ISO-9126, ha establecido un estándar internacional para la evaluación de la calidad de productos de software el cual fue publicado en 1992 con el nombre de *“Information technology –Software product evaluation: Quality characteristics and guidelines for their use”*, en el cual se establecen las características de calidad para productos de software. El estándar ISO-9126[7] establece que cualquier componente de la calidad del software puede ser descrito en términos de una o más de seis características básicas, las cuales son: funcionalidad, confiabilidad, usabilidad, eficiencia, mantenibilidad y portatilidad; cada una de las cuales se detalla a través de un conjunto de subcaracterísticas que permiten profundizar en la evaluación de la calidad de productos de software. La tabla 1 muestra la pregunta central que atiende cada una de estas características.

MANTENIBILIDAD

La fase de mantenimiento se centra en el cambio que va asociado a la corrección de errores, a las adaptaciones requeridas a medida que evoluciona el entorno del software y cambios debidos a las mejoras producidas por los requisitos cambiantes del cliente [PRS02].

FUNCIONALIDAD

La medición del tamaño funcional es una parte importante del proceso de desarrollo de software ya que es uno de los factores clave que potencialmente afecta el coste y el tiempo de los proyectos software. El tamaño de sistema software es derivado de la cuantificación de los requisitos funcionales. [ABR05].

MÉTRICAS ORIENTADAS A LA FUNCIÓN. Son medidas indirectas del software y del proceso por el cual se desarrolla, las métricas orientadas a la función se centran en la funcionalidad o utilidad del programa.

Las métricas orientadas a la función fueron el principio propuestas por Albercht quien sugirió un acercamiento a la medida de la productividad denominado método del punto de función. Los puntos de función que obtienen utilizando una función empírica basando en medidas cuantitativas del dominio de información del software y valoraciones subjetivos de la complejidad del software.

Cálculo de métricas de punto de función.

FACTOR DE PONDERACIÓN					
Parámetro de medición	Cuenta	Simple	Medio	Complejo	
Número de entradas de usuario	<input type="text"/>	x 3	4	6	= <input type="text"/>
Número de salidas de usuario	<input type="text"/>	x 4	5	7	= <input type="text"/>
Número de peticiones de usuario	<input type="text"/>	x 3	4	6	= <input type="text"/>
Número de archivos	<input type="text"/>	x 7	10	15	= <input type="text"/>
Número de interfaces externas	<input type="text"/>	x 5	7	10	= <input type="text"/>
Cuenta = Total	→				<input type="text"/>

Fuente:[PRS02]

Se determinan 5 características del ámbito de la información y los cálculos aparecen en la posición apropiada de la tabla. Los valores del ámbito de información están definidos de la siguiente manera.

1. *Números de entrada de usuario.* Se cuenta cada entrada del usuario que proporcione al software diferentes datos orientados a la aplicación. Las entradas deben ser distinguidas de las peticiones que se contabilizan por separado.
2. *Número de salida del usuario.* Se encuentra cada salida que proporciona al usuario información orientada a la aplicación. En este contexto las salidas se refieren a informes, pantalla, mensajes de error. Los elementos de datos individuales dentro de un informe se encuentran por separado.
3. *Números de peticiones al usuario.* Una petición está definida como una entrada interactiva que resulta de la generación de algún tipo de respuesta en forma de salida interactiva. Se cuenta cada petición por separado.

1. *Número de archivos.* Se cuenta cada archivo maestro lógico, o sea una agrupación lógica de datos que puede ser una parte en una gran base de datos o un archivo independiente.
2. *Número de interfaces externas.* Se cuentan todas las interfaces legibles por la maquina por ejemplo: archivos de datos, en cinta o discos que son utilizados para transmitir información a otro sistema.

Cuando han sido recogidos los datos anteriores se asocian el valor de complejidad a cada cuenta. Las organizaciones que utilizan métodos de puntos de función desarrollan criterios para determinar si una entrada es denominada simple, media o compleja. No obstante la determinación de la complejidad es algo subjetivo.

Para calcular los puntos de función se utiliza la siguiente relación.

$$PF = \text{Cuenta total} * (0.65 + 0.01 * \sum Fi)$$

Donde:

CUENTA_TOTAL es la suma de todas las entradas de PF obtenidas de la tabla anterior.

Evaluar cada factor en escala 0 a 5 como se muestra en la siguiente tabla:

Tabla 2.2 Factor de escala

0	1	2	3	4	5
Sin influencia	Incidental	Moderado	Medio	Significativo	Esencial

Fuente: [PRS02]

Valores de ajuste de complejidad	Fi
1. ¿Requiere el sistema copias de seguridad y de recuperación fiables?	
2. ¿Se requiere comunicación de datos?	
3. ¿Existen funciones de procesamiento distribuido?	
4. ¿Es crítico el rendimiento?	
5. ¿Se ejecutará el sistema en un entorno operativo existente y fuertemente utilizado?	
6. ¿Requiere el sistema entrada de datos interactiva?	
7. ¿Requiere la entrada de datos interactiva que las transacciones de entrada se lleven a cabo sobre múltiples pantallas u operaciones?	
8. ¿Se actualiza los archivos maestros de forma interactiva?	
9. ¿Son complejas las entradas, las salidas, los archivos o las peticiones?	
10. ¿Es complejo el procesamiento interno?	
11. ¿Se ha diseñado el código para ser reutilizable?	
12. ¿Están incluidas en el diseño la conversión y la instalación?	
13. ¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?	
14. ¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizada por el usuario?	
SUMATORIA Fi	

Los valores constantes de la ecuación anterior y los factores de peso aplicados en las encuestas de los ámbitos de información han sido determinados empíricamente.

Portatilidad

En este caso, se refiere a la habilidad del software de ser transferido de un ambiente a otro, y considera los siguientes aspectos:

- Adaptabilidad. Evalúa la oportunidad para adaptar el software a diferentes ambientes sin necesidad de aplicarle modificaciones.
- Facilidad de Instalación. Es el esfuerzo necesario para instalar el software en un ambiente determinado.
- Conformidad. Permite evaluar si el software se adhiere a estándares o convenciones relativas a portabilidad.
- Capacidad de reemplazo. Se refiere a la oportunidad y el esfuerzo usado en sustituir el software por otro producto con funciones similares.





CAPÍTULO III

DESARROLLO DEL PROYECTO

3. Diseño del Sistema

En este capítulo se mostrara el desarrollo del Análisis y Desarrollo del sistema Control y Seguimiento de Establecimientos de Salud SEDES LAPAZ.

3.1. ANÁLISIS, DESARROLLO E IMPLEMENTACIÓN

En este punto se encarga de analizar las funciones que realiza la unidad de Acreditación y Certificación y sus respectivos servicios para un mejor funcionamiento en el acceso de manejo de información

Una vez investigada la metodología, se aplicara la metodología RUP, utilizando como herramienta de descripción al UML.

En este apartado se detalla la planificación inicial del proyecto para la fase de inicio y la fase de elaboración (según la definición de la metodología RUP) y el diario de ejecución del proyecto, sesión tras sesión de trabajo.

3.2. FASES DE LA METODOLOGÍA RUP

En las fases de la metodología RUP se describen cada de las etapas que constituyen la configuración del RUP y que se utilizará para el proyecto.

3.2.1 Modelado del Negocio

Un Modelo de Negocios define una estructura de información adecuada, que permite contar con un espacio que dé a conocer la información generada por los diferentes actores

El diagrama que representa los diferentes subsistemas en los que se ha dividido la empresa a nivel de abstracción es el siguiente

La siguiente figura muestra el diagrama de modelado de negocios del sistema de Control y Seguimiento de Establecimientos de Salud. Y sus respectivos subsistemas:

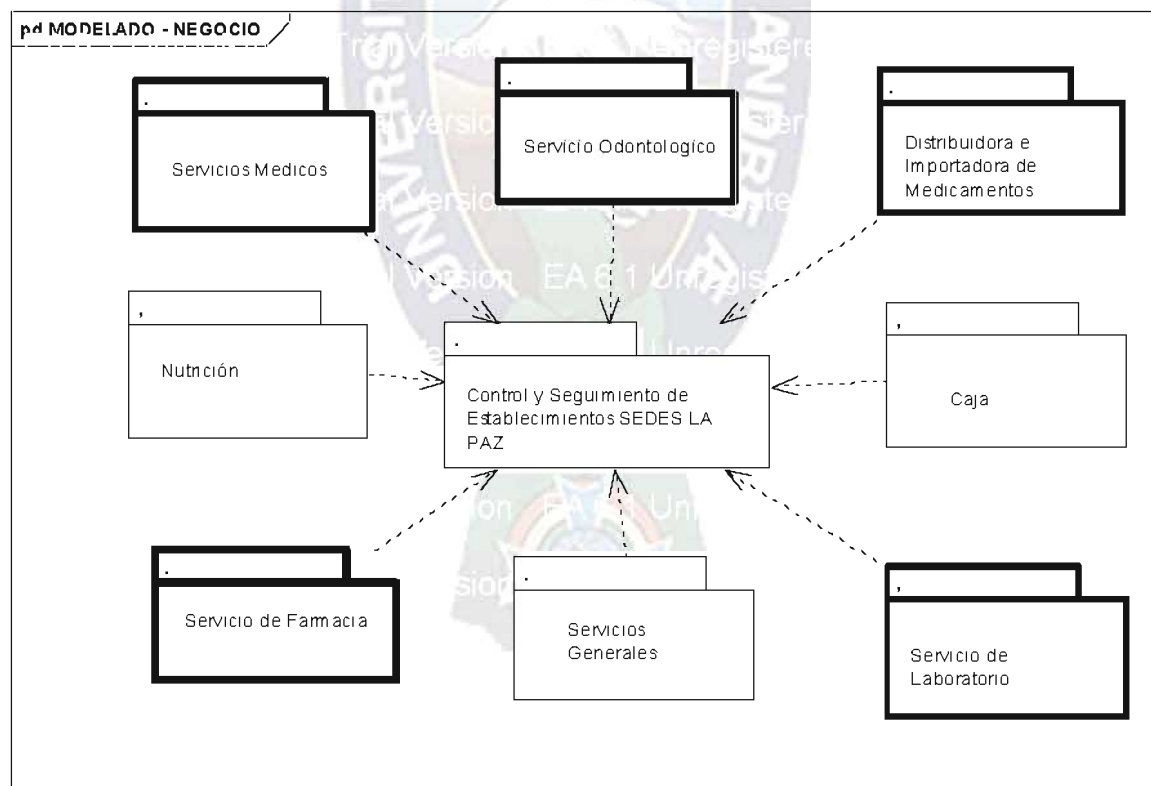


Fig. 3.1. Modelado de Negocios

3.2.1.1. Diagrama de casos de uso

Muestran un conjunto de casos de uso y actores (tipo especial de clases) y sus relaciones. Cubren la vista estática de los casos de uso y son especialmente importantes para el modelado y organización del comportamiento.

La interacción con distintos elementos externos se denominan actores que forman parte del sistema, están formados por:

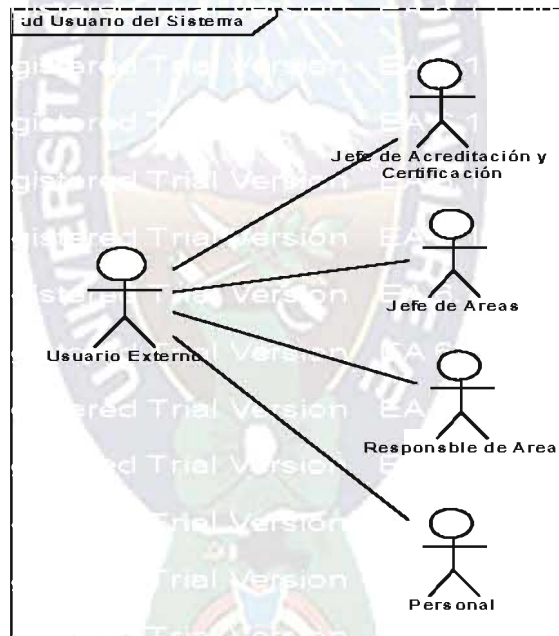


Fig.3.2. Diagrama de usuario del sistema

Mostraremos a continuación los casos de uso más importantes para el sistema de manera general:

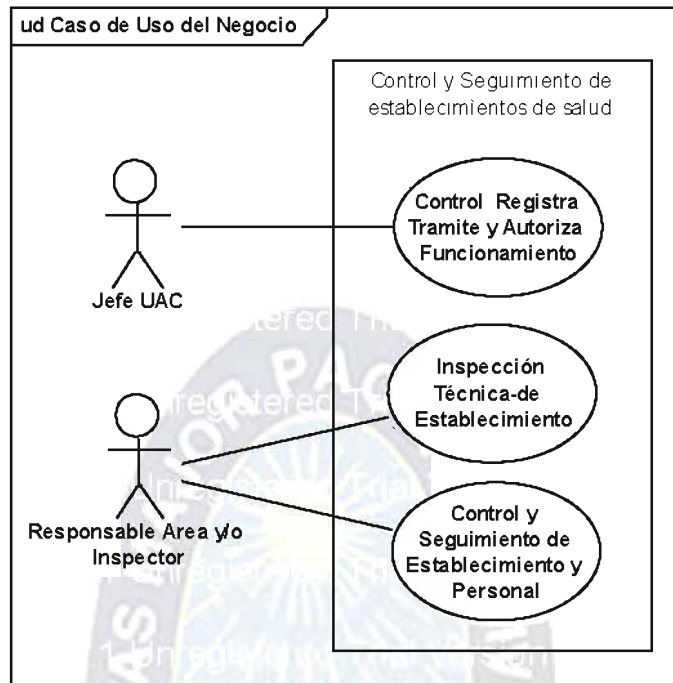


Fig. 3.3. Diagrama de casos de uso del Negocio

- Control , Registra Tramite y Autoriza Funcionamiento
- Inspección Técnica-de Establecimiento
- Control de Establecimiento y Personal

3.2.1.2 Modelo de Objetos

Muestran un conjunto de objetos y sus relaciones, son como fotos instantáneas de los diagramas de clases y cubren la vista de diseño estática o la vista de procesos estática desde la perspectiva de casos reales o prototipos.

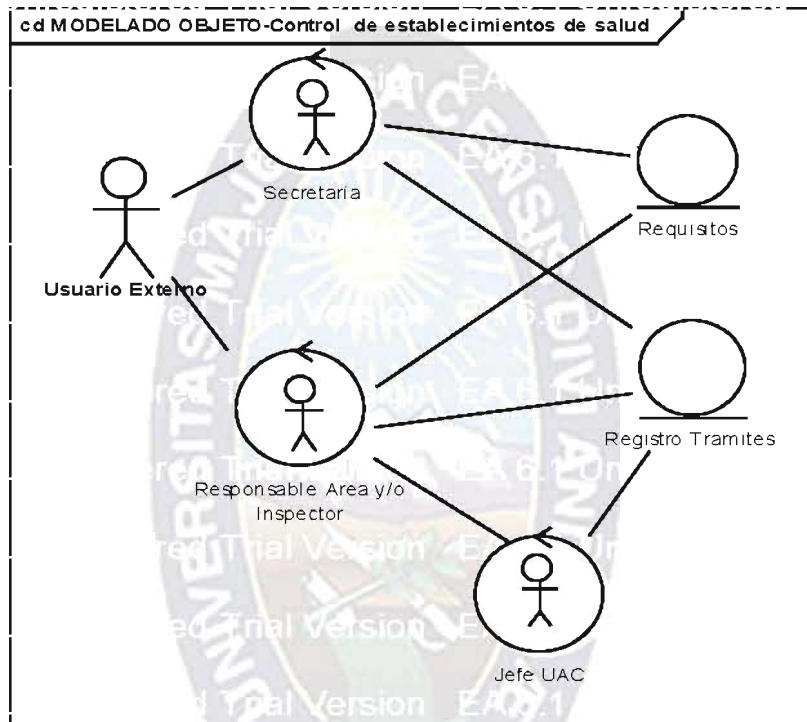


Fig. 3.4. Modelo de objetos de trámite Apertura, renovación y traslado

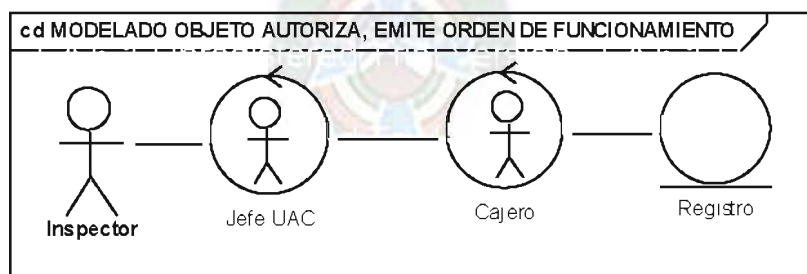


Fig. 3.5. Modelo de objetos Aceptación registró Trámite

Caso de Uso: Tramite Acreditación

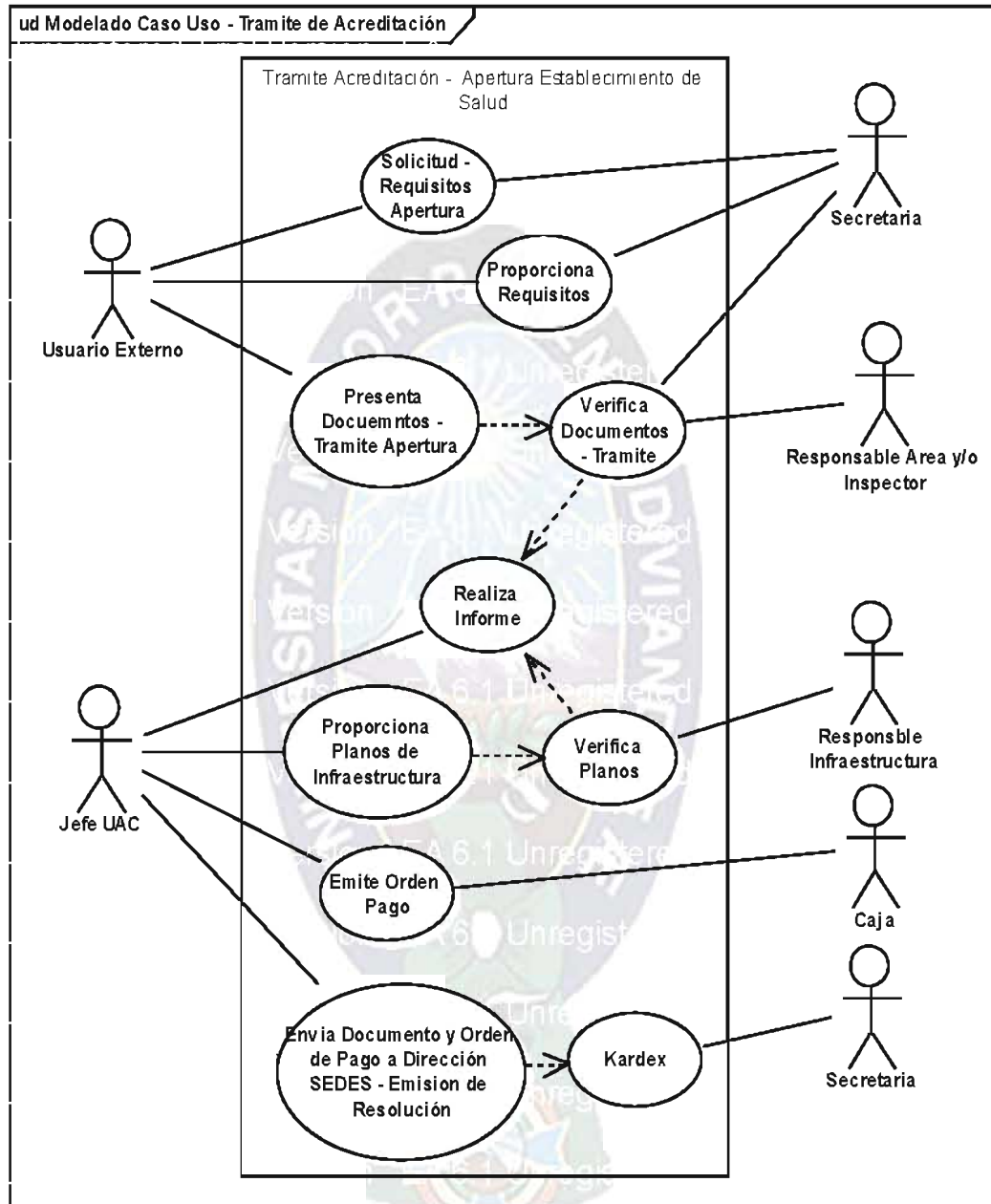


Fig. 3.6. Caso de Uso Tramite Acreditación

Subsistema Inspección Técnica-de Establecimiento

Caso de Uso: Inspección de Establecimientos de Salud

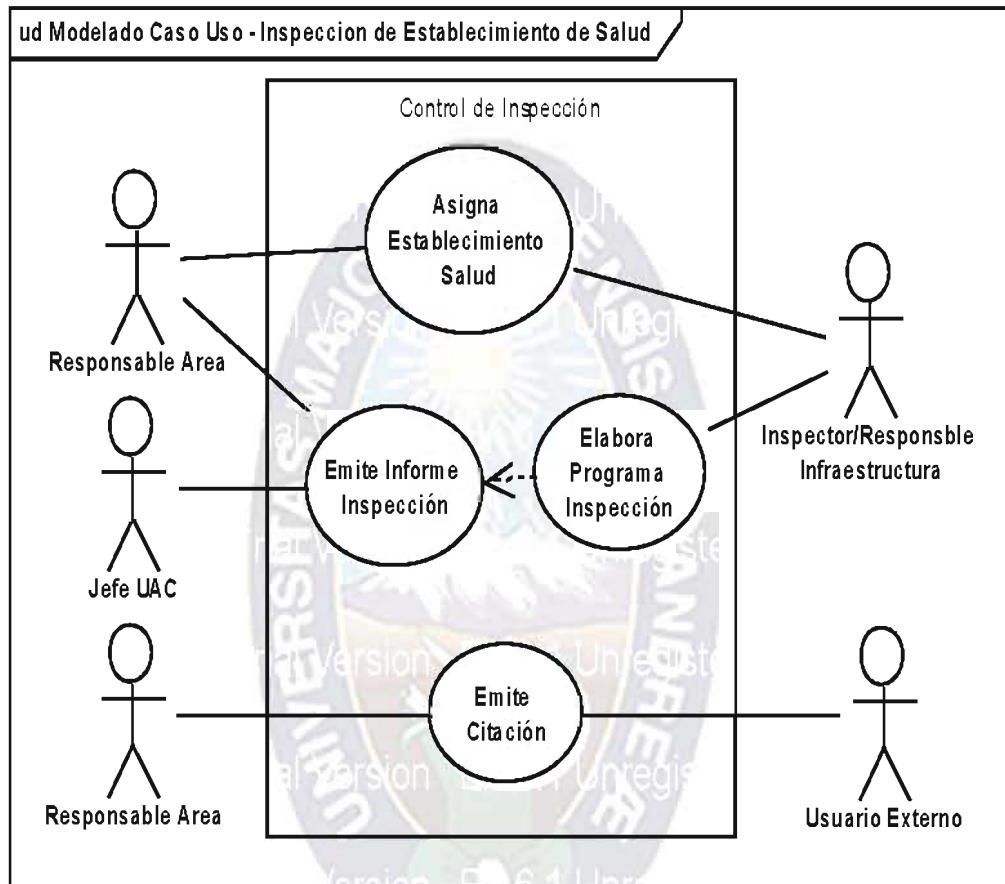


Fig. 3.9. Inspección de Establecimientos de Salud

3.2.2 Diagramas de Actividad

Son tareas que deben ser realizadas, cada actividad puede ser seguida por otra actividad. A continuación se observa los diagramas de actividades.

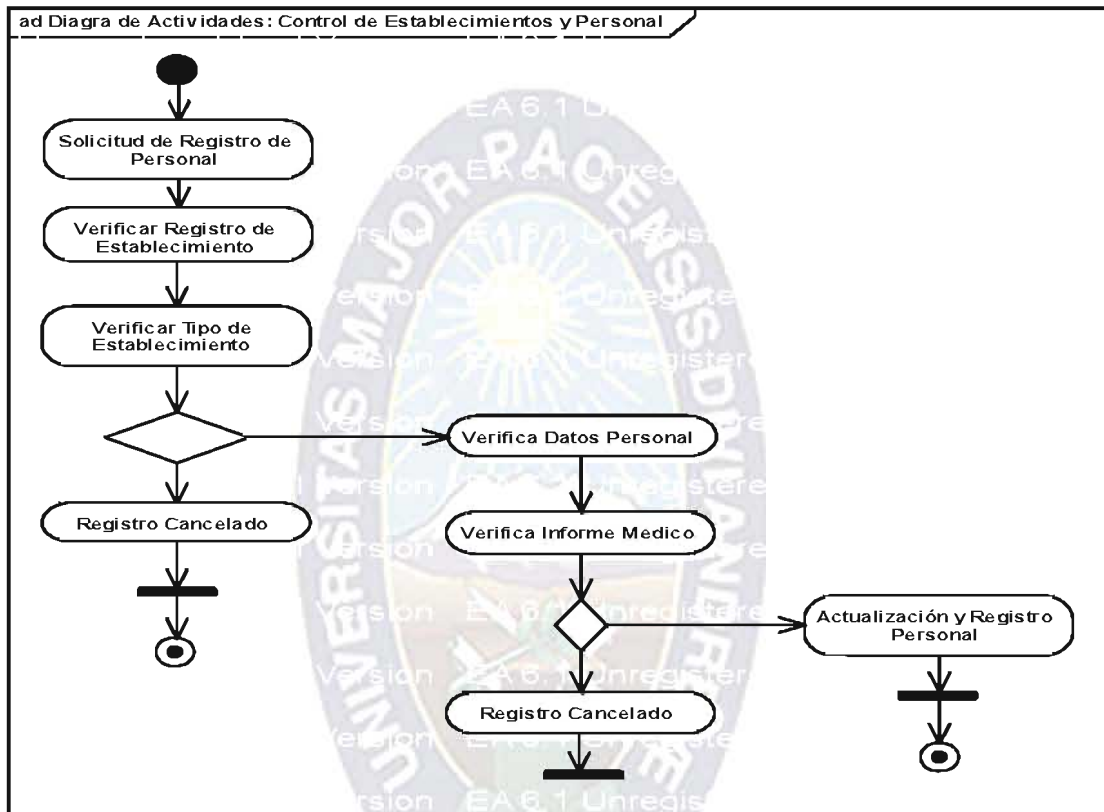


Fig.3.12. Diagrama de Actividades Control de establecimientos y Personal

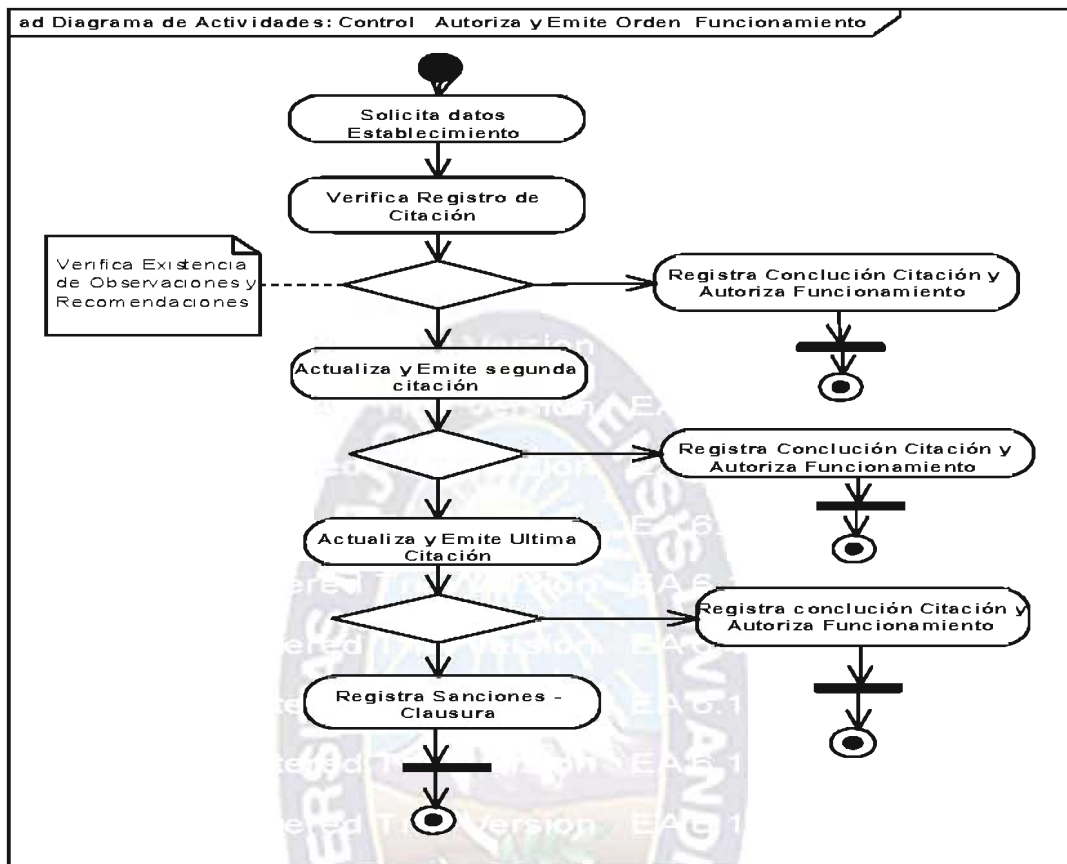


Fig. 3.13. Diagrama de Actividades Control Autoriza y Emite orden de Funcionamiento

3.2.3 Modelado de Análisis

Este modelo consiste en obtener una visión del sistema modelando los requisitos, es por ese motivo se utiliza los diagramas de secuencia.

3.2.3.1 Diagramas de Secuencia

Tanto los diagramas de secuencia como los diagramas de colaboración son un tipo de diagramas de interacción. Constan de un conjunto de objetos y sus relaciones, incluyendo los mensajes que se pueden enviar unos objetos a otros. Cubren la vista dinámica del sistema. Los diagramas de secuencia

enfatan el ordenamiento temporal de los mensajes mientras que los diagramas de colaboración muestran la organización estructural de los objetos que envían y reciben mensajes. Los diagramas de secuencia se pueden convertir en diagramas de colaboración sin pérdida de información, lo mismo ocurren en sentido opuesto.

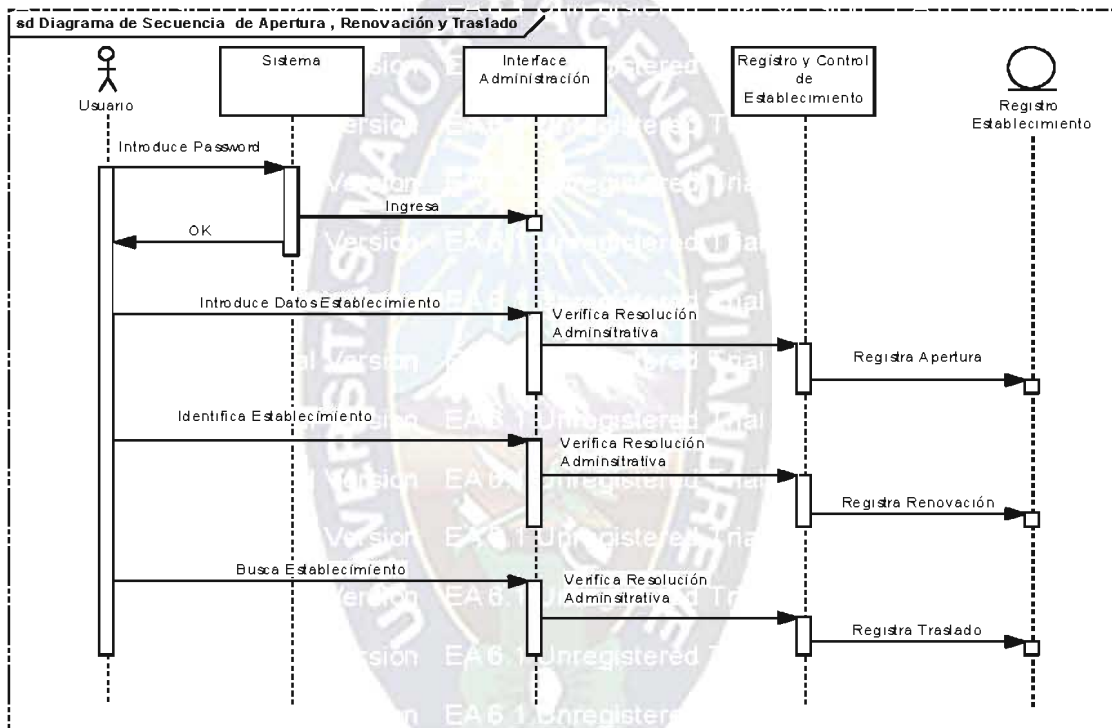


Fig. 3.15. Diagrama de Secuencias de Apertura, Renovación y Traslado

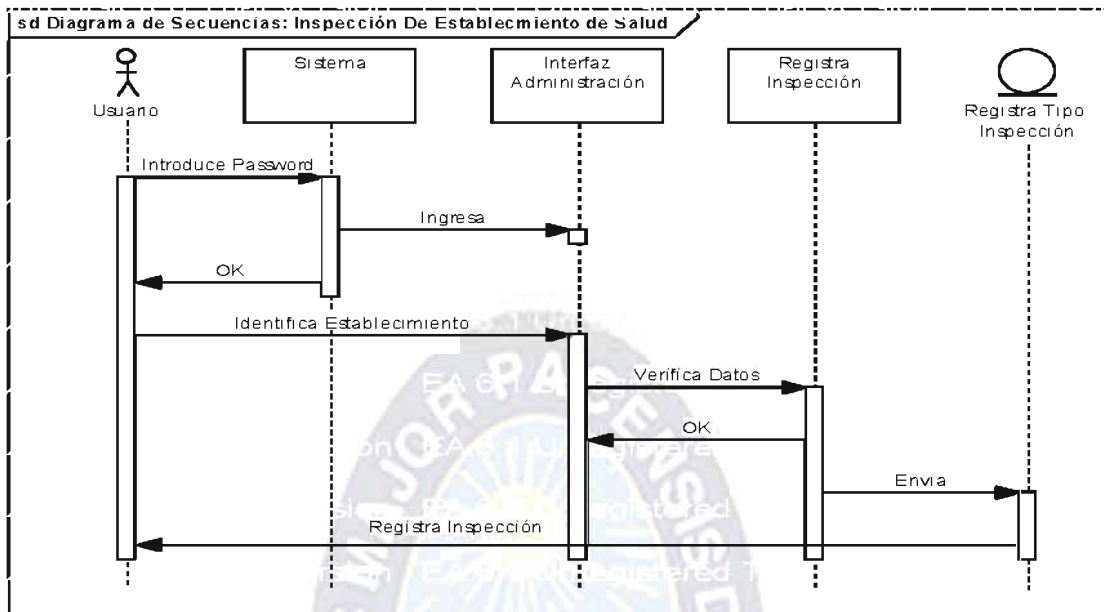


Fig. 3.15. Diagrama de Secuencias de Inspección Técnica de Establecimientos

3.2.4 Modelado de Diseño

A continuación se presentan los modelos definidos en RUP como modelo de datos y modelo de análisis / diseño. Constará de un diagrama de clases en el que se muestran tan sólo las clases generadas a partir de los casos de uso incorporados a la aplicación hasta la segunda iteración de la fase de construcción, y de un modelo de datos (modelo relacional) donde se muestran las entidades que participan en las relaciones definidas en el proyecto (teniendo en cuenta de nuevo que se alcanzó únicamente la segunda iteración de la fase de construcción).

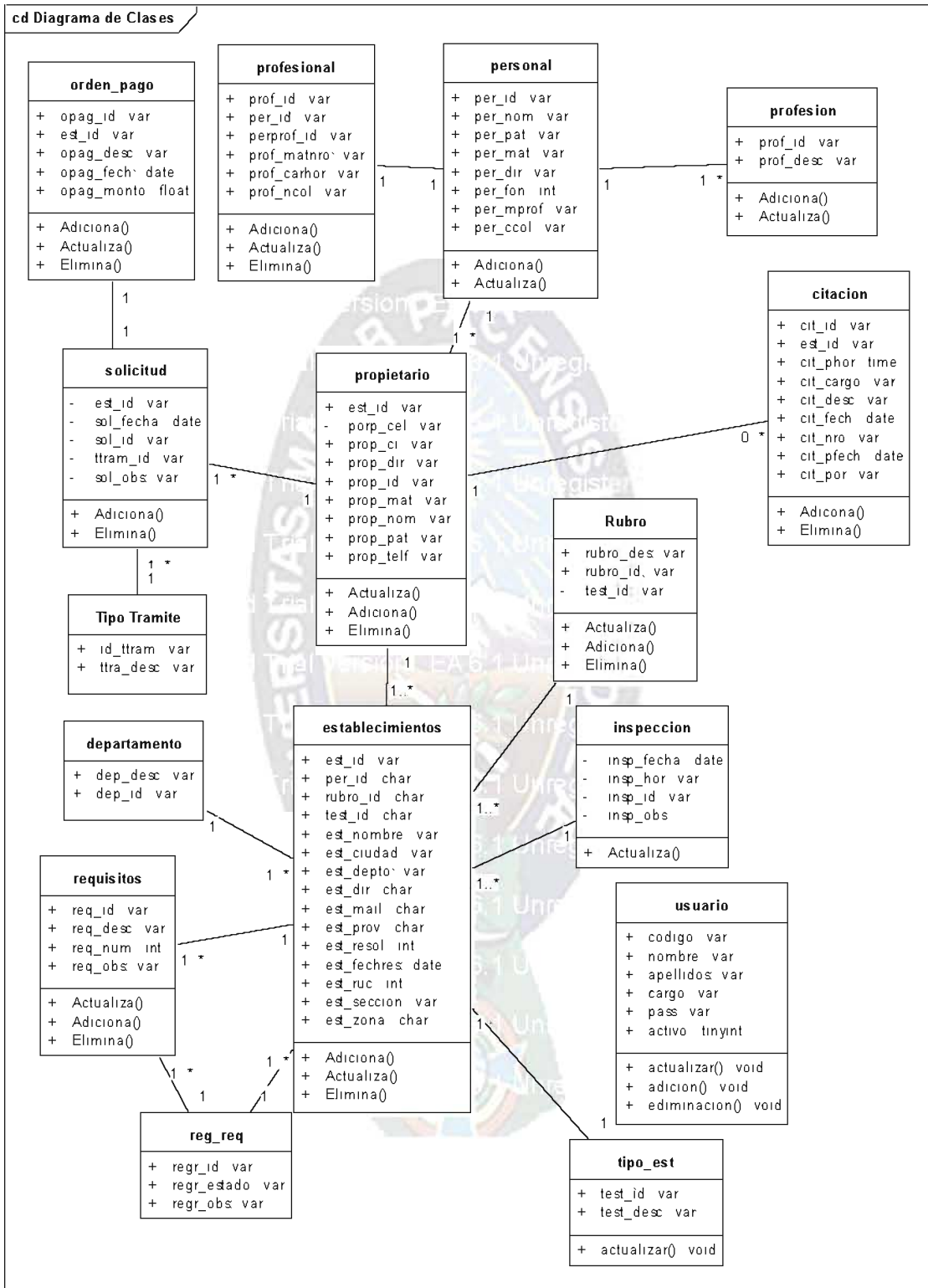


Figura 3.16: Diagrama de Clases

3.2.5 Modelo Relacional

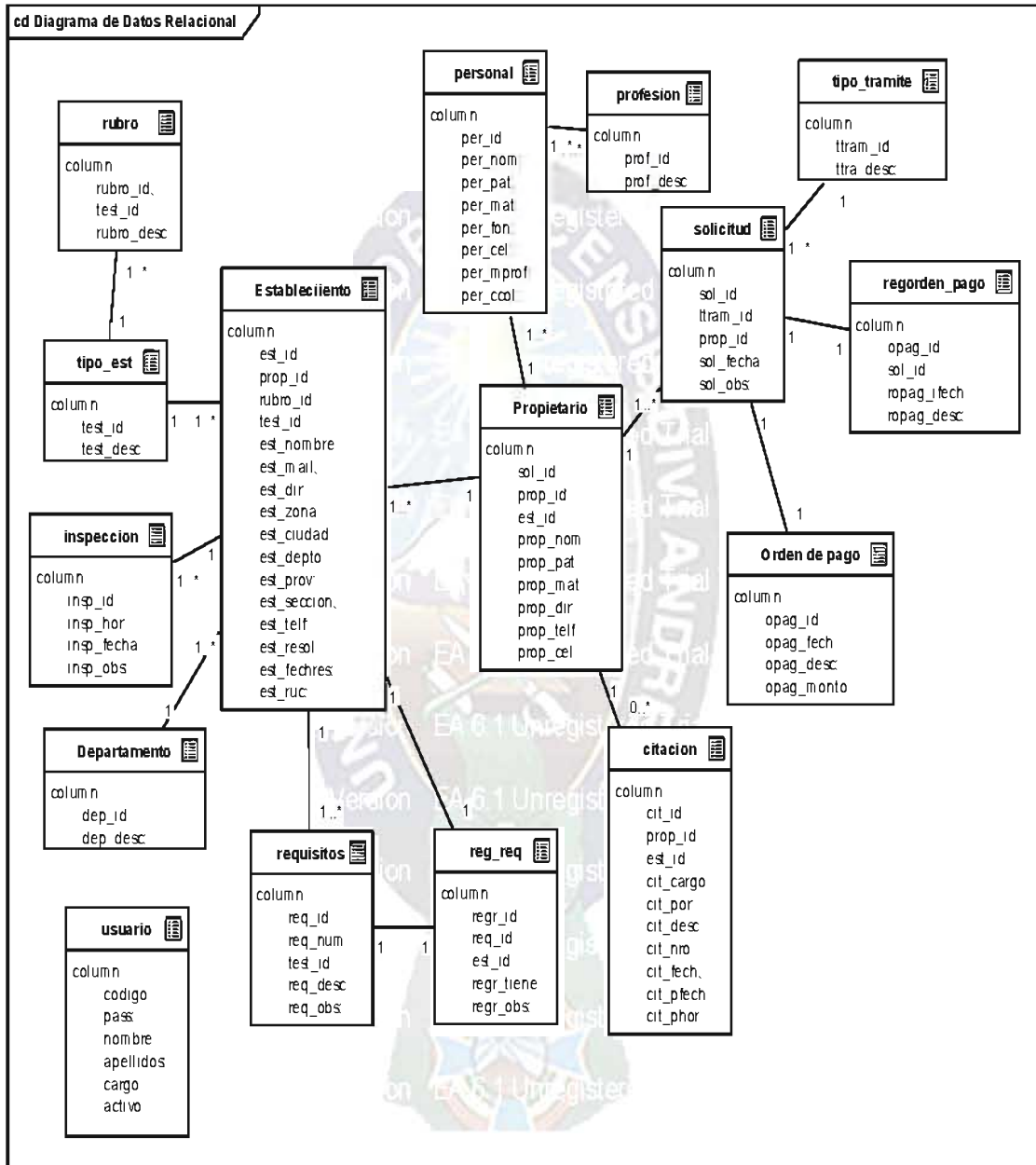


Figura 3.17: Diagrama Datos Relacional

3.2.5 Modelado de Implementación

3.2.5.1 Diagrama de Componentes

Muestra la organización y las dependencias entre un conjunto de componentes. Cubren la vista de la implementación estática y se relacionan con los diagramas de clases ya que en un componente suele tener una o más clases, interfaces o colaboraciones

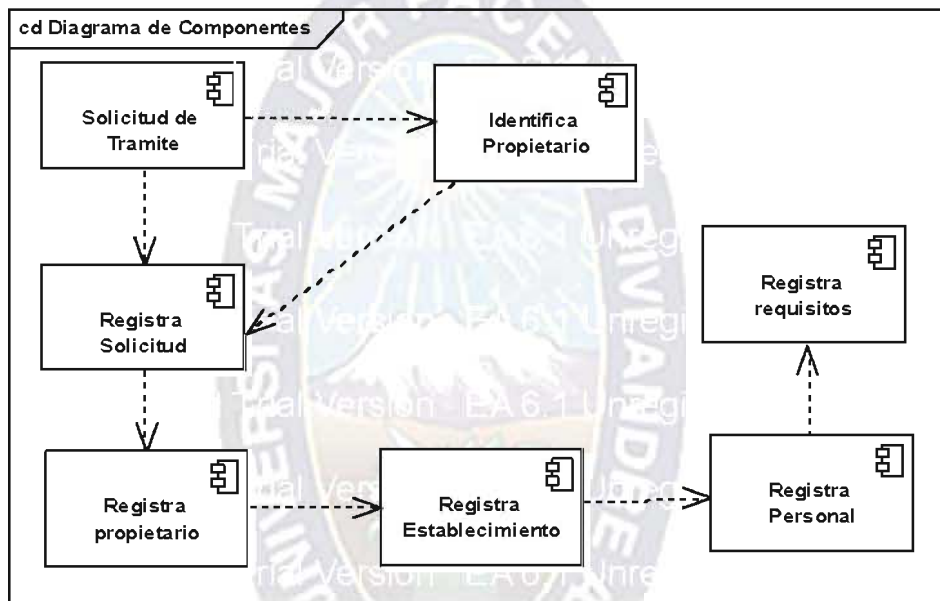


Figura 3.18. Diagrama de componentes: Seguimiento y Control Tramite

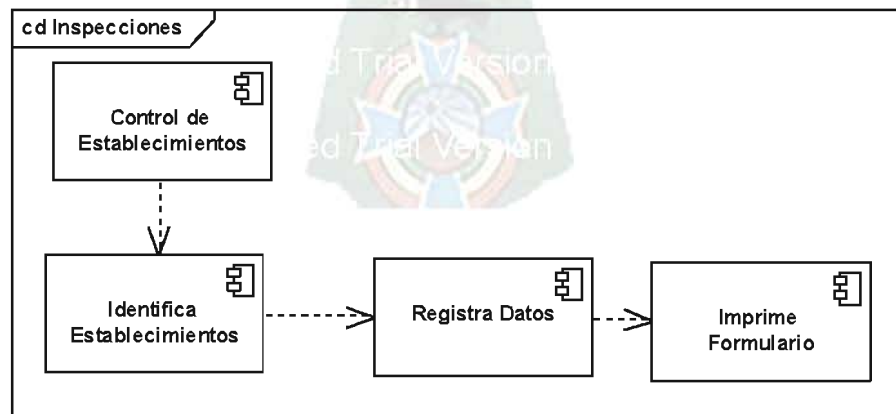


Figura 3.19. Diagrama de componentes: Seguimiento de inspecciones

3.2.5.2 Diagrama de Despliegue

Representan la configuración de los nodos de procesamiento en tiempo de ejecución y los componentes que residen en ellos. Muestran la vista de despliegue estática de una arquitectura y se relacionan con los componentes ya que, por lo común, los nodos contienen uno o más componentes

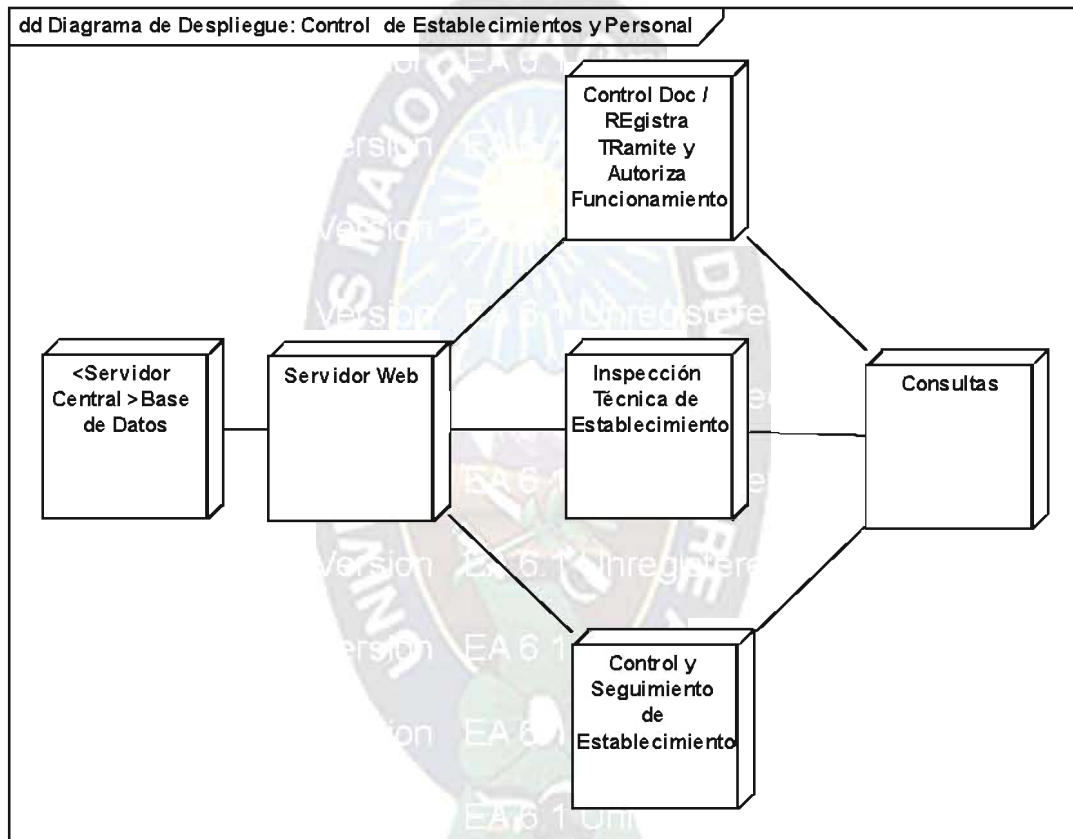


Figura 3.20. Diagrama de despliegue: Control de Establecimientos de Salud

3.2.5.3 Pantallas del Sistema

El diseño de la interfaz de usuario fue desarrollado de manera transparente al usuario, de tal manera que puede ser utilizado incluso por personas que no tengan conocimiento del área, esto con un mínimo de capacitación.

A continuación se presentan las pantallas más significativas del sistema.



Figura 3.21: Autenticación de Usuario
Fuente: Elaboración Propia

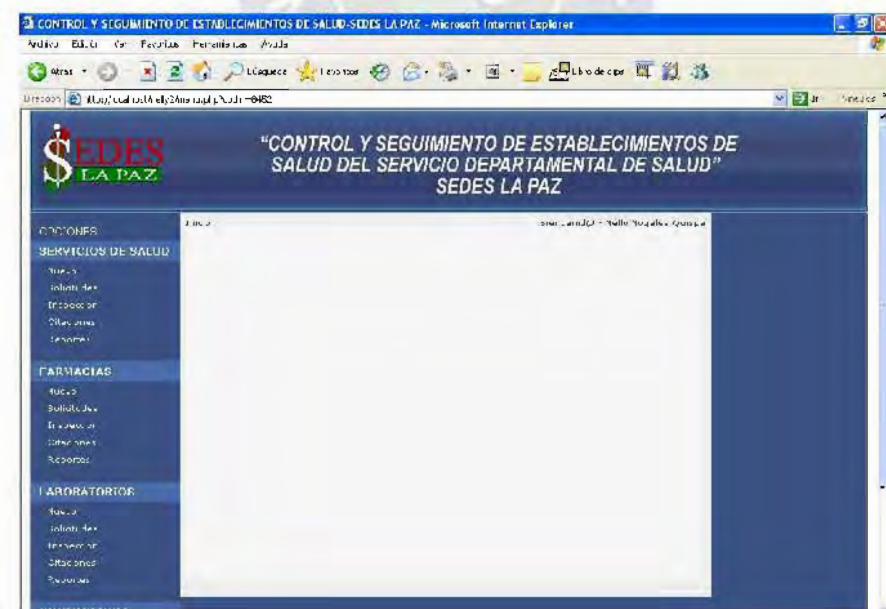


Figura 3.22: Pantalla Principal
Fuente: Elaboración Propia

CONTROL Y SEGUIMIENTO DE ESTABLECIMIENTOS DE SALUD SEDES LA PAZ - Microsoft Internet Explorer

Inicio / Admin / Personal / Datos Personal Bienvenido Heidy Nogales Quiroga

Solicitud Propietario Establecimiento Personal Registros

REGISTRO TRAMITE

Tipo de Trámite: APERTURA

Fecha: 16 ene 2010

DATOS PROPIETARIO

C.I.: 3465180

Matr. Prof.: 11811 Céd. Leg.: 1985

Nombre: NELLY

Apellido: NOGALES Matrimo.: JLISPE

Teléfono: 226142 Móvil: 226411

Dirección: AV. PERIFERICA

Figura 3.23: formulario de Llenado de datos de Propietario de Establecimiento
Fuente: Elaboración Propia

CONTROL Y SEGUIMIENTO DE ESTABLECIMIENTOS DE SALUD SEDES LA PAZ - Microsoft Internet Explorer

Inicio / Admin / Personal / Datos Personal Bienvenido Heidy Nogales Quiroga

Solicitud Propietario Establecimiento Personal Registros

DATOS ESTABLECIMIENTO

Solicitud: 9

Establecimiento: FARMACIAS

Rubro: FARMACIA FARMACIA

Razon Social: FARMACIA

Email: nogales@hotmail.com

Dirección: Av. Per...

Zona: U. ALLENDEGUA

Ciudad: LA PAZ

Teléfono: 226142

Departamental: LA PAZ

Resolucion No.: 2345

Fecha de Registro: 16 ene 2010

Provincia: MURILLO

Municipio: MURILLO

RUC: 22442

Figura 3.24: Formulario de Llenado de datos de Establecimiento
Fuente: Elaboración Propia

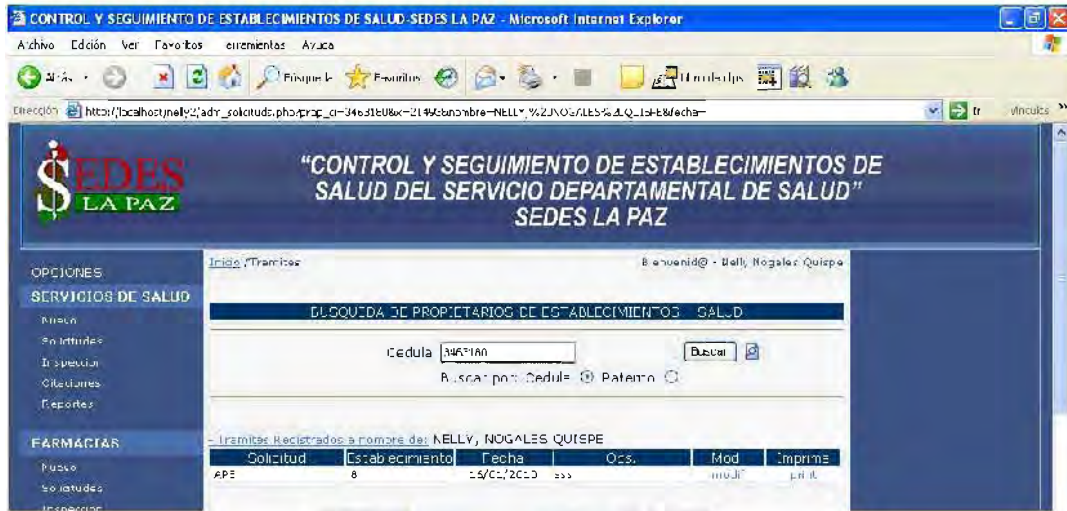


Figura 3.25: Formulario de solicitudes
Fuente: Elaboración Propia



Figura 3.26: Formulario de impresión
Fuente: Elaboración Propia

3.2.5 Modelado de Pruebas

Las pruebas del software son un elemento esencial para la garantía de calidad del software y representa una revisión final de las especificaciones, del diseño y de la codificación.

En esta etapa se lleva a cabo la verificación y validación del sistema.

La verificación responde a la pregunta ¿Se está construyendo el software correctamente?

La validación responde a la pregunta ¿Se está construyendo el producto correcto?

Existen varios tipos de prueba entre ellos las pruebas de unidad la cual enfoca dos tipos de prueba el de caja blanca y caja negra.

Pruebas de Caja Blanca

La prueba de caja blanca del software se comprueba los caminos lógicos del software proponiendo casos de prueba que ejerciten conjuntos específicos de condiciones y/o bucles. Se puede examinar el «estado del programa» en varios puntos para determinar si el estado real coincide con el esperado o mencionado.

Complejidad Ciclomática

La complejidad ciclomática es una métrica de software que proporciona una medida cuantitativa de la complejidad lógica de un programa.

La complejidad ciclomática se basa en la teoría gráfica y se calcula de una de tres maneras que son:

El número de regiones corresponde a la complejidad ciclomática.

La complejidad ciclomática $V(G)$, de una gráfica de flujo, G se define como:

$$V(G) = A - N + 2$$

Donde:

A = Número de aristas.

N = Número de nodos de la gráfica de flujo.

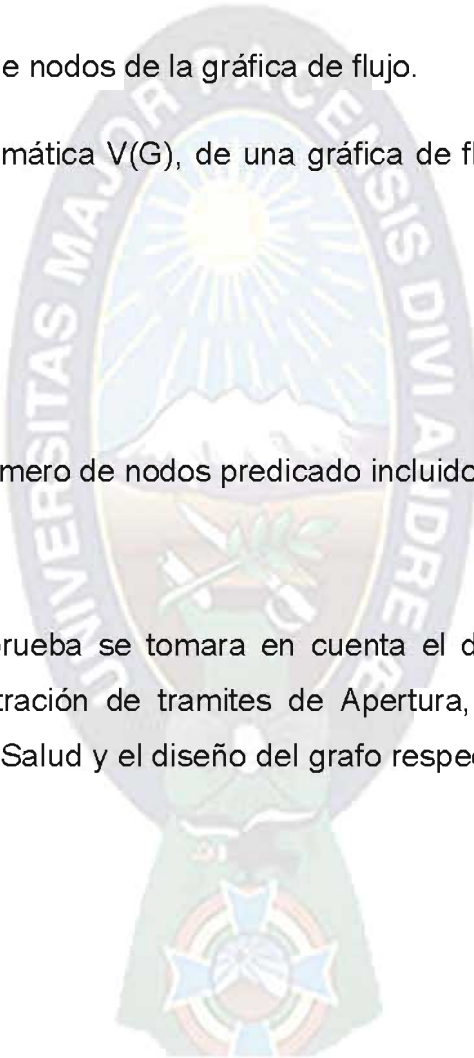
La complejidad ciclomática $V(G)$, de una gráfica de flujo G también se define como:

$$V(G) = NP + 1$$

Donde:

NP = Es el número de nodos predicado incluidos en la gráfica de flujo G

Para realizar esta prueba se tomara en cuenta el diseño procedimental del módulo de Administración de tramites de Apertura, Renovación y otros de establecimientos de Salud y el diseño del grafo respectivo es como muestra la siguiente figura.



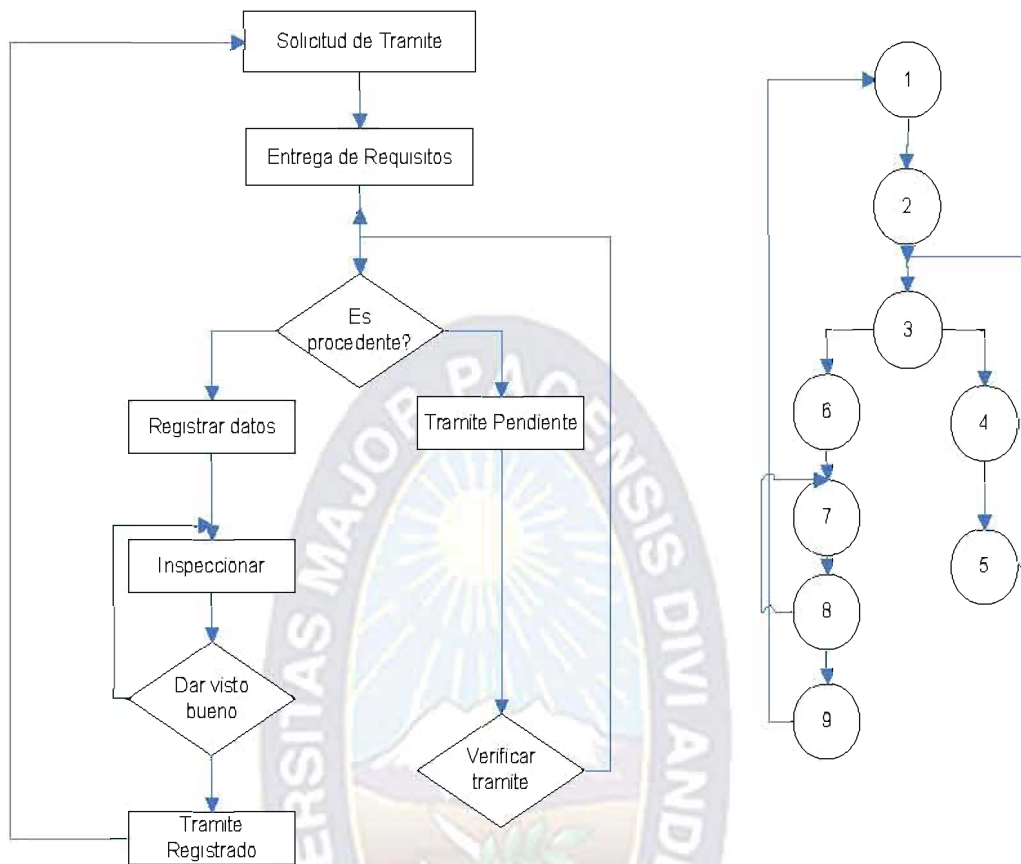


Figura 3.27: Grafo de Administración de trámites, renovación y traslado
Fuente: Elaboración Propia

De acuerdo al grafo se obtiene los siguientes caminos independientes

Camino1: 1-2-3-4-5-3-6-7-8-9

Camino2: 1-2-3-6-8-7-9

Camino3: 1-2-3-6-7-8-7-8-9

Ahora se halla la Complejidad Ciclomática

El grafo tiene cinco regiones

$V(G) = \# \text{ de Regiones} = 3$

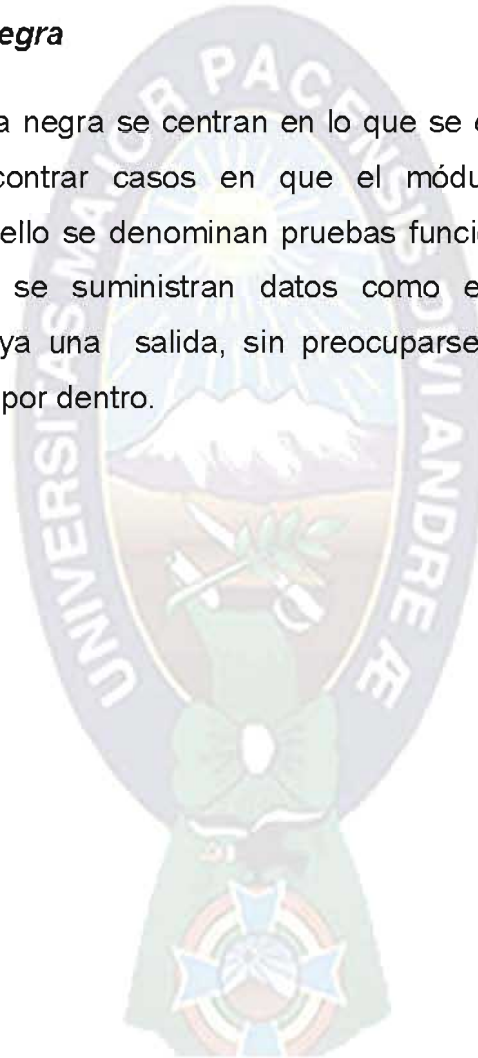
$$V(G) = A - N + 2 = 11 - 9 + 2 = 4$$

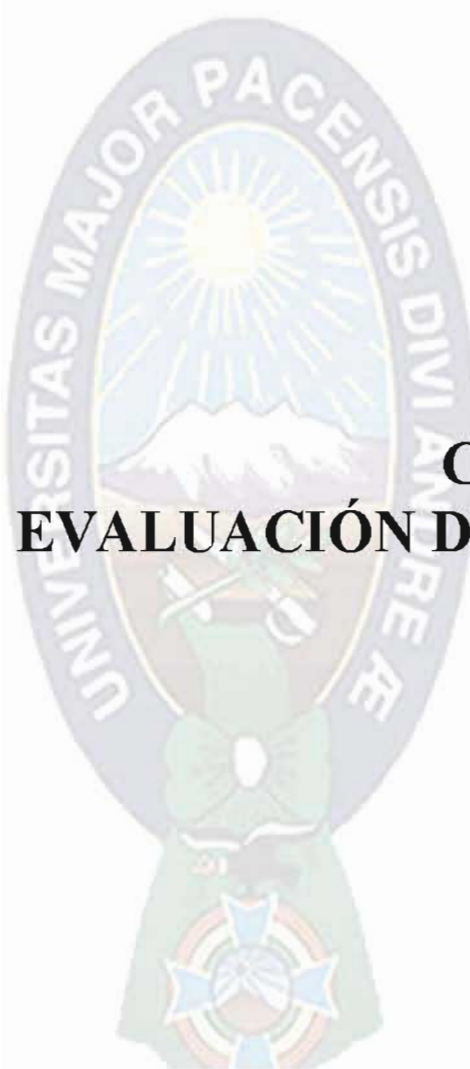
$$V(G) = NP + 1 = 3 + 1 = 4$$

Por tanto la complejidad ciclomática es 4, esto quiere decir que existen 4 caminos independientes para el proceso de índice de establecimientos.

Pruebas de Caja Negra

Las pruebas de caja negra se centran en lo que se espera de un módulo, es decir, intentan encontrar casos en que el módulo no se atiene a su especificación. Por ello se denominan pruebas funcionales, por lo tanto para probar el sistema se suministran datos como entrada y se estudia y comprueba que haya una salida, sin preocuparse de lo que pueda estar haciendo el módulo por dentro.





CAPÍTULO IV EVALUACIÓN DEL SISTEMA

4. Evaluación Del Sistema

4.1. CALIDAD DE SOFTWARE

Para medir la calidad de un producto de software se emplean modelos que especifican la calidad mediante definición de un conjunto de atributos o características que pueden ser medidos mediante métricas, las cuales analizaremos a continuación.

La calidad del software consiste en aquellos procedimientos, técnicas e instrumentos aplicados por entes capacitados para garantizar que un producto cumpla o supere un nivel mínimo aceptable para su comercialización, si es que así se lo ha planeado, lo que hasta el momento no se tiene estándares del software, no es lo mismo que las pruebas del sistema, estos son implícitos al momento de hacer las pruebas de integración final del sistema.

Por lo tanto para medir la calidad del sistema se tomara las medidas de calidad indirecta en el cual está incluida la Norma **ISO 9126** que facilitara y dará bases valiosas para determinar un sistema la misma contiene 6 atributos claves de calidad que son:

- Funcionalidad
- Confiabilidad
- Facilidad de mantenimiento
- Portabilidad

4.1.1. FUNCIONALIDAD

La funcionalidad es una característica muy importante dentro del desarrollo del software.

La funcionalidad de un sistema se mide tomando algunas características de dominio de información, donde se le asocia un valor de complejidad. Para realizar la medición de los puntos de función se debe determinar características como: número de entradas del usuario, número de salidas del usuario, número de peticiones del usuario, número de archivos y número de interfaces externas.

De acuerdo a los anteriores parámetros se tiene la siguiente tabla de valores:

VALOR DEL DOMINIO DE PONDERACIÓN	CONTEO	PONDERACIÓN	TOTAL
Entradas externas	10	3	30
Salidas externas	20	4	80
Consultas externas	7	3	21
Archivos de lógica interna	20	7	140
Archivos de lógica externa	5	5	25
Total			296

Tabla 4.1.1 a): Cálculo de Cuenta Total de la métrica de Punto Función
Fuente: Elaboración Propia

Preguntas	Factor de ajuste
El sistema requiere respaldo y recuperación confiables	5
Se requieren comunicaciones de datos especializadas para transferir información a la aplicación, u obtener de ella.	5
Hay funciones distribuidas de procesamiento	4
El desempeño es crítico	3
El sistema se ejecutará en un entorno existente que tiene un uso pesado de operaciones.	1
El sistema requiere entrada de datos en línea.	5
La entrada de datos en línea requiere de la transacción de entrada se construya en varias pantallas u operaciones	4
Los archivos lógicos internos se actualizan en línea.	4
Las entradas, las salidas, los archivos o las consultas son complejos	3
Es complejo el procesamiento interno.	3
La aplicación está diseñada para aplicar el cambio y para que el usuario lo use fácilmente	5
TOTAL	42

Tabla 4.1.1 b):: Cálculo de Ajuste de complejidad
Fuente: Elaboración Propia

Determinar el valor de complejidad es subjetivo, para calcular los puntos de función se realiza la siguiente operación.

$$PF = \text{Cuenta total} * (0.65 + 0.01 * \sum Fi)$$

Donde:

Cuenta total: Es la suma de los valores obtenidos en el cálculo de métricas de punto función obtenidas.

Fi : Son valores de ajuste de complejidad dadas según las respuestas obtenidas.

Remplazando valores obtenidos se tiene el siguiente valor:

$$PF = 296 * (0.65 + 0.01 * 42)$$

$$PF = 316.72$$

4.1.2. CONFIABILIDAD

Es la cantidad de tiempo en que el software esta disponible para usarse según los siguientes atributos: madurez, tolerancia a fallas y facilidad de recuperación.

La confiabilidad del sistema esta dado por la siguiente relación:

$$R(t) = R1(t) * R2(t) * R3(t) \dots Rn(t)$$

Donde R(t) esta dado por:

$$R(t) = e^{-HP(t)}$$

Donde:

R(t) = Es la confiabilidad del subsistema en un tiempo t

P(t) = Es la probabilidad de falla

λ = Es la tasa constante de fallo

Después de realizar las pruebas necesarias al sistema se obtuvieron los siguientes resultados:

R1 = Registro y control tramites = $R(t) = e^{-0.8*0.2}=0.85$

R2= Registro y control de establecimientos de salud = $R(t) = e^{-0.9*0.1}=0.91$

R3 = Registro y control de propietarios = $R(t) = e^{-0.8*0.2}=0.85$

R4 = Registro y control de empleados = $R(t) = e^{-0.9*0.1}=0.91$

R5 = Registro y control de inspecciones = $R(t) = e^{-0.9*0.1}=0.91$

R6 = Registro y control de citaciones = $R(t) = e^{-0.9*0.1}=0.91$

R7 = Registro y control de orden de pago = $R(t) = e^{-0.8*0.2}=0.85$

R8 = Registro y control de requisitos = $R(t) = e^{-0.8*0.2}=0.85$

La confiabilidad en paralelo se obtiene de la siguiente forma:

$R9 = 1 - (1 - R2) * (1 - R1) = 1 - (1 - 0.91) * (1 - 0.85) = 0.98$

$R10 = 1 - (1 - R9) * (1 - R4) = 1 - (1 - 0.98) * (1 - 0.85) = 0.99$

$R11 = 1 - (1 - R5) * (1 - R6) = 1 - (1 - 0.91) * (1 - 0.85) = 0.98$

$R12 = 1 - (1 - R8) * (1 - R7) = 1 - (1 - 0.91) * (1 - 0.91) = 0.99$

La confiabilidad en paralelo es

$R13 = R9 * R7 = 0.98 * 0.91 = 0.89$

$R14 = R4 * R11 = 0.95 * 0.98 = 0.93$

De acuerdo a la ecuación planteada para medir la confiabilidad se tiene:

$R = 1 - (1 - R9) * (1 - R10) * (1 - R11) * (1 - R12) * (1 - R13) * (1 - R14)$

$R = 1 - (1 - 0.98) * (1 - 0.99) * (1 - 0.98) * (1 - 0.99) * (1 - 0.89) * (1 - 0.93)$

$R = 0.99$

De acuerdo al resultado anterior se deduce que el sistema tiene un 99% de confiabilidad.

4.1.3. FACILIDAD DE MANTENIMIENTO

Para aplicar la métrica de mantenimiento se utilizará el Estándar IEEE 982 que nos da un índice de madurez del software que proporciona un indicador de la estabilidad de un producto software.

Esta métrica está dada por los siguientes parámetros:

Mt = Número de módulo de la versión actual.

Fi = Número de módulos en la versión actual que han cambiado

Fa = Número de módulos en la versión actual que se han añadido

Fd = Número de módulos en la versión anterior que se han borrado en la versión actual.

El índice de madurez de software (IMS) se calcula mediante la siguiente fórmula:

$$IMS = [Mt - (Fa + Fi + Fd)]/Mt$$

86

Realizando las pruebas necesarias se obtuvieron los siguientes valores

Mt = 20

Fi = 1

Fa = 1

Fd = 0

Remplazando valores en la fórmula anterior se tiene:

$$IMS = [20 - (1 + 1 + 0)]/20$$

IMS = 0.9

De acuerdo al valor obtenido se deduce que el Índice de Madurez del Proyecto es del 90 %.

4.1.4. PORTABILIDAD

La portabilidad permite transportar datos de un sistema a otro, en el caso del presente sistema es portable en un 99% si las características del hardware y software son las siguientes:.

Servidor

Hardware:

- Computadora Pentium 4 – 2GHZ
 - Disco Duro de 20 Gbyte

- Micro Procesador Mayor o igual 1 Ghz
- Memoria RAM 512 Mb
- Monitor super VGA
- Tarjeta de Video mayor o igual 64Mb

Software:

- Windows 2000
- Lenguaje de programación PHP
- Gestor de Base de Datos MySql
- Servidor de páginas Web Xampp

Cliente

Hardware

- Computadora Pentium 4 – 600MHz
 - Disco Duro de 20 Gbyte
 - Micro Procesador Mayor o igual 1 Ghz
 - Memoria RAM 128 Mb
 - Monitor super VGA
 - Tarjeta de Video mayor o igual 64Mb

Software:

- Lenguaje de programación PHP
- Gestor de Base de Datos MySql
- Servidor de páginas Web Xampp
- Firefox o Safari
- Herramienta diseño Case Architec
- Fireworks 8
- Dreamweaver 8



CAPÍTULO V

CONCLUSIONES Y

RECOMENDACIONES

5. CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

Con la implementación del Sistema se logró cumplir con el objetivo principal que es: "Desarrollar e implementar un sistema de información para el control y seguimiento de la Unidad de Acreditación y Certificación, para la prestación de servicios de manera oportuna y confiable".

También se contribuyó a cumplir con los aspectos del marco reglamentario, código de salud, y otros reglamentos, mediante las inspecciones que se realiza a los diferentes establecimientos garantizando la confiabilidad en el aspecto de salud a la población que recurre a las entidades que son legalmente acreditadas y certificadas.

5.2 RECOMENDACIONES

Sería conveniente desarrollar:

Un Sistema Georeferencial para establecimientos de salud.

Un Sistema Experto de Pronóstico de Dotaciones basado en indicadores frecuentes y poblacionales para el área de farmacia y laboratorio.

BIBLIOGRAFÍA

[**GS12000**] Grupo de sistemas de información avanzados: El Análisis del problema, las labores del análisis, Ingeniería de software I, Universidad de Zagarra(2000).

[**BOC1998**] Booch G. 1991: Análisis y Diseño Orientado a Objetos, Segunda Edición, Pag.857, Editorial Addison- Wesley, Dias de Santo, México.

[**JBR2000**] Jacobson I. Booch G. Rambaugt J. 2000: El Proceso Unificado de Desarrollote software, Primera Edición.

[**KEN1999**] KENDALL J. 1999: Análisis y Diseño de Sistemas, tercera edición, Editorial Prentice may, Mexico.

[**PRE 2002**] Presman R. 2002: Ingeniería de Software Quinta Edición.,MCGRAW- ILL, Mexico.

