

**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO**

CARRERA: INGENIERÍA DE SISTEMAS

Tesis previa a la obtención del título de: INGENIERO DE SISTEMAS

TEMA:

**ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DE UNA APLICACIÓN WEB QUE
PERMITA AUTOMATIZAR LOS PROCESOS ADMINISTRATIVOS DEL CENTRO
PSICOLÓGICO DEL CAMPUS GIRÓN DE LA UNIVERSIDAD POLITÉCNICA
SALESIANA.**

AUTOR:

VÍCTOR HERNÁN QUIMBIAMBA LANCHIMBA

DIRECTORA:

PATSY MALENA PRIETO VÉLEZ

Quito, septiembre 2013

DECLARATORIA DE RESPONSABILIDAD Y AUTORIZACIÓN DE USO DEL
TRABAJO DE GRADO

Yo Víctor Hernán Quimbiamba Lanchimba autorizo a la Universidad Politécnica Salesiana la publicación total o parcial de este trabajo de grado y su reproducción sin fines de lucro.

Además declaro que los conceptos y análisis desarrollados y las conclusiones del presente trabajo son de exclusiva responsabilidad del autor.

Víctor Hernán Quimbiamba Lanchimba

CC 1718596412

DEDICATORIA

Este trabajo está dedicado a mis padres, gracias a su constante apoyo e inmensa paciencia, día a día son un ejemplo y me han brindado la fuerza para seguir siempre adelante.

ÍNDICE

INTRODUCCIÓN.....	1
CAPÍTULO 1.....	7
1. Marco teórico.....	7
1.1 JSF 2.0.....	7
1.1.1 Definiciones.....	8
1.1.2 Ciclo de vida de JSF.....	9
1.1.3 Modelo de navegación JSF.....	10
1.1.4 Managed beans.....	12
1.1.4.1 Alcance de Managed beans.....	13
1.1.5 Ventajas de jsf.....	14
1.2 EJB 3.0.....	14
1.2.1 Tipos de beans.....	15
1.2.1.1 Entity beans.....	15
1.2.1.2 Session beans.....	16
1.2.1.3 Message-driver beans.....	16
1.2.2 Anotaciones ejb.....	17
1.3 Primefaces.....	18
1.3.1 Comparativa.....	19
1.4 Oracle database.....	19
1.4.1 Historia.....	20
1.4.2 ¿Por qué usar Oracle database?.....	20
1.4.3 Ventajas de usar Oracle database.....	22
1.5 Glassfish.....	21
1.5.1 Diseño modular.....	21
1.5.2 Actualización e integración de la herramienta.....	21
1.6 Eclipse JPA.....	22
1.6.1 Mapeo de objetos.....	22
CAPÍTULO 2.....	24
2. Metodología rup.....	24

2.1 Características.....	24
2.2 Definición fases.....	24
2.2.1 Inicio.....	24
2.2.2 Elaboración.....	25
2.2.3 Construcción.....	25
2.2.4 Transición.....	25
2.3 Fases.....	26
2.3.1 Casos de uso.....	26
2.3.1.1 Préstamos de materiales.....	26
2.3.1.2 Registro paciente.....	28
2.3.1.3 Registro atención.....	30
2.3.1.4 Administración de materiales.....	32
2.3.1.5 Administración de usuarios.....	34
2.3.1.6 Administración de tipo de material.....	36
2.3.1.7 Administración de servicio.....	38
2.3.1.8 Administración de horarios.....	40
2.3.1.9 Administración de período.....	42
2.3.2 Modelo de análisis.....	43
2.3.2.1 Préstamo de materiales.....	43
2.3.2.2 Registro paciente.....	44
2.3.2.3 Registro de atención.....	45
2.3.2.4 Administración de materiales.....	46
2.3.2.5 Administración de usuarios.....	47
2.3.2.6 Administración de tipo material.....	48
2.3.2.7 Administración de servicios.....	49
2.3.2.8 Administración de horario.....	50
2.3.2.9 Administración de periodo.....	51
2.3.3 Diagrama de secuencia.....	52
2.3.3.1 Préstamo de materiales.....	52
2.3.3.2 Registro paciente.....	53
2.3.3.3 Registro atención.....	54

2.3.3.4 Administración de material.....	55
2.3.3.5 Administración de usuarios.....	56
2.3.3.6 Administración de tipo material.....	57
2.3.3.7 Administración de servicio.....	58
2.3.3.8 Administración de horarios.....	59
2.3.3.9 Administración de periodo.....	60
2.3.4 Diagrama de clases.....	61
2.3.5 Diagrama de componentes.....	62
CAPÍTULO 3.....	63
3. Diseño e implementación.....	63
3.1 Diseño de interfaces.....	63
3.2 Diseño de base de datos.....	69
3.2.1 Modelo lógico.....	70
3.2.2 Modelo físico.....	71
3.2.3 Diccionario de datos.....	72
3.3 Implementación.....	84
3.3.1 Manual de instalación.....	84
3.3.2 Manual de usuario.....	94
CAPÍTULO 4.....	113
4. Pruebas.....	113
4.1 Pruebas funcionales.....	113
4.2 Pruebas no funcionales.....	117
Conclusiones.....	121
Recomendaciones.....	122
Lista de referencias.....	123
Anexos.....	124

ÍNDICE DE FIGURAS

Figura 1 – Aplicaciones de varios niveles.....	8
Figura 2 – Ciclo de vida de JSF.....	10
Figura 3 – Estructura de regla de navegación.....	11
Figura 4 – Regla de navegación directa.....	12
Figura 5 – Registro de un Managed bean con anotación.....	12
Figura 6 – Registro de un Managed bean en facesconfig.xml.....	12
Figura 7 – Registro del alcance a un Managed bean con anotación.....	13
Figura 8 – Registro de EJB.....	15
Figura 9 – Declaración de un Bean de entidad.....	16
Figura 10 – Contenedor EJB.....	17
Figura 11 – Entidad JPA.....	23
Figura 12 – Caso de uso: Préstamo de materiales.....	26
Figura 13 – Caso de uso: Registro paciente.....	28
Figura 14 – Caso de uso: Registro atención.....	30
Figura 15 – Caso de uso: Administración de materiales.....	32
Figura 16 – Caso de uso: Administración de usuarios.....	34
Figura 17 – Caso de uso: Administración de tipo material.....	36
Figura 18 – Caso de uso: Administración de servicio.....	38
Figura 19 – Caso de uso: Administración de horarios.....	40
Figura 20 – Caso de uso: Administración de periodo.....	42
Figura 21 – Modelo de análisis: préstamos de materiales.....	43
Figura 22 – Modelo de análisis: registro paciente.....	44
Figura 23 – Modelo de análisis: registro de atención.....	45
Figura 24 – Modelo de análisis: administración de materiales.....	46
Figura 25 – Modelo de análisis: administración de usuarios.....	47
Figura 26 – Modelo de análisis: administración de tipo material.....	48

Figura 27 – Modelo de análisis: administración de servicio.....	49
Figura 28 – Modelo de análisis: administración de horarios.....	50
Figura 29 – Modelo de análisis: administración de periodos.....	51
Figura 30 – Diagrama de secuencia: préstamo de materiales.....	52
Figura 31 – Diagrama de secuencia: registro paciente.....	53
Figura 32 – Diagrama de secuencia: registro atención.....	54
Figura 33 – Diagrama de secuencia: administración de material.....	55
Figura 34 – Diagrama de secuencia: administración de usuarios.....	56
Figura 35 – Diagrama de secuencia: administración tipo material.....	57
Figura 36 – Diagrama de secuencia: administración de servicio.....	58
Figura 37 – Diagrama de secuencia: administración de horarios.....	59
Figura 38 – Diagrama de secuencia: administración de periodos.....	60
Figura 39 – Diagrama de clase.....	61
Figura 40 – Diagrama de componentes.....	62
Figura 41 – Pantalla inicio sesión.....	63
Figura 42 – Plantilla principal.....	64
Figura 43 – Área de usuario.....	65
Figura 44 – Área de cabecera.....	65
Figura 45 – Área del menú principal.....	65
Figura 46 – Área de las opciones de administración.....	66
Figura 47 – Área de pie de página.....	66
Figura 48 – Lista de administración.....	67
Figura 49 – Pantalla de procesos.....	68
Figura 50 – Icono de error.....	68
Figura 51 – Mensaje de proceso exitoso.....	69
Figura 52 – Mensaje de proceso fallido.....	69
Figura 53 – Modelo lógico.....	70

Figura 54 – Modelo físico.	71
Figura 55 – Directorio del Glassfish.....	85
Figura 56 – Dirección de librería para el dominio 1.....	86
Figura 57 – Comando para iniciar un dominio de Glassfish.....	86
Figura 58 – Consola de administración de Glassfish.....	87
Figura 59 – Creación de JDBC Connection pools 1.....	88
Figura 60 – Creación de JDBC Connection pools 2.....	88
Figura 61 – Propiedades de conexión.....	89
Figura 62 – Tabla de pools de conexiones 1.....	89
Figura 63 – Tabla de pools de conexiones 2.....	90
Figura 64 – Tabla de pools de conexiones 3.....	90
Figura 65 – Creación de recurso jdbc 1.....	91
Figura 66 – Creación de recurso jdbc 2.....	92
Figura 67 – Despliegue de aplicaciones o módulos – selección de archivo.....	92
Figura 68 – Lista de aplicación o módulos desplegados.....	93
Figura 69 – Pantalla de presentación de la aplicación.....	94
Figura 70 – Pantalla de ingreso a la aplicación.....	95
Figura 71 – Pantalla principal del perfil administrativo.....	97
Figura 72 – Lista de administración de usuarios.....	98
Figura 73 – Formulario de creación y edición de usuarios.....	99
Figura 74 – Lista de tipos de materiales.....	100
Figura 75 – Formulario de creación y edición de tipo material.....	101
Figura 76 – Formulario de creación y edición de paciente.....	102
Figura 77 – Lista de pacientes.....	103
Figura 78 – Creación de cita médica.....	103
Figura 79 – Lista de citas.....	104
Figura 80 – Formulario de administración de préstamos.....	105

Figura 81 – Formulario de administración de devolución.....	106
Figura 82 – Lista médicos.....	106
Figura 83 – Lista de configuración de horarios.....	107
Figura 84 – Formulario de creación y edición de horarios.....	107
Figura 85 – Lista de periodos.....	108
Figura 86 – Ventana emergente de creación de nuevo periodo.....	108
Figura 87 – Mensaje de confirmación para la creación de un nuevo periodo.....	109
Figura 88 – Formulario de generación de reportes.....	109
Figura 89 – Pantalla principal del perfil médico.....	110
Figura 90 – Pantalla principal del perfil médico.....	111
Figura 91 – Ficha de paciente.....	112
Figura 92 – Resultados de las pruebas al inicio de sesión.....	118
Figura 93 – Resultados de las pruebas a la creación de usuario.....	119
Figura 94 – Resultados de las pruebas al registro de cita médica.....	119
Figura 95 – Resultados de las pruebas al registro de paciente.....	120
Figura 96 – Selección de idioma de instalación.....	124
Figura 97 – Selección de opción de instalación.....	124
Figura 98 – Descarga de componentes adicionales.....	125
Figura 99 – Configuración de red.....	125
Figura 100 – Particionado de discos.....	125
Figura 101 – Progreso de instalación del sistema base.....	126
Figura 102 – Configuración de usuario y contraseña - nombre real.....	126
Figura 103 – Configuración del gestor de paquetes.....	127
Figura 104 – Selección de programas.....	127
Figura 105 – Descarga de componentes adicionales.....	128
Figura 106 – Pantalla de bienvenida de instalación JRE.....	129
Figura 107 – Pantalla de bienvenida de instalación JRE.....	130

Figura 108 – Pantalla de instalación nuevo componente en Firefox.....	131
Figura 109 – Barra de herramientas con ícono Selenium IDE.....	132
Figura 110 – Ventana de Selenium IDE.....	132
Figura 111 – Barra de herramientas Firebug en explorador.....	133

ÍNDICE DE TABLAS

Tabla 1 – Comparativa primefaces con otros frameworks basados en JSF.....	19
Tabla 2 – Especificación del caso de uso: Préstamo de materiales.....	27
Tabla 3 - Especificación del caso de uso: Registro paciente.....	29
Tabla 4 - Especificación del caso de uso: Registro atención.....	31
Tabla 5 - Especificación del caso de uso: Administración de materiales.....	33
Tabla 6 - Especificación del caso de uso: Administración de usuarios.....	35
Tabla 7 – Especificación del caso de uso: Administración de tipo material.....	37
Tabla 8 – Especificación del caso de uso: Administración de servicio.....	39
Tabla 9 – Especificación del caso de uso: Administración de horarios.....	41
Tabla 10 – Especificación del caso de uso: Administración de periodos.....	42
Tabla 11 – Tabla tipos_materiales.....	72
Tabla 12 – Tabla materiales.....	73
Tabla 13 – Tabla prestamos.....	74
Tabla 14 – Tabla periodos.....	75
Tabla 15 – Tabla usuarios.....	76
Tabla 16 – Tabla especialidades.....	77
Tabla 17 – Tabla horarios.....	78
Tabla 18 – Tabla tipos_servicios.....	79
Tabla 19 – Tabla pacientes.....	80
Tabla 20 – Tabla citas_medicas.....	81
Tabla 21 – Tabla ficha.....	82
Tabla 22 – Especificación ingreso a la aplicación.....	113
Tabla 23 – Especificación de registro de pacientes.....	114
Tabla 24 – Especificación de registro de cita médica.....	115
Tabla 25 – Especificación de préstamos y devoluciones.....	116
Tabla 26 – Especificación de registro de pacientes.....	117

Tabla 27 – Resultados de pruebas inicio de sesión.....	118
Tabla 28 – Resultados de pruebas creación de usuario.	119
Tabla 29 – Resultados de las pruebas al registro de cita médica.....	120
Tabla 30 – Resultados de las pruebas al registro de paciente.....	120

ÍNDICE DE ANEXOS

Anexo 1 – Instalación Ubuntu Server.....	124
Anexo 2 – Instalación de la máquina virtual de Java.....	139
Anexo 3 – Selenium.....	131
Anexo 4 – Firebug.....	133

AGRADECIMIENTO

Agradezco mucho a la universidad politécnica salesiana por el apoyo que brinda a todos los estudiantes, a los docentes que tuve el agrado de conocer durante toda mi carrera universitaria y en especial a mi tutora de tesis por la paciencia y la sabiduría para guiar este trabajo hasta el final.

RESUMEN

Esta tesis se desarrolló pensando en solucionar los problemas administrativos que posee el centro psicológico del campus girón de la Universidad Politécnica Salesiana, específicamente el problema asignación y atención de citas médicas, la administración de recursos y la gestión de préstamos de materiales. Para ello se ha analizado, diseñado y desarrollado una aplicación Web que permita automatizar estos procesos, es decir, tener una mayor organización, disminuir los tiempos en los procesos, reducir recursos materiales y espacio físico.

El uso de la metodología “RUP” hizo que el trabajo se lleve de una manera organizada, esto significó una ayuda muy útil pues el proceso se encaminaba muy bien y se sabía en qué punto se encontraba el trabajo, y que parte del trabajo faltaba. Principalmente el uso de los casos de uso pues estos son fragmentos del sistema y dividir la funcionalidad y analizarla por separada, esto lleva a un mejor entendimiento.

El desarrollo de la aplicación web luego de haber tenido un buen proceso de análisis fue muy claro puesto que un buen análisis conlleva a minimizar los cambios al momento de desarrollar una aplicación. Esta aplicación fue desarrollada bajo la última versión de la plataforma de programación Java EE, esto implica la separación de tres o más capas en el desarrollo y la inclusión de varias especificaciones de Java EE, el uso de esta plataforma incrementa la confiabilidad del desarrollo pues este ha tenido varios refinamientos y desde su creación ha estado respaldada por grandes empresas de tecnología.

El desarrollo de la documentación y organización de la información resulta ser una parte fundamental y muy complicada si no desde el inicio no se la lleva con orden, con el uso correcto de la metodología y el acompañamiento de un líder los proyectos siempre salen adelante.

ABSTRACT

This thesis was developed with resolving administrative problems that owns the campus psychological center girón Salesian Polytechnic University, specifically the allocation problem of medical appointments and care, resource management and materials management loans. To this extent, designed and developed a Web application that automates these processes, ie having a more organized, reduce process times, reduce resources and physical space.

The use of the methodology "RUP" made the work is carried in an organized manner, this meant a very useful aid for the process was moving very well and knew what was about work, and that part was missing work. Mainly using the use cases for these are fragments and dividing system and analyzed by separate functionality, this leads to better understanding.

The web application development after having had a good analysis process was very clear since a good analysis leads to minimize changes when developing an application. This application was developed under the latest version of the Java EE programming platform, this involves the separation of three or more layers in the development and inclusion of several Java EE specifications, the use of this platform development increases reliability because this has had several refinements and since its inception has been supported by major technology companies.

The development of the documentation and organization of information turns out to be a fundamental and very difficult if not from the beginning not carrying it with order, with the correct use of the methodology and the accompaniment of a project leader always go forward.

INTRODUCCIÓN

El Centro Psicológico Salesiano, “DR. P. Emilio Gambirasio” del Campus Girón de la Universidad Politécnica Salesiana es un centro de ayuda y capacitación que brinda sus servicios profesionales al público que lo requiera. El Centro Psicológico Salesiano, tiene determinado sus procesos de atención a sus pacientes y de administración de recursos.

La administración de recursos materiales y organización de espacios físicos se lo realiza de manera manual, esto genera inconvenientes en la organización y en el manejo de los recursos. Al no contar con una herramienta eficaz para el manejo de sus procesos, los mismos se ven afectados con retardos de tiempo, confusión, incomodidad y en casos hasta pérdida de datos.

La búsqueda de la información tiene un costo de tiempo demasiado alto y provoca incomodidades tanto para los pacientes como para el personal del centro psicológico. Llevar registros físicos de la información es también una pérdida de espacio que podría ser utilizado para otras actividades.

Con respecto a las Fichas Clínicas el almacenamiento de estas se hace mediante la utilización de carpetas, separadores, hojas de papel bond y esto representa un gasto económico y al ser un material deteriorable representa un riesgo a cualquier eventualidad en donde los documentos físicos se vean afectados. Existe además desorden en los horarios y utilización de espacios que el personal puede ocupar, pues los horarios muchas de las veces no se respetan o al realizarlos de forma manual se comete errores pues no existe ningún tipo de validación para la asignación de espacios y sus horarios de uso.

El seguimiento económico es bastante básico pero al no tener una herramienta que facilite la organización de las finanzas esto llega a ser a veces una tarea demasiada complicada y laboriosa.

La generación de reportes es una tarea que se realiza de forma manual, el caso económico es mediamente controlable pero la búsqueda de fichas clínicas se convierte en una tarea casi imposible pues, si se desea tener datos exactos se tendría que revisar cada una de las fichas registradas.

Objetivos

a. Objetivo general

Desarrollar una aplicación WEB que permita la administración de Fichas Clínicas y recursos del Centro Psicológico Salesiano, “DR. P. Emilio Gambirasio” del Campus Girón de la Universidad Politécnica Salesiana.

b. Objetivo específico

- i. Realizar los procesos de recolección de requerimientos y obtención de la información que sea necesaria para la realización de la solución informática orientada al centro psicológico.
- ii. Optimizar los procesos de administración de recursos y control de Fichas Clínicas del Centro Psicológico.
- iii. Utilizar la metodología RUP para obtener una aplicación WEB cumpliendo con los requerimientos establecidos por el centro psicológico.
- iv. Desarrollar e Implementar una Aplicación Web para la automatización de procesos del Centro Psicológico usando tecnología Java Web
- v. Realizar pruebas funcionales de unidad e integración.
- vi. Realizar pruebas funcionales de aceptación.
- vii. Capacitar al usuario para el uso correcto de la aplicación

Justificación

En la actualidad y con el cambio de las tecnologías es muy común que las organizaciones, instituciones o pequeñas empresas quieran utilizar nuevas herramientas tecnológicas en sus ámbitos de trabajo, pues la realización de tareas automatizadas permite optimizar tiempos y asegurar la información, a la vez que ayuda con el ahorro de recursos tanto de espacios como de personal.

La aplicación WEB será un instrumento de ayuda en el Centro Psicológico Salesiano, “DR. P. Emilio Gambirasio” para mejorar los procesos de administración de recursos y control de fichas clínicas, pues así se ahorrará tiempo y recursos, también se garantiza que la información este segura, ahorrando espacio físico. Un registro digital de la información sirve de ayuda cuando se requiere recuperar la información.

Alcance

La aplicación WEB será instalada en el departamento de informática de la universidad Politécnica Salesiana Campus Girón, puesto que la institución cuenta con los recursos necesarios para su instalación.

El uso de la aplicación será a través de niveles de acceso definidos y por petición de un usuario y una clave, la misma que será encriptado para proporcionar un mejor nivel de seguridad.

La aplicación será desarrollada en Java como lenguaje de programación y Oracle DB como base de datos por requerimiento de la institución, además estas son las herramientas que lideran el mercado tecnológico del desarrollo de software.

Módulo I

Registro de citas médicas, este es el módulo de inicio para el control de fichas clínicas, pues todo comienza con la solicitud de una cita clínica, eso se realiza dependiendo del motivo, la edad del paciente, conjuntamente con la disponibilidad

de los espacios y la disponibilidad de los psicólogos con sus respectivos horarios. El funcionamiento en esta parte sería que se seleccione un médico, un horario, y se ingrese un paciente, la duplicidad de horarios-médicos será controlada a nivel de consultas, es decir no se correrá el riesgo de que existan más de una cita médica asignada al mismo Psicólogo en el mismo horario.

Cabe mencionar que el proceso es casi el mismo para un usuario nuevo o para un usuario registrado, con la única diferencia que si el usuario está registrado se le buscará según sus nombres, también se registrará el costo de la consulta, este valor se registra hasta después de que el paciente es atendido, es decir la cita médica no se cierra hasta que el usuario cancela el costo de la atención y el número de documento con el cual se registra el pago.

Módulo II.

Representa uno de los módulos más importantes de la aplicación pues es la gestión de fichas clínicas, este proceso se realiza a través de una cita médica, una vez asignada la cita médica, el paciente asiste el día y en el horario indicado para posteriormente ser atendido, existen dos casos de registros:

- La primera es cuando el paciente es nuevo, en este caso se crea una ficha clínica dependiendo del caso y la edad del paciente.
- La segunda es cuando el paciente ya está registrado, en este caso se añade nueva información por cita, y se actualiza la información del paciente de ser necesario. Los formularios para el registro y control de fichas clínicas ya están definidos y clasificados en dos grupos, una para niños y adolescentes y otra para adultos o mayores de edad.

Módulo III

Se encuentra definido por

1. Inventarios de materiales.
2. Inventarios de equipos.

- a. Para el inventario de materiales. Se registran las materias o pruebas psicológicas, el formulario de datos está definido, la presentación de esta información será en forma de catálogo. El uso de este módulo será de visualización en forma de catálogo, préstamos, devoluciones e historial codificado de préstamos y devoluciones.
- b. Para el inventario de equipos. Se registran acorde a la información de los formularios que ya están definidos, además del registro los equipos pueden darse de baja, no se asignan ni están en otro lugar que no sea el laboratorio de psicología.

Módulo IV

Está dedicado a la administración de:

1. Usuarios: permite crear, editar y deshabilitar usuarios, existen dos tipos de usuarios los cuales son: administradores y psicólogos, cada usuario tiene su rol, y cada rol tiene sus respectivos permisos.
2. Horarios: permite crear, editar y deshabilitar los horarios asignados a cada usuario por espacio de trabajo y por día, este horario se establece una vez cada semestre y es el mismo durante ese lapso de tiempo a menos que se lo modifique.
3. Parámetros: permite crear, editar y deshabilitar los parámetros del centro, estos parámetros serán definidos por la institución acorde a la necesidad de su utilización.

Módulo V

Está dedicado a la generación de reportes los cuales están definidos por el centro psicológico en base a su tiempo y experiencia, se clasifican por fichas clínicas y por Inventario:

1. En la parte de fichas clínicas, se generan informes parametrizables según los datos especificados en las fichas: mes, año, tipo de consulta. Así también por

el tipo de paciente: adultos y niños. También se generará un informe mensual de los costos acumulados por Citas Médicas cobradas, está a su vez se clasificará de acuerdo al tipo de consulta.

2. En la parte de reportes por inventario estos se imprimen a manera de catálogo, siempre son el mismo formato sin parámetros, es decir es solo una lista de todos los materiales y equipos.

Descripción del desarrollo

La aplicación se desarrollará usando el estándar de Java Enterprise Application, esto incluye tecnologías como como JPA 2.0, EJB 3.0, utilizando la librería de componentes JSF PrimeFaces y GlassFish como servidor de aplicaciones. Además la instrucción hace uso de una de los mejores gestores de bases de datos, en este caso Oracle Database.

CAPÍTULO 1

MARCO TEÓRICO

1.1 JFS 2.0

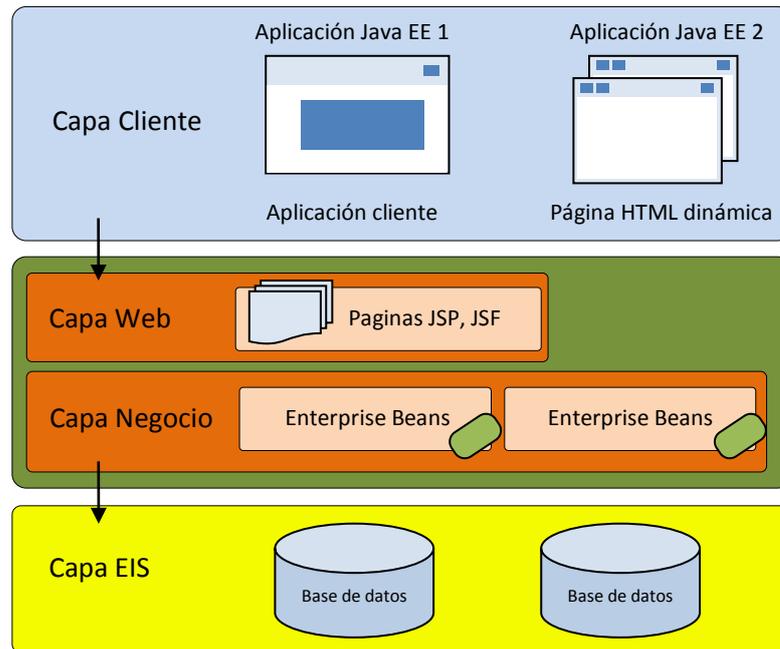
Java Platform, Enterprise Edition (Java EE), es una colección de especificaciones que definen una infraestructura para desarrollar aplicaciones distribuidas multicapa, el objetivo de la plataforma Java EE es proporcionar a los desarrolladores un fuerte conjunto de Interfaz de programación de aplicaciones (IPA), mejorando el rendimiento y reduciendo la complejidad de las aplicaciones.

Java EE simplifica el desarrollo de aplicaciones y reduce el trabajo para programadores al crear componentes modulares normalizados y reutilizables, así como al permitir controlar muchos aspectos de la programación automáticamente por nivel.

El modelo de aplicaciones Java EE define una arquitectura para la implementación de servicios como aplicaciones de varios niveles que ofrecen la escalabilidad, accesibilidad y facilidad de gestión que necesitan las aplicaciones de nivel empresarial. El desarrollador puede confiar en la plataforma para ofrecer soluciones a los diferentes problemas a nivel de sistemas de desarrollo de un servicio de varios niveles.

Aunque una aplicación Java EE puede constar de los tres o cuatro niveles mostrados en la Figura 1, las aplicaciones Java EE de niveles múltiples se consideran generalmente ser de tres niveles debido a que están distribuidas en tres lugares: Máquinas cliente, la máquina de servidor Java EE, y la base de datos o máquinas existentes en el back-end.

Figura 1. Aplicaciones de varios niveles



Fuente: The Java EE 6 Tutorial

La especificación Java EE define los siguientes componentes:

- Los clientes de aplicaciones y applets son componentes que se ejecutan en el cliente.
- Java Servlets, JavaServer Faces (JSF) y JavaServer Pages (JSP) componentes de la tecnología web que se ejecutan en el servidor.
- Enterprise JavaBeans (EJB) son componentes de negocio que se ejecutan en el servidor.

1.1.1 Definiciones

“JSF es un marco estándar de Java para la creación de interfaces de usuario en aplicaciones Web” (Ed Burns, Chris Schalk, with Neil Griffin – 2009)

En mayo del 2001 varios proveedores de software libre incluidos Sun, Oracle, IBM y BEA a través del Java Community Process, consideran favorable continuar con el

desarrollo de la especificación completa y detallada para crear aplicaciones Java EE, el principal objetivo de esta era proporcionar un estándar simple para construir User Interface (UI) para aplicaciones Java Web. Esto dio lugar al Java Specification Request (JSR) #127 y fue donde nació JavaServer Faces.

JSF se basa en el patrón de diseño Modelo Vista Controlador (MVC) con componentes de desarrollo basado en interfaz de usuario que simplifica notoriamente el desarrollo, mediante el uso de etiquetas que generan contenido HyperText Markup Language (HTML).

JSF proporciona a los desarrolladores las siguientes características principales:

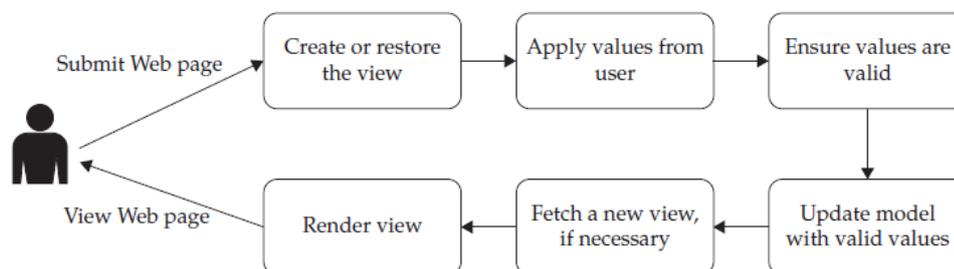
- Especificación de navegación para página.
- Componentes estándar de interfaz de usuario como campos de entrada, botones y enlaces.
- Validación de entrada de usuario.
- Fácil manejo de errores.
- Java Beans de gestión.
- Gestión de eventos.
- Soporte de internacionalización.

1.1.2 Ciclo de vida de JSF

El ciclo de vida de JSF inicia con una petición Hypertext Transfer Protocol (HTTP) por parte del usuario a través de un navegador y finaliza cuando el servidor responde con una página adecuada. HTTP es un protocolo sin estado, debido a esto no es capaz de recordar las transacciones anteriores entre el usuario y el servidor. JSF controla esto manteniendo vistas en el lado del servidor. Una vista es un árbol de componentes que representa la UI. Entonces, el ciclo de vida de JSF se preocupa de sincronizar las vistas del lado del servidor y lo que se le muestra al usuario.

El ciclo de vida de JSF posee seis fases mostrados en la Figura 2

Figura 2. Ciclo de vida de JSF



Fuente: JavaServer Faces 2.0, The Complete Reference

1. *Restore View*: En esta fase se construye el árbol de componentes.
2. *Apply Request Values*: En esta fase se leen los valores de la petición y se aplican sobre los componentes. Además se llama a los convertidores. Si existe algún error de conversión pasará directamente a la fase de “render response”.
3. *Process Validation*: En esta fase se validan los componentes. Si se encuentra algún fallo en las validaciones pasara directamente a la fase de “render response”.
4. *Update Model Value*: En esta fase los valores de los BackBean de JSF se actualizan con los valores de los componentes.
5. *Invoke application*: En esta fase se invoca a los métodos de los beans.
6. *Render Response*: En esta fase se presenta la respuesta al usuario. Dibuja los componentes del árbol de componentes.

1.1.3 Modelo de navegación JSF

Con JSF, la navegación entre las páginas de aplicación se define por un conjunto de reglas, estas reglas son configuradas comúnmente en el archivo de configuración “faces-config.xml”.

Una regla de navegación tiene una estructura definida como se muestra en la Figura 3, en donde se ve la navegación de “pagina1.jsf” a “pagina2.jsf”.

Figura 3. Estructura de regla de navegación.

```
<navigation-rule>
  <from-view-id>pagina1.jsf</from-view-id>
  <navigation-case>
    <from-outcome>aceptar</from-outcome>
    <to-view-id>pagina2.jsf</to-view-id>
  </navigation-case>
</navigation-rule>
```

Elaborado por: Víctor Quimbiamba

La Figura 3 muestra la estructura de una regla de navegación de JSF en donde:

<from-view-id>: Es el Localizador de Recurso Uniforme (URL) de la página de origen.

<navegation-case>: Es un caso de navegación.

<from-outcom>: Es la clave con el cual la regla va a ser invocada.

<to-view-id>: Es el URL de la página destino.

La etiqueta ‘navigation-rule’ puede tener uno o varios ‘navegation-case’ pero solo un “from-view-id”.

La forma de llamar a una regla de navegación es mediante una acción que puede ser en un botón, link, etc. El cual debe estar relacionado con un método en el Managed Bean, este método deberá retornar un String que coincida con el valor del ‘from-outcom’ de la regla de navegación.

Desde la versión 2.0 de JSF esta navegación se puede realizar de manera directa por medio de un método que devuelva un String que coincida con el URL de la nueva página.

En la figura 4 se observa como el método “irOtraPagina” realiza una regla de navegación directa.

Figura 4. Regla de navegación directa.

```
public String irOtraPagina{  
    //más código.  
    return "pagina2.jsf";  
}
```

Elaborado por: Víctor Quimbiamba

1.1.4 Managed Beans

Un Managed Bean es una clase Java que no extiende de ninguna otra clase, sino que posee una anotación la cual hace que se construya y se inyecte en el marco de JSF.

Como se aprecia en la Figura 5 un Managed Bean se registra usando una anotación sobre el nombre de la clase.

Figura 5. Registro de un Managed Bean con anotación.

```
@ManagedBean  
public class Clase {  
    .....  
}
```

Elaborado por: Víctor Quimbiamba

Alternativamente un Managed Bean también puede ser registrado en el archivo de propiedades “facesconfig.xml”, la Figura 6 muestra cómo se registraría un Managed Bean de una manera alternativa. La versión 2.0 de JSF solventa este proceso creando la anotación sobre el nombre de la clase.

Figura 6 – Registro de un Managed Bean en facesconfig.xml.

```
<managed-bean>  
    <managed-bean-name>claseBean</managed-bean-name>  
    <managed-bean-class>com.ejemplo.Clase</managed-bean-class>  
    <managed-bean-scope>session</managed-bean-scope>  
</managed-bean>
```

Elaborado por: Víctor Quimbiamba

1.1.4.1 Alcance de Managed Beans

El alcance de un Managed Bean describe el período de tiempo en que una instancia de un Managed Bean está disponible para otras partes del programa.

JSF incluye los siguientes tipos de alcances:

- **@NoneScoped**
Es un Bean no gestionado, se crea una instancia según la demanda de un Bean gestionado y se conservará mientras el Bean gestionado exista.
- **@RequestScoped**
Este tipo de Bean que se instancia bajo cualquier petición HTTP y perdurará hasta que exista una petición HTTP diferente a la que se creó.
- **@ViewScoped**
Este Bean se crea y estará disponible mientras el usuario permanezca en la misma vista.
- **@SessionScoped**
Este Bean se almacena en una sesión HTTP. Esto quiere decir que el Bean perdurará más allá de una petición HTTP.
- **@ApplicationScoped**
Este Bean mantiene los valores a lo largo de la duración de la aplicación y están disponibles para todos los usuarios
- **@CustomScoped**
Define una implementación propia de `java.util.Map`, generalmente incluidas en marcos de trabajos que extiende de JSF.

Todo Managed Bean tiene un “scope” o alcance, a partir de la versión 2.0 de JSF esto se especifica con una anotación en la clase como se observa en la figura 7.

Figura 7 – Registro del alcance a un Managed Bean con anotación.

```
@ManagedBean
@SessionScoped
public class Clase {
    .....
}
```

Elaborado por: Víctor Quimbiamba

1.1.5 Ventajas de JSF

JSF simplifica el desarrollo y es fácil de usar, genera código HTML a partir de etiquetas, es fácilmente extensible.

Existen muchos marcos de trabajo basados en JSF con componentes pre-diseñados, su estándar está muy bien definido y hay un gran soporte en la comunidad. Además de ser uno de los marcos de trabajo más usados en el mercado y recomendado.

JSF se integra dentro de la página Extensible HyperText Markup Language (XHTML) y se encarga de coleccionar y generar los valores de los elementos de la página.

El código JSF para crear vistas es muy parecido al HTML estándar.

JSF permite la introducción de código JavaScript en la página, para acelerar la respuesta de la vista.

JSF es ampliable, esto quiere decir que se pueden desarrollar nuevos componentes a partir de los existentes.

JSF resuelve validaciones, conversiones, mensajes de error e internacionalización.

Ofrece una notoria separación basado en el modelo MVC.

1.2 EJB 3.0.

EJB es una plataforma para crear aplicaciones de negocio escalable, portable y reutilizable con el lenguaje de programación Java.

Desde su primera incorporación, EJB se ha promocionado como un modelo de componente o un marco que permite crear aplicaciones empresariales usando Java sin tener que reinventar servicios tales como transacciones, seguridad y persistencia, automatizada para la construcción de una aplicación.

Los EJB's están diseñados para simplificar el proceso de construcción de aplicaciones de componentes empresariales distribuidos en Java. Son componentes del lado del servidor ejecutan lógica de negocio y se invocan desde clientes locales y remotos.

La versión EJB 3 está diseñada para simplificar mucho el desarrollo mediante la adopción de un modelo de programación Plain Old Java Object (POJO).

Como se muestra en la figura 8 el POJO se transforma en un EJB añadiéndole una anotación.

Figura 8. Registro de EJB



Fuente: EJB 3 in Action

1.2.1 Tipos de Beans

1.2.1.1 Entity Beans

Los beans de entidad se utilizan para representar un registro de una base de datos. Cada bean de entidad representa una tabla en la base de datos, cada instancia de esta clase representa un registro, lo que se consigue con los beans de entidad es crear un mapeo entre las propiedades de una clase y los campos de una tabla.

Un POJO se convierte en un bean de entidad cuando tiene la anotación @Entity un bean de entidad debe tener una clave primaria el cual identifica a ese registro como único en la tabla, el identificador se lo identifica con la anotación @Id, como se muestra en la figura 9.

Figura 9. Declaración de un Bean de entidad

```
@Entity
public class Usuario {

    @Id
    private int id;

    private String nombre;

    .
    .
    .
}
```

Elaborado por: Víctor Quimbiamba

1.2.1.2 Session Beans

Representan una acción o proceso de negocio que va a ser ejecutada de forma sincrónica.

Existen dos tipos de Beans de sesión.

Stateful - Con estado

El contenedor garantiza el mantenimiento del estado del componente entre distintas invocaciones se serializan y viceversa.

Stateless - Sin estado

El contenedor de EJBs no garantiza que un mismo EJB guarde su estado entre dos invocaciones del mismo cliente. Son más ligeros que los Stateful.

1.2.1.3 Message-Drive Beans

Representa una acción o proceso de negocio, pero los Beans son controlados por mensajes y se ejecutan de forma asíncrona.

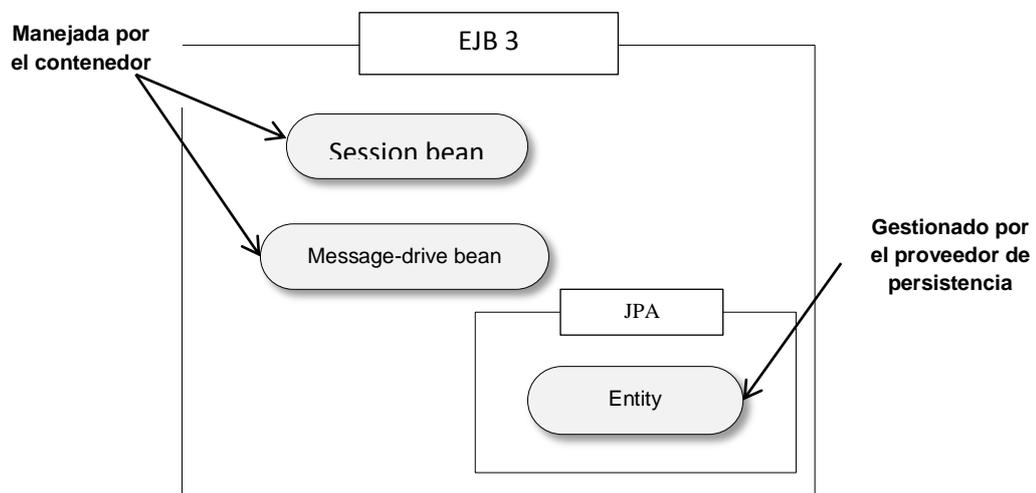
Cada tipo de Bean tiene un propósito y se puede utilizar un subconjunto específico de los servicios EJB. El propósito real de los tipos de Beans es proteger contra sobrecargas con los servicios. La clasificación también le ayuda a comprender y organizar la aplicación de una forma razonable.

En resumen los Beans de sesión y los Beans controlados por mensaje se utilizan para construir lógica de negocio, y viven en el contenedor.

El contenedor EJB gestiona los Bean y proporciona servicios a los mismos, además provee persistencia que gestiona entidades, como se puede ver en la figura 10.

El proveedor de persistencia es conectado dentro del contenedor EJB se abstrae detrás del Java Persistence API (JPA).

Figura 10. Contenedor EJB



Fuente: EJB 3 in Action
Elaborado por: Víctor Quimbiamba

1.2.2 Anotaciones EJB

A partir de la versión 3 de EJB se incorpora las anotaciones para hacer de un POJO un Bean, las anotaciones definen que tipo de Bean será el POJO, las anotaciones son:

Sesion beans

@Stateless

@Stateful

Message-driven beans

@MessageDrive

Entities Beans

@Entity

Los EJB's no son llamados directamente sino a través de su interfaz remota o local, en el caso de que el EJB no posea una interfaz el contenedor de EJB le proporciona una interfaz local de acceso.

1.3 Primefaces

Primefaces es un framework Open Source, está bajo la licencia de Apache License V2 desarrollada y mantenida por Prime Technology, que cuenta con un conjunto de componentes JSF, soporta Asynchronous JavaScript And XML (AJAX) con despliegue nativo incluyendo Push/Comet.

Primefaces posee etiquetas fáciles de usar y componentes muy ricos en funcionalidad. Posee una documentación bastante amplia, además como todo proyecto Open Source existe una comunidad siempre activa que reporta y corrige bugs.

Es un framework ligero y no necesita dependencias ni configuración para su funcionamiento, está diseñado para ser extendido, posee más de 35 temas visuales y herramientas de edición.

Usa JQuery como framework JavaScript, además posee una versión para dispositivos móviles.

1.3.1 Comparativa

Tabla 1. Comparativa Primefaces con otros frameworks basados en JSF

	Richfaces	Icefaces	Primefaces
Longitud de la documentación	47442 bytes	76072 bytes	53904 bytes
Solicitudes por segundo	28.59 [#/sec]	11.12 [#/sec]	33.46 [#/sec]
Tiempo por petición	174.869 [ms]	899.281 [ms]	149.422 [ms]
Velocidad de transferencia	1329.59 [Kbytes/sec]	829.41 [Kbytes/sec]	1767.06 [Kbytes/sec]
Tiempos de conexión (ms)	766	12969	1094

Fuente: <http://www.javahispano.org>

Elaborado por: Víctor Quimbiamba

1.4 Oracle Database

Oracle Database es un sistema para la gestión de bases de datos relacionales, es una solución integral considerada uno de los sistemas de bases de datos más completos.

Las nuevas versiones de Oracle Database proporcionan a los usuarios la posibilidad de aumentar su eficiencia, productividad y capacidades de respuesta debido a que posee soporte de transacciones, es decir aseguran que las transacciones no se queden a medio camino cuando se haga un trabajo múltiple sobre la misma tabla, además posee escalabilidad y soporte para multimedia.

El acceso por consulta Structured Query Language (SQL), a Oracle Database se pueden realizar mediante PL/SQL, también es posible realizar consultas a la base de datos Oracle mediante SQL plus.

1.4.1 Historia

Oracle surge a finales el año 1970 del nombre de Relational Software a partir de un estudio sobre Sistema de Gestión de Bases de Datos (SGBD) Computer World definió este estudio como uno de los más completos jamás escritos sobre bases de datos. Usaba la filosofía de las bases de datos relacionales, algo que por aquella época era todavía desconocido.

La tecnología Oracle se encuentra prácticamente en todas las industrias alrededor del mundo.

Oracle es la primera compañía de software que desarrolla e implementa software para empresas 100 por ciento activado por Internet a través de toda su línea de productos: base de datos, aplicaciones comerciales y herramientas de desarrollo de aplicaciones y soporte de decisiones.

Oracle garantiza el funcionamiento de sus bases de datos, que en caso de caídas del servidor compensa económicamente con cifras cercanas a las 7 cifras. Tomado de <https://iessanvicente.com/colaboraciones/oracle.pdf>

1.4.2 ¿Por qué usar Oracle Database?

Posee múltiples versiones para los sistemas operativos más usados como UNIX, LINUX, Windows, y otros.

Ofrece soporte mundial a través de sus centros de soporte y sus sitios Web.

Es la Base de Datos más utilizada por las empresas más grandes del mundo, debido a su robustez y seguridad.

Siempre que la licencia lo proporcione, suministrará actualización de versiones gratis al cliente.

1.4.3 Ventajas de usar Oracle Database:

- Uso de herramienta líder y de alta experiencia.
- Asegura soporte.
- Provee una excelente oportunidad profesional.
- Tiene una amplia suite EBusiness del mercado.
- Posee productos para interactuar con otras Bases de Datos.

1.5 Glassfish

Es un servidor ligero, desarrollado para la plataforma Java Enterprise Edition y servicios Web Java, ofreciendo rendimiento, escalabilidad y fiabilidad.

1.5.1 Diseño modular

En GlassFish Server Versión 3.x, el código del servidor se divide en módulos para proporcionar flexibilidad y rendimiento en tiempo de ejecución mejorada.

El diseño modular ofrece la posibilidad de:

- Implementar paquetes Open Services Gateway Initiative (OSGi).
- Implementar biblioteca de Java Archive.
- Reemplazar la funcionalidad existente con otra aplicación.

1.5.2 Actualización e integración de la herramienta

La herramienta de actualización está integrada en la consola de administración de Sun GlassFish Enterprise Server 3.0.1. Esta herramienta facilita la administración de componentes y aplicaciones relacionadas que están disponibles para la ampliación de funciones.

La consola de administración proporciona acceso a la página de la herramienta de actualización a través del árbol de navegación. Entre las opciones se tiene:

- Los componentes que se instalan.
- Actualizaciones disponibles para los componentes instalados.
- Componentes adicionales que están disponibles y se pueden instalar.

1.6 Eclipselink JPA

JPA es el estándar Java para mapear objetos Java a una base de datos relacional. A pesar de que los productos propietarios de mapeo como Hibernate y TopLink todavía existen, son ahora enfocadas en proveer su funcionalidad a través de la API de JPA, lo que permite que todas las aplicaciones sean portables a través de las implementaciones de JPA.

1.6.1 Mapeo de objetos

La unidad básica de la persistencia de JPA es la entidad, que no es más que una clase Java con metadatos para describir una tabla de base de datos. Los metadatos pueden estar en la forma de anotaciones en la clase de entidad en sí, o puede ser un archivo Extensible Markup Language (XML) adjunto, pero se utilizan anotaciones ya que son más fáciles de entender y usar.

Cada clase de entidad debe estar anotado sobre el nombre la clase con “@Entity” y un campo de identificador, indicado por “@Id”, que se asigna a la columna de clave principal en la base de datos. Además se debe especificar la tabla a la cual hace referencia la entidad con la anotación “@Table” y la propiedad “name”

Cuando una entidad tiene un campo que hace referencia a una o más entidades, JPA provee un mecanismo para describir las relaciones entre tablas con anotaciones.

Para especificar el nombre de la columna en un atributo de la entidad se utiliza la

anotación “@Column” y la propiedad “name”, si el nombre del atributo coincide con el nombre no es necesario agregar la anotación, como se puede ver en la figura 11.

Figura 11. Entidad JPA

```
@Table(name = "persona")
@Entity
public class Persona implements Serializable {

    @Id
    private Long id;

    @Column(name="nombre_completo")
    private String nombreCompleto;

    @Column
    private Integer edad;

    .
    .
    .
    .
    .
```

Elaborado por: Víctor Quimbiamba

CAPÍTULO 2

METODOLOGÍA RUP

Rational Unified Process (RUP) es un proceso integral probado para el desarrollo de software, fue desarrollado por la empresa Rational Software, actualmente propiedad de IBM. Junto con el Unified Modeling Language (UML), forma la metodología estándar más utilizada para el análisis, diseño, implementación y documentación de sistemas orientados a objetos. Ofrece orientación sobre las mejores prácticas adecuadas para el desarrollo.

2.1 Características:

- Una de las principales características del RUP es la de ser iterativo e incremental.
- Pretende implementar las mejores prácticas de Ingeniería de Software.
- Existe un claro control de cambios.
- Asegura continuamente la calidad.
- Se adapta a las necesidades del cliente.
- Forma disciplinada de asignar tareas y responsabilidades, quién, cuándo y cómo hacerlo.
- Uso de arquitectura basado en componentes.

2.2 Definición fases

2.2.1 Inicio

En la fase inicial se plantea la idea central del producto, se construye el documento de visión, se examinan y confirma el entendimiento sobre los objetivos centrales del negocio.

La fase de inicio establece la viabilidad del producto y delimita el alcance del proyecto.

2.2.2 Elaboración

En la fase de elaboración se analiza el dominio del problema, la mayoría de los Casos de Uso son especificados en detalle y la arquitectura del sistema es diseñada.

En esta fase se identifican los riesgos significativos y se prepara el calendario, el equipo de trabajo y el costo del proyecto.

2.2.3 Construcción

En la fase de construcción se enfoca en alcanzar la capacidad operativa del producto de forma incremental a través de las sucesivas iteraciones, así también se refina el diseño para llevarlo a código fuente basado en las líneas base de la arquitectura.

2.2.4 Transición

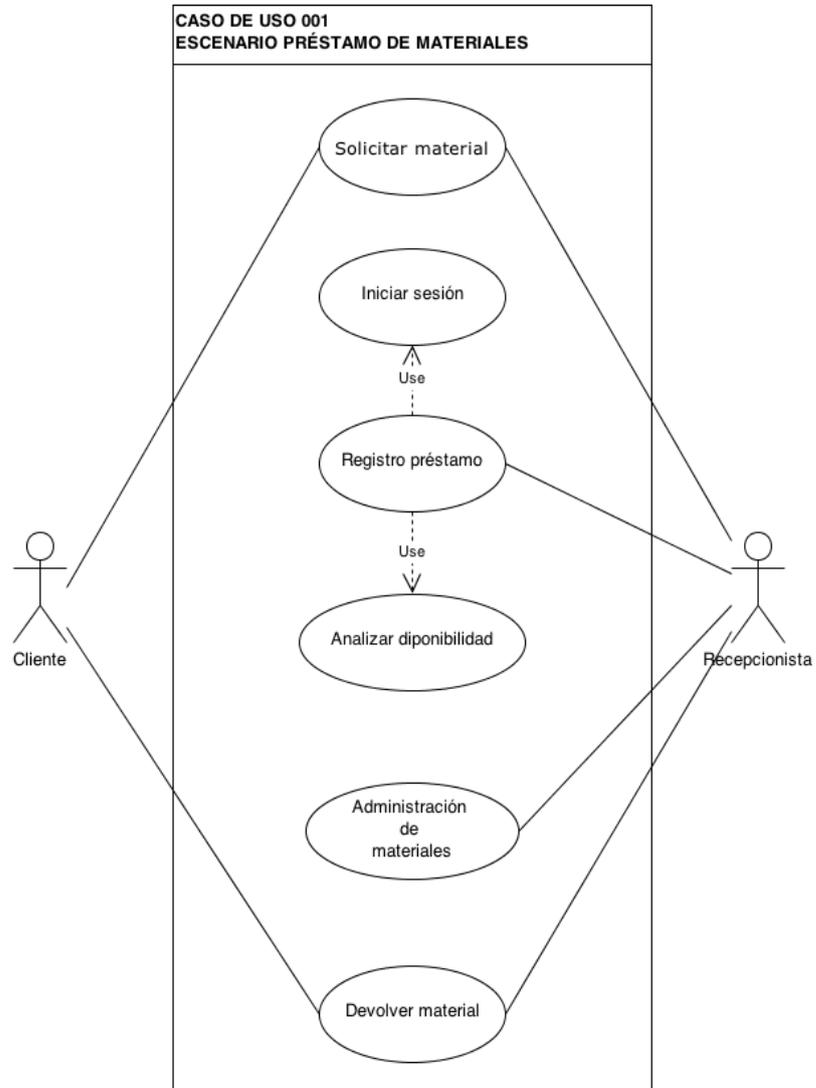
En la fase de transición se garantiza que el software esté listo para la entrega al usuario.

2.3 Fases

2.3.1 Casos de uso

2.3.1.1 Préstamo de materiales

Figura 12. Caso de Uso: Préstamo de Materiales



Elaborado por: Víctor Quimbiamba

En este caso de uso el cliente realiza la solicitud de un préstamo de material para que el recepcionista verifique y registre el préstamo.

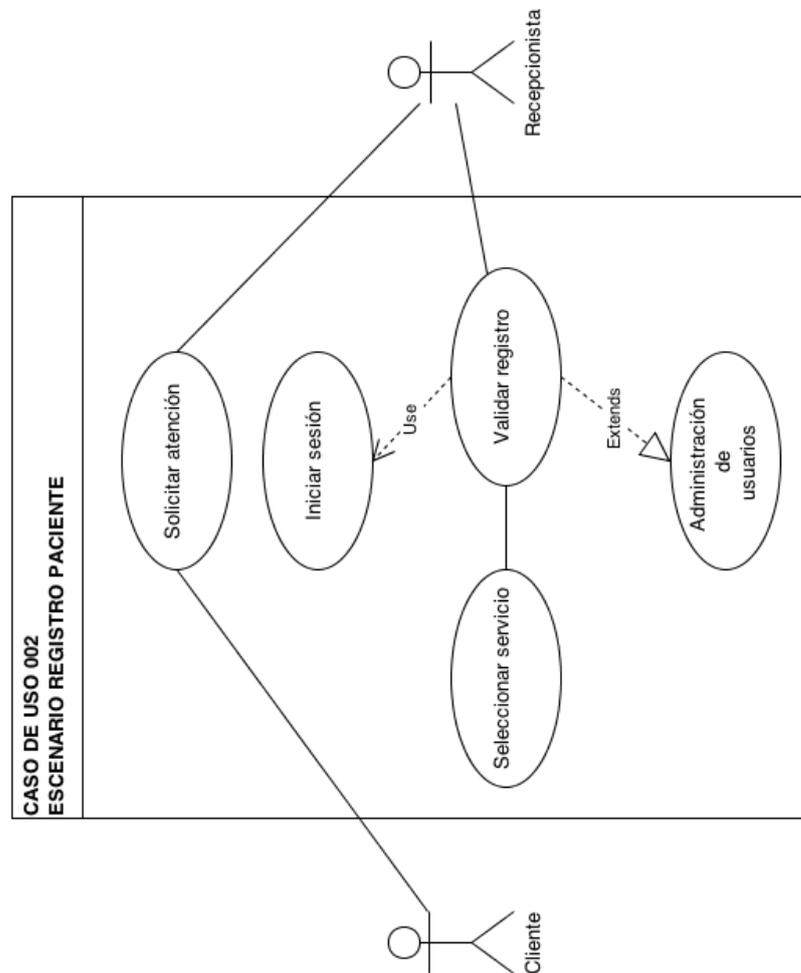
Tabla 2. Especificación del caso de uso: Préstamo de Materiales

Especificación del caso de uso: Préstamo de Materiales	
Código	CU001
Nombre	Préstamo de material
Descripción	En este caso de uso el cliente realiza la solicitud de un préstamo de material para que el recepcionista verifique y registre el préstamo.
Actores	Cliente, recepcionista - Usuario
Precondiciones	El cliente estudiante debe presentar un documento de identificación y el carnet estudiantil. El cliente docente debe presentar un documento de identificación. El material deberá estar disponible.
Pos condiciones	El material es entregado y se registra el préstamo cambiando de estado al material, hasta que es devuelto.
Flujo normal	<ol style="list-style-type: none"> 1. El cliente solicita un material. 2. El recepcionista ingresa al sistema. 3. El recepcionista verifica disponibilidad. 4. El cliente recibe un material. 5. El recepcionista registra el préstamo. 6. El cliente devuelve el material. 7. El cliente registra la devolución.
Flujo alternativo	<ol style="list-style-type: none"> 1. El cliente solicita un material. 2. El recepcionista ingresa al sistema 3. El recepcionista verifica disponibilidad. 4. El recepcionista informa no disponibilidad
Excepciones	El material no está disponible.

Elaborado por: Víctor Quimbiamba

2.3.1.2 Registro paciente

Figura 13. Caso de Uso: Registro Paciente.



Elaborado por: Víctor Quimbiamba

En este caso de uso el cliente realiza la solicitud de atención y en el caso de no constar como paciente el recepcionista realiza el proceso de registro.

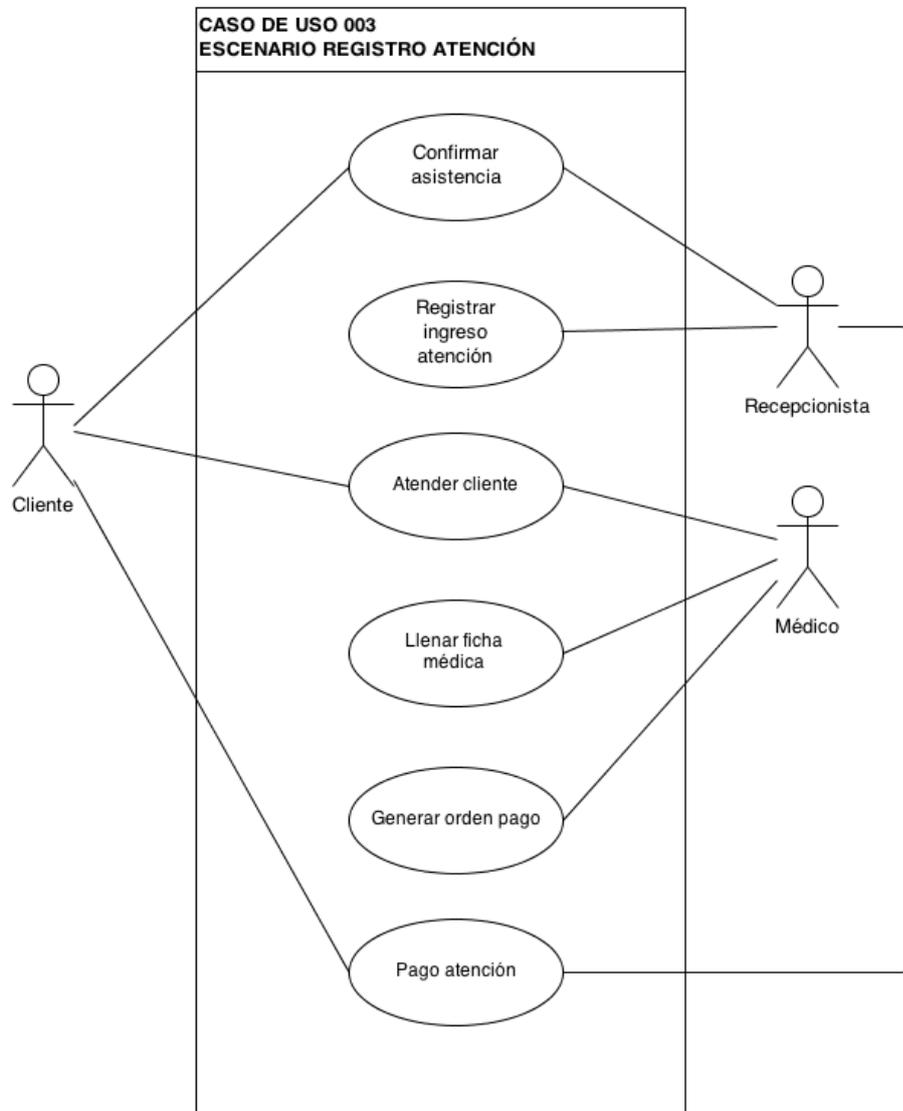
Tabla 3. Especificación del caso de uso: Registro Paciente

Especificación del caso de uso: Registro Paciente	
Código	CU002
Nombre	Registro paciente
Descripción	En este caso de uso el cliente realiza la solicitud de atención y en el caso de no constar como paciente el recepcionista realiza el proceso de registro.
Actores	Cliente, recepcionista
Precondiciones	El cliente no debe estar registrado en el sistema.
Pos condiciones	Los datos son verificados y el cliente es registrado.
Flujo normal	<ol style="list-style-type: none"> 1. El cliente solicita una atención. 2. El recepcionista ingresa al sistema. 3. El recepcionista verifica registro paciente. 4. El recepcionista registra el paciente.
Flujo alternativo	<ol style="list-style-type: none"> 1. El cliente solicita una atención. 2. El recepcionista ingresa al sistema. 3. El recepcionista verifica registro paciente. 4. El recepcionista informa que el cliente ya está registrado.
Excepciones	El cliente está registrado.

Elaborado por: Víctor Quimbiamba

2.3.1.3 Registro atención

Figura 14. Caso de Uso: Registro Atención.



Elaborado por: Víctor Quimbiamba

En este caso de uso el usuario recepcionista confirma la cita con el cliente, una vez confirmada el cliente asiste a la atención con el usuario médico. El usuario médico registra la atención y genera la orden de pago. El cliente realiza el pago y el usuario recepcionista registra el pago y la finalización de la cita.

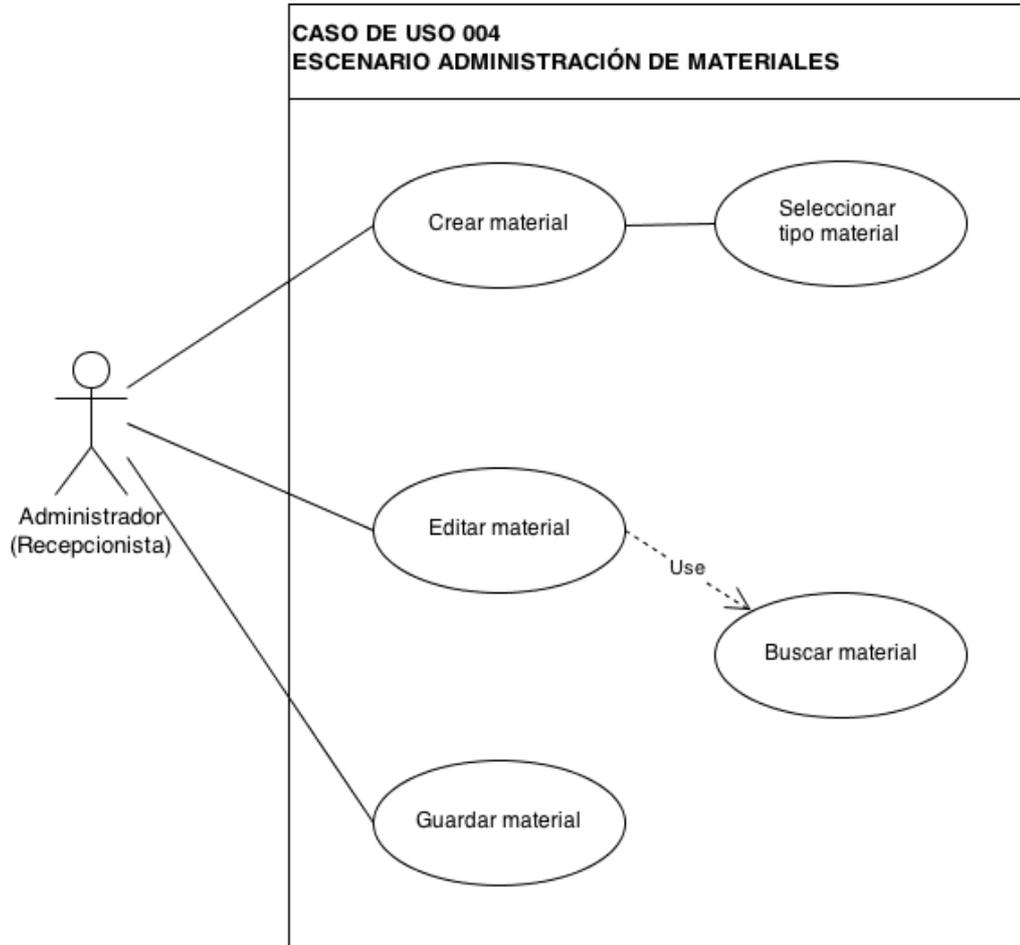
Tabla 4. Especificación del caso de uso: Registro Atención

Especificación del caso de uso: Registro Atención	
Código	CU003
Nombre	Registro atención
Descripción	En este caso de uso el usuario recepcionista confirma la cita con el cliente, una vez confirmada el cliente asiste a la atención con el usuario médico. El usuario médico registra la atención y genera la orden de pago. El cliente realiza el pago y el usuario recepcionista registra el pago y la finalización de la cita.
Actores	Cliente, usuario recepcionista, usuario médico.
Precondiciones	El usuario recepcionista verifique la confirmación con el cliente. El cliente asiste a la cita.
Pos condiciones	El usuario médico atiende al cliente. El usuario recepcionista registra el pago.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario recepcionista verifica la confirmación. 2. El cliente asiste a la cita. 3. El usuario médico atiende al cliente. 4. El usuario médico genera la orden de pago. 5. El cliente realiza el pago. 6. El usuario recepcionista registra el pago y se finaliza la cita.
Flujo alternativo	<ol style="list-style-type: none"> 1. El usuario recepcionista verifica la confirmación. 2. El cliente cancela la cita. 3. El usuario recepcionista registra la cancelación de la cita.
	<ol style="list-style-type: none"> 1. El usuario recepcionista verifica la confirmación. 2. El cliente asiste a la cita. 3. El usuario médico no asiste a la atención. 4. El usuario recepcionista registra cambio de cita.
Excepciones	El usuario cancela la cita. El médico no realiza la atención.

Elaborado por: Víctor Quimbiamba

2.3.1.4 Administración de materiales

Figura 15. Caso de Uso: Administración de Materiales



Elaborado por: Víctor Quimbiamba

En este caso de uso se administra los materiales, se crean y editan.

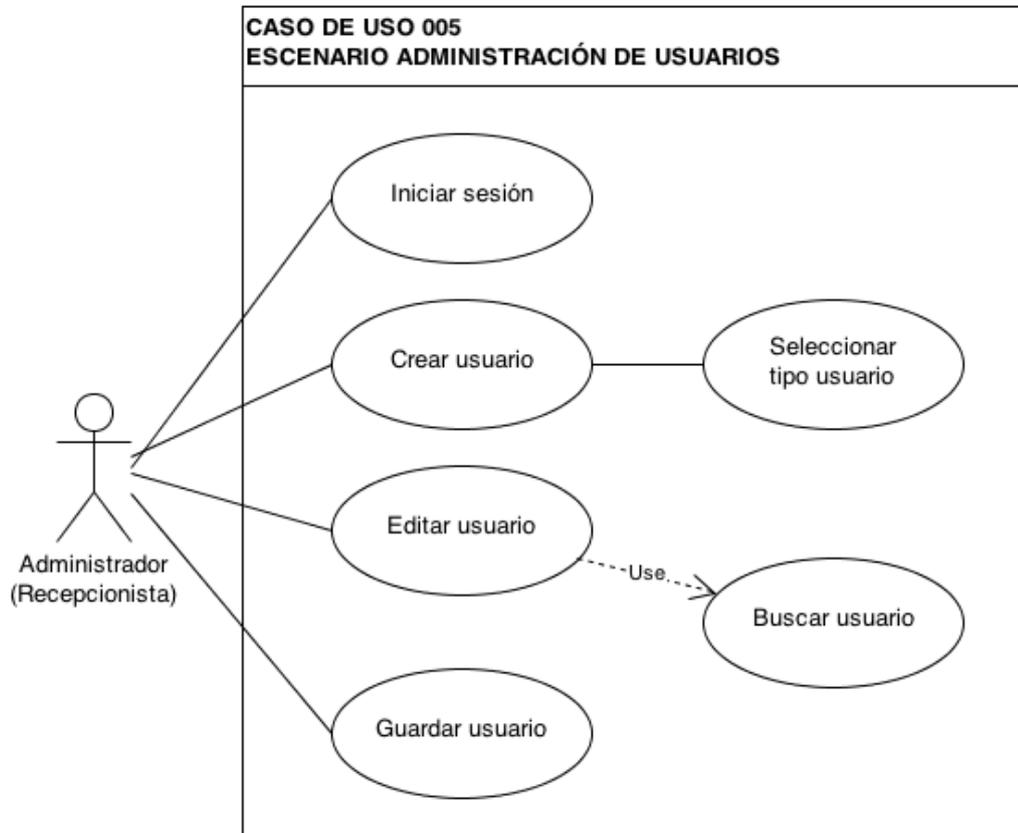
Tabla 5. Especificación del caso de uso: Administración de Materiales

Especificación del caso de uso: Administración de Materiales	
Código	CU004
Nombre	Administración de materiales
Descripción	En este caso de uso se administra los materiales para tener un catálogo.
Actores	Usuario recepcionista.
Precondiciones	Registrar un tipo de material. Conocer el estado del material.
Pos condiciones	Se registra el material.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario recepcionista recibe un material. 2. Se usuario recepcionista crea un material. 3. El usuario recepcionista edita un material. 4. EL usuario recepcionista guarda los cambios.
Flujo alternativo	
Excepciones	El nombre del material ya está siendo usado.

Elaborado por: Víctor Quimbiamba

2.3.1.5 Administración de usuarios

Figura 16. Caso de Uso: Administración de Usuarios



Elaborado por: Víctor Quimbiamba

En este caso de uso se describe la administración de los usuarios que usan el sistema.

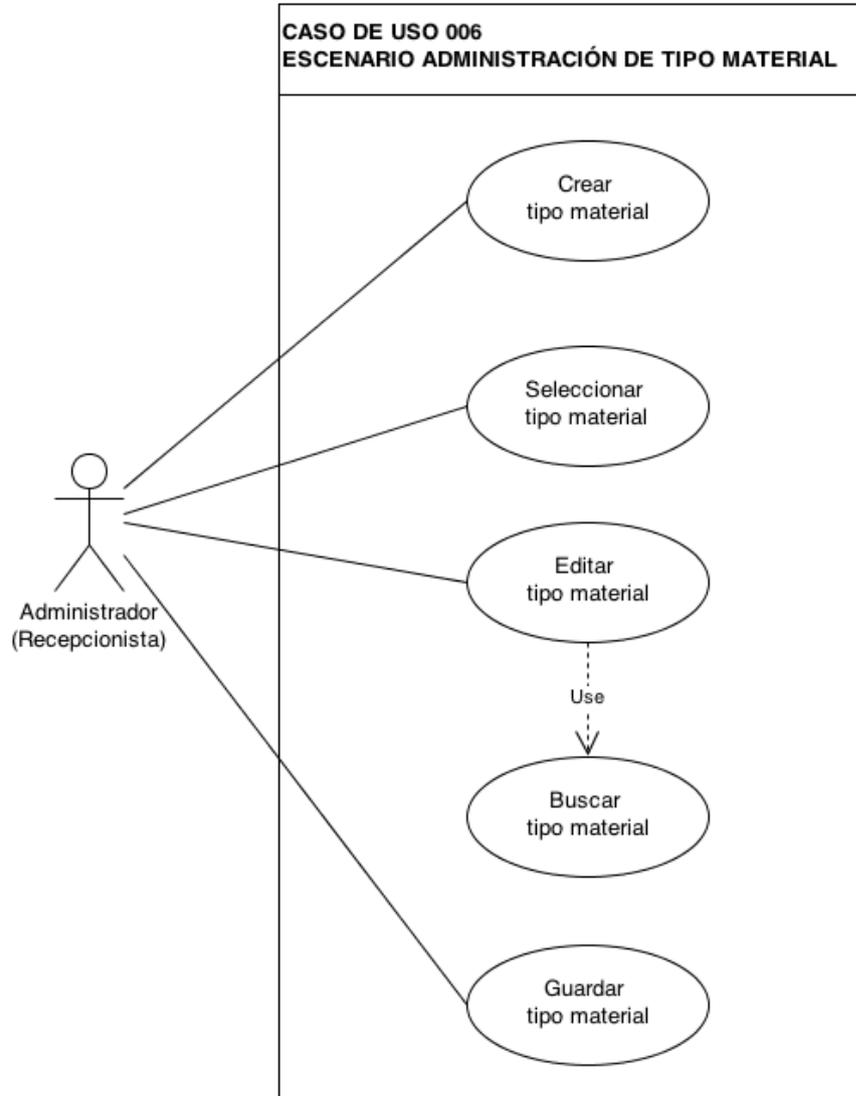
Tabla 6. Especificación del caso de uso: Administración de Usuarios

Especificación del caso de uso: Administración de Usuarios	
Código	CU005
Nombre	Administración de usuarios
Descripción	En este caso de uso se describe la administración de los usuarios que usan el sistema.
Actores	Usuario recepcionista.
Precondiciones	Usuario pertenezca a la institución.
Pos condiciones	El usuario será registrado en el sistema.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario recepcionista registra el nuevo usuario. 2. El usuario recepcionista edita un usuario. 3. El usuario recepcionista elimina un usuario. 4. El usuario recepcionista guarda los cambios.
Flujo alternativo	
Excepciones	

Elaborado por: Víctor Quimbiamba

2.3.1.6 Administración de tipo de material

Figura 17. Caso de uso: Administración de Tipo Material



Elaborado por: Víctor Quimbiamba

En este caso de uso se administra el Tipo de Material

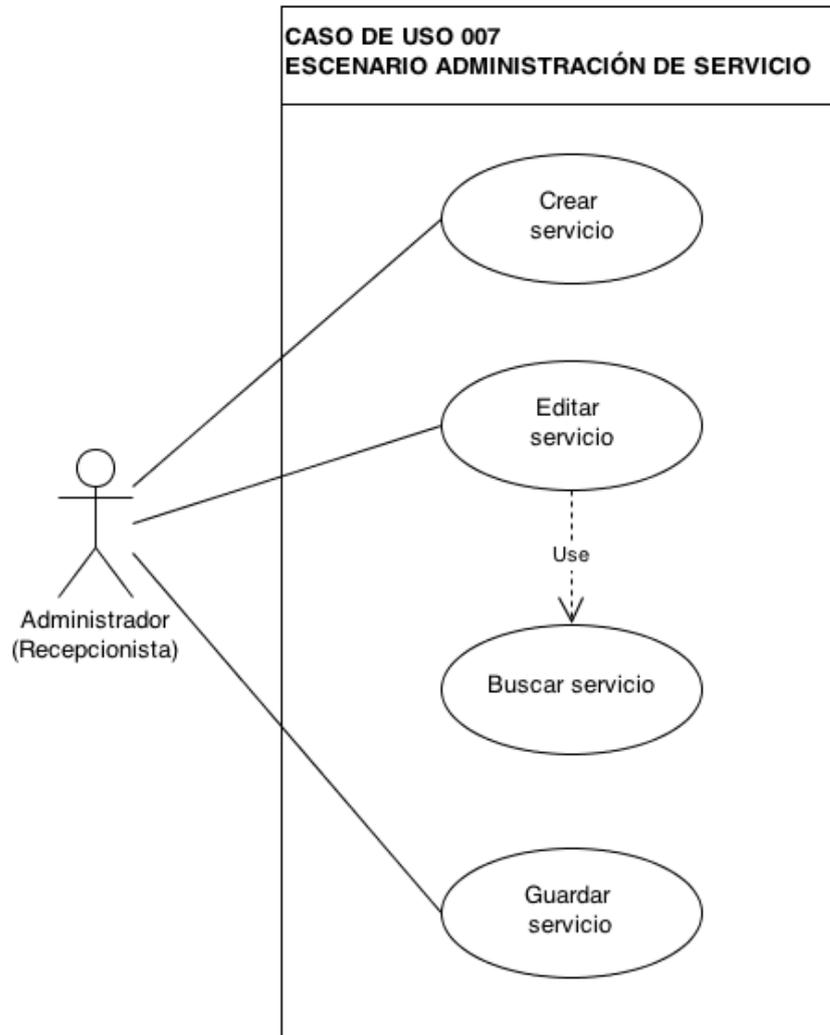
Tabla 7. Especificación del caso de uso: Administración de Tipo Material

Especificación del caso de uso: Administración de tipo material	
Código	CU006
Nombre	Administración de tipo material
Descripción	En este caso de uso se administra el tipo de material
Actores	Cliente, usuario recepcionista, usuario médico.
Precondiciones	Tipo material no este registrado.
Pos condiciones	El tipo de material se registre correctamente.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario recepcionista registra un nuevo tipo de material. 2. El usuario recepcionista edita un tipo de material. 3. El usuario recepcionista guarda los cambios.
Flujo alternativo	
Excepciones	El tipo de material ya existe.

Elaborado por: Víctor Quimbiamba

2.3.1.7 Administración de servicio

Figura 18. Caso de Uso: Administración de Servicio



Elaborado por: Víctor Quimbiamba

En este caso de uso se administran los Servicios

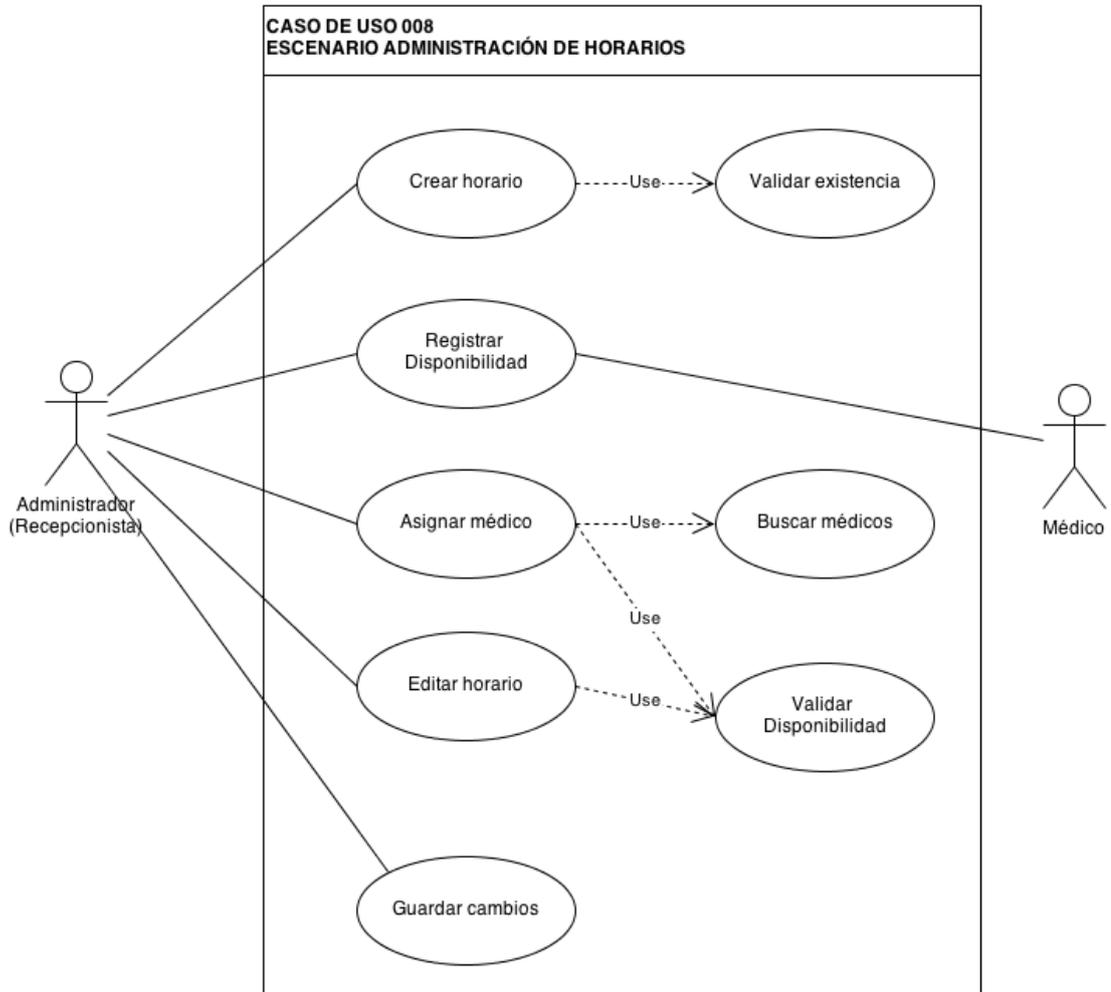
Tabla 8. Especificación del caso de uso: Administración de Servicio

Especificación del caso de uso: Administración de servicio	
Código	CU007
Nombre	Administración de servicio
Descripción	En este caso de uso se administran los servicios
Actores	Usuario recepcionista.
Precondiciones	Servicio no registrado.
Pos condiciones	El Servicio sea registrado correctamente.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario recepcionista registra un nuevo servicio. 2. El usuario recepcionista edita un servicio. 3. El usuario recepcionista guarda los cambios.
Flujo alternativo	
Excepciones	El servicio ya está registrado.

Elaborado por: Víctor Quimbiamba

2.3.1.8 Administración de horarios

Figura 19. Caso de Uso: Administración de Horarios



Elaborado por: Víctor Quimbiamba

En este caso de uso se administran los Horarios de los usuarios Médicos.

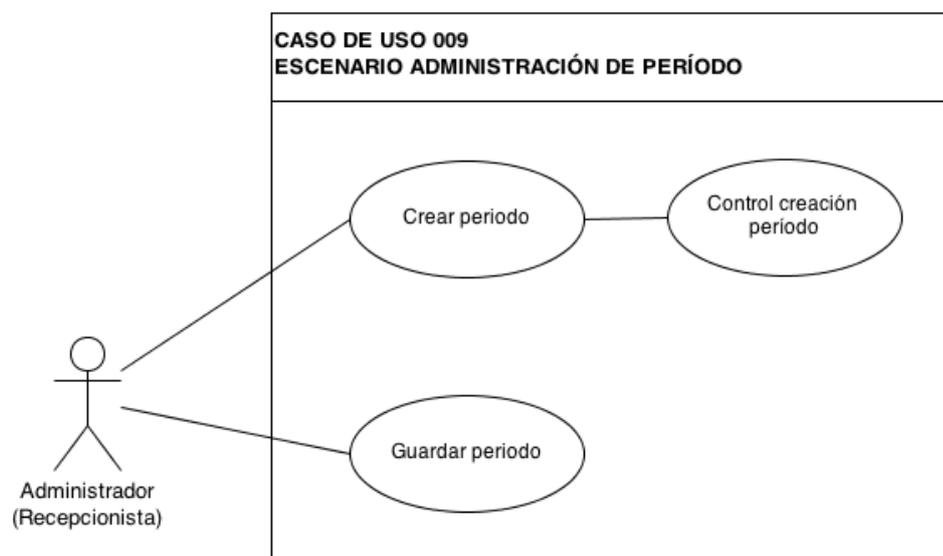
Tabla 9. Especificación del caso de uso: Administración de Horarios

Especificación del caso de uso: Administración de horarios	
Código	CU008
Nombre	Administración de horarios
Descripción	En este caso de uso se administran los horarios de los usuarios médicos.
Actores	Usuario recepcionista, usuario médico.
Precondiciones	Usuario médico este registrado.
Pos condiciones	Los horarios se registrarán correctamente.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario recepcionista registra un nuevo horario para un usuario médico. 2. El usuario Médico informa disponibilidad. 3. El usuario recepcionista edita un horario. 4. El usuario recepcionista elimina un horario. 5. El usuario recepcionista guarda los cambios.
Flujo alternativo	
Excepciones	El horario ya esté siendo ocupado.

Elaborado por: Víctor Quimbiamba

2.3.1.9 Administración de período

Figura 20. Caso de Uso: Administración de Período



Elaborado por: Víctor Quimbiamba

Tabla 10 – Especificación del caso de uso: Administración de Periodos

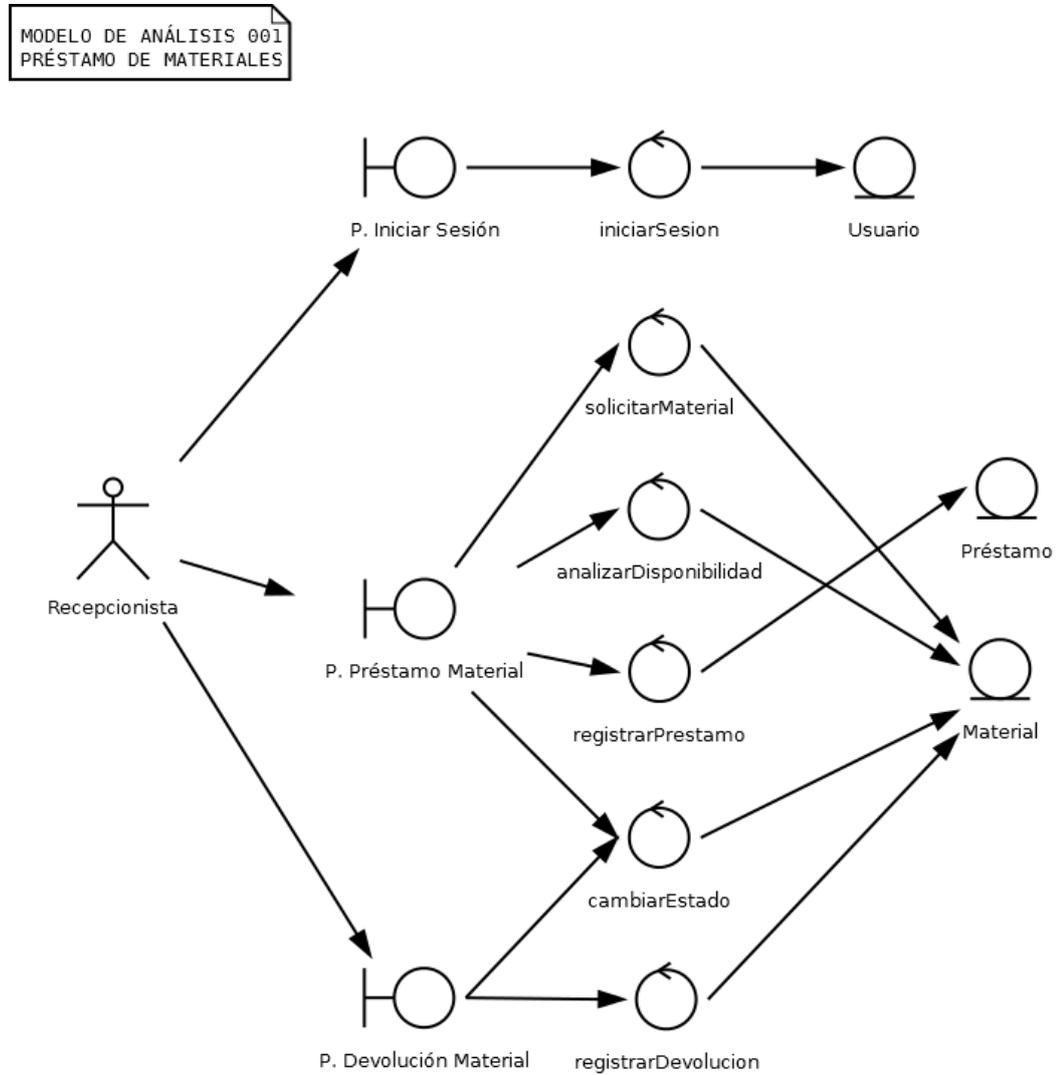
Especificación del caso de uso: Administración de periodo	
Código	CU009
Nombre	Administración de periodo
Descripción	En este caso de uso se administran los periodos con los que trabaja la aplicación.
Actores	Usuario recepcionista
Precondiciones	El periodo anterior haya finalizado No exista un periodo
Pos condiciones	La aplicación actualice su funcionamiento al nuevo periodo
Flujo normal	<ol style="list-style-type: none"> 1. Inicia con la lista de periodos 2. Crea un nuevo periodo, confirma la creación. 3. La aplicación solicita inicio de sesión.
Flujo alternativo	<ol style="list-style-type: none"> 1. Inicia con la lista de periodo 2. No se crea el periodo
Excepciones	

Elaborado por: Víctor Quimbiamba

2.3.2 Modelo de análisis

2.3.2.1 Préstamo de materiales

Figura 21. Modelo de Análisis: Préstamos de Materiales

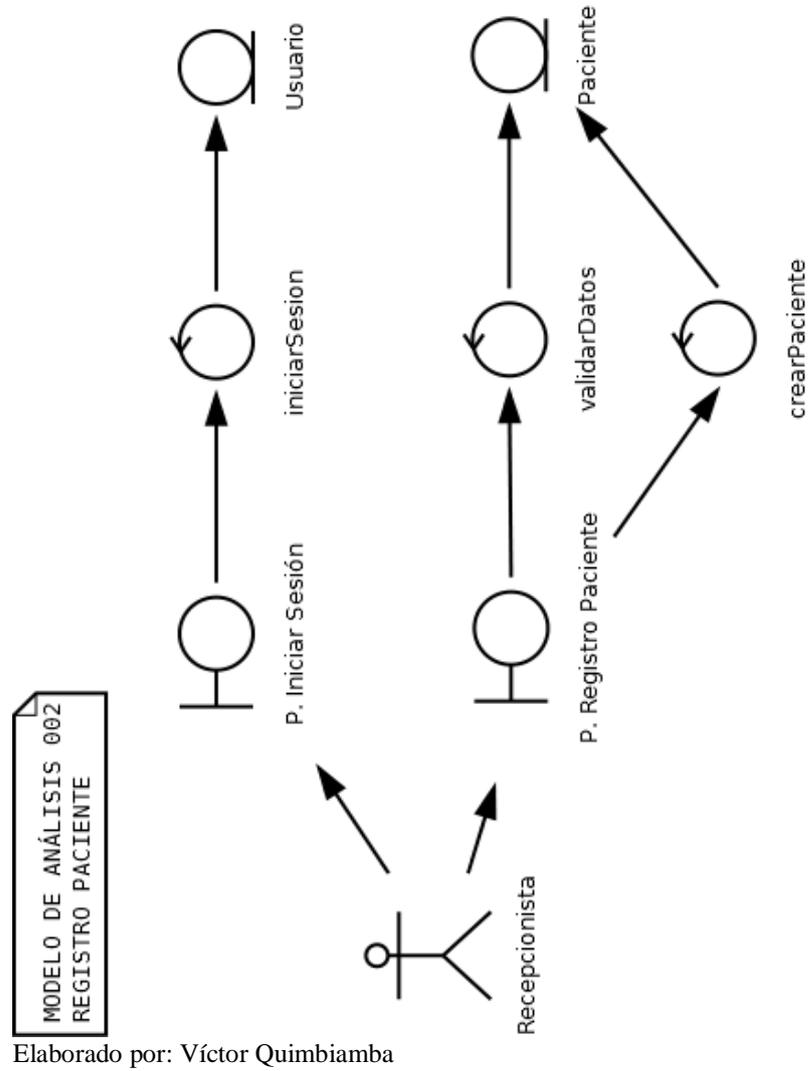


Elaborado por: Víctor Quimbiamba

La figura 21 describe el modelo de análisis del escenario en el préstamo de un material.

2.3.2.2 Registro paciente

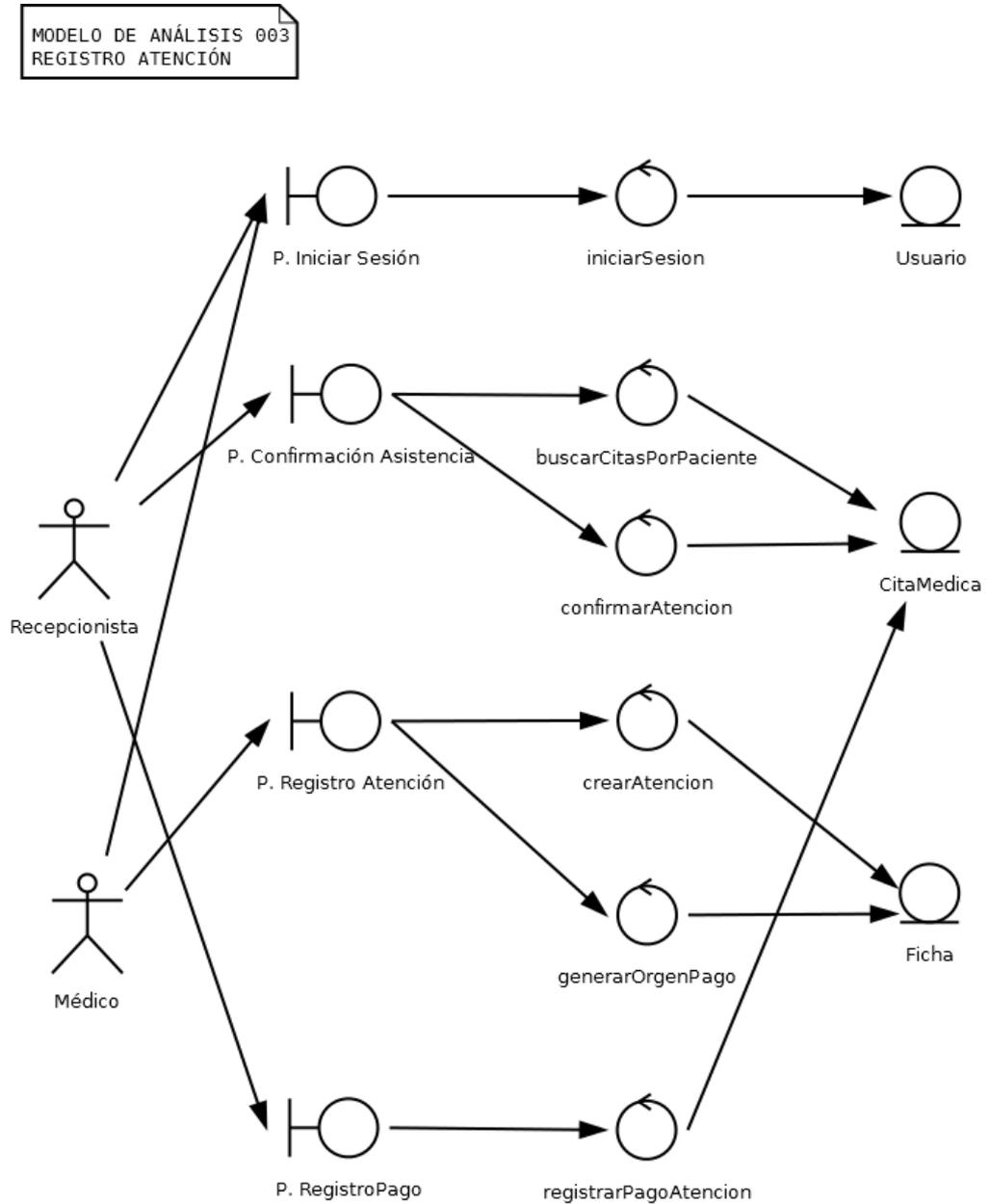
Figura 22. Modelo de Análisis: Registro Paciente



La figura 22 describe el modelo de análisis del escenario de registro de paciente.

2.3.2.3 Registro de atención

Figura 23. Modelo de Análisis: Registro de Atención

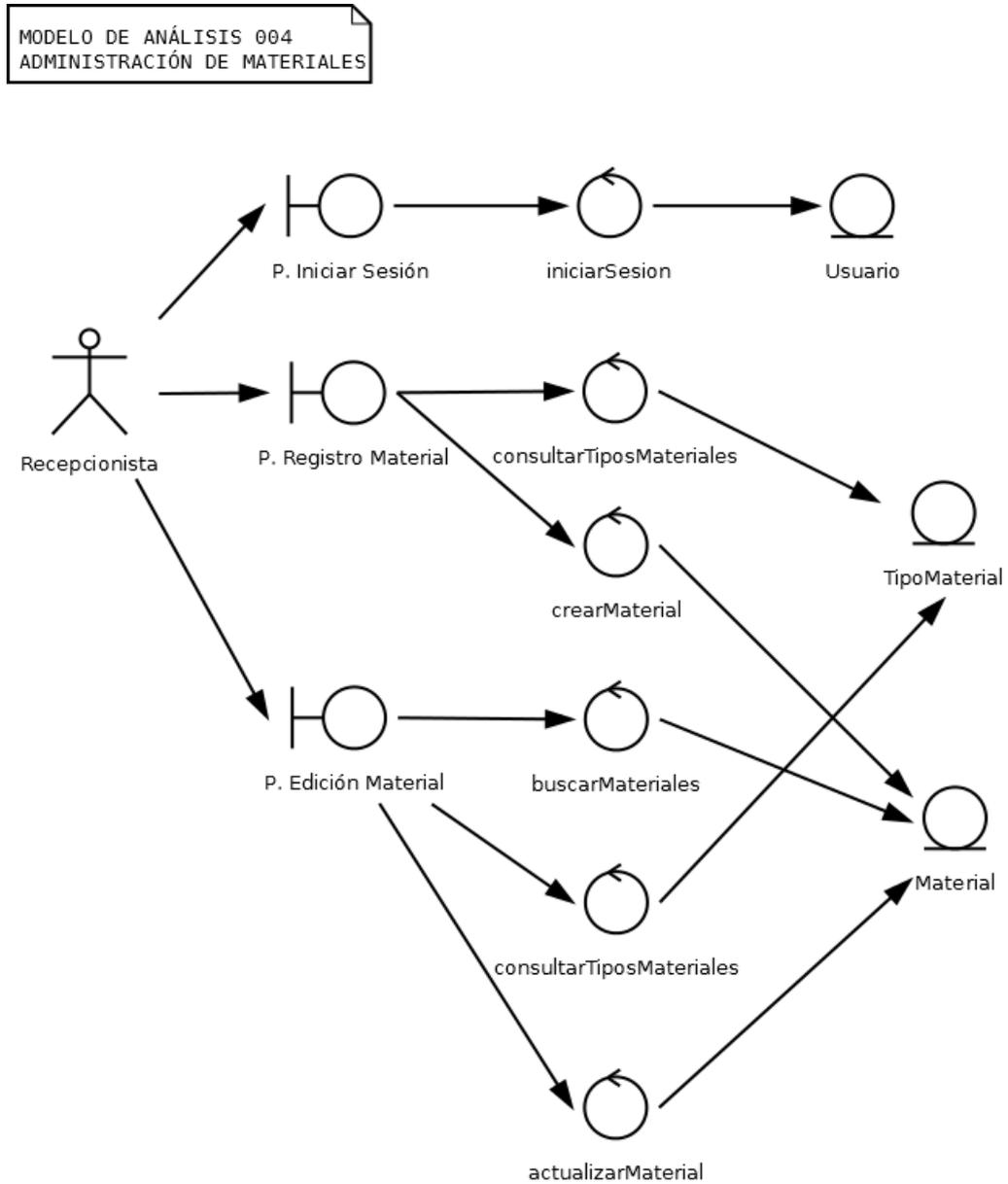


Elaborado por: Víctor Quimbiamba

La figura 23 describe el modelo de análisis del escenario de registro de atención.

2.3.2.4 Administración de materiales

Figura 24. Modelo de Análisis: Administración de Materiales

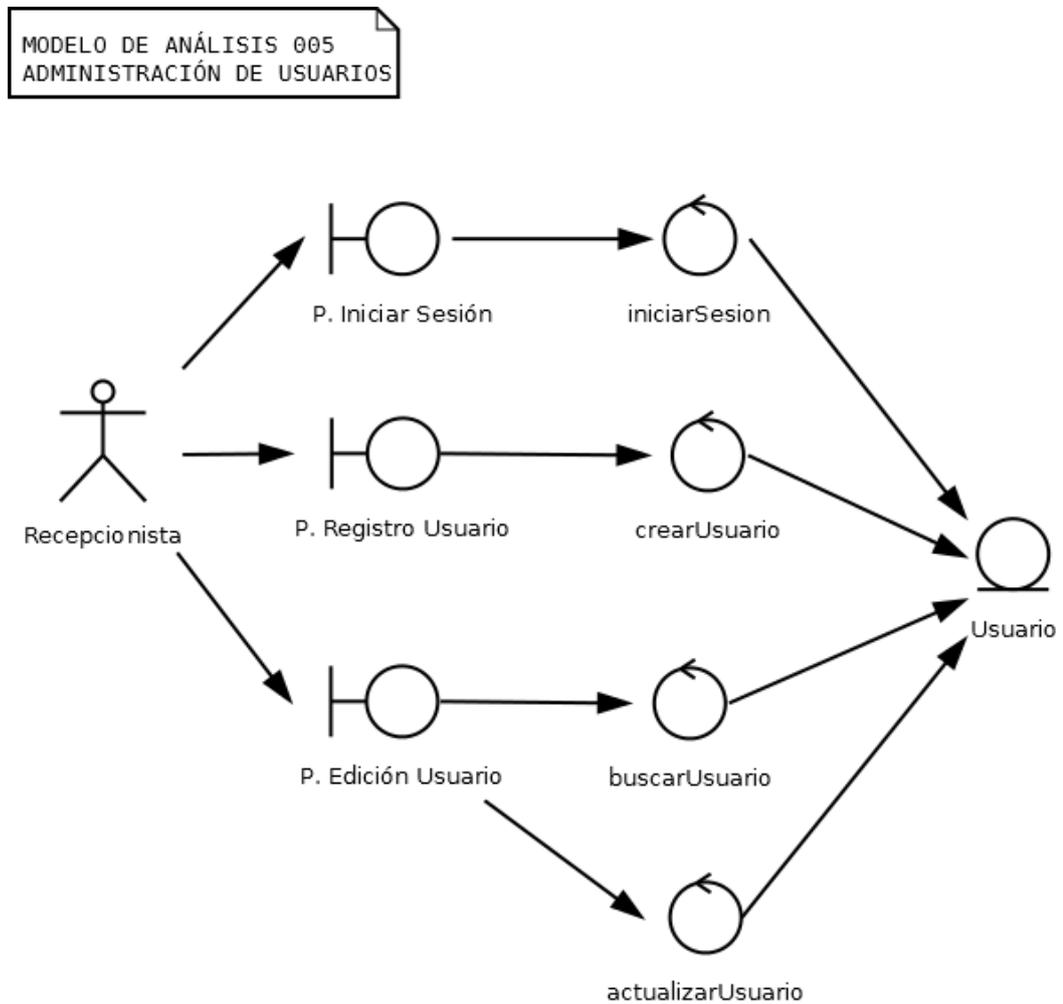


Elaborado por: Víctor Quimbiamba

La figura 24 describe el modelo de análisis del escenario de administración de materiales.

2.3.2.5 Administración de usuarios

Figura 25. Modelo de Análisis: Administración de Usuarios

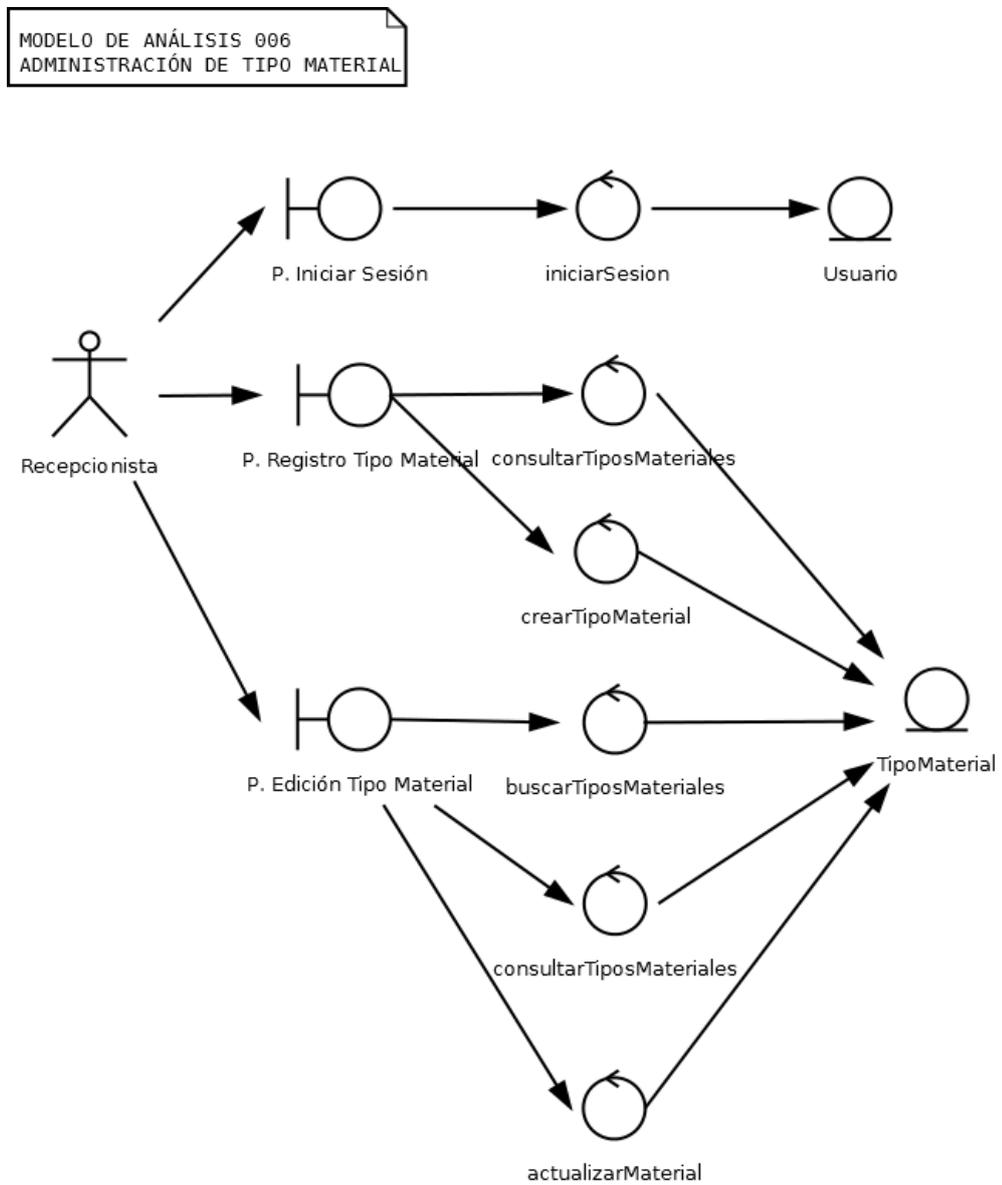


Elaborado por: Víctor Quimbiamba

La figura 25 describe el modelo de análisis del escenario de administración de usuarios.

2.3.2.6 Administración de tipo material

Figura 26 – Modelo de Análisis: Administración de Tipo Material

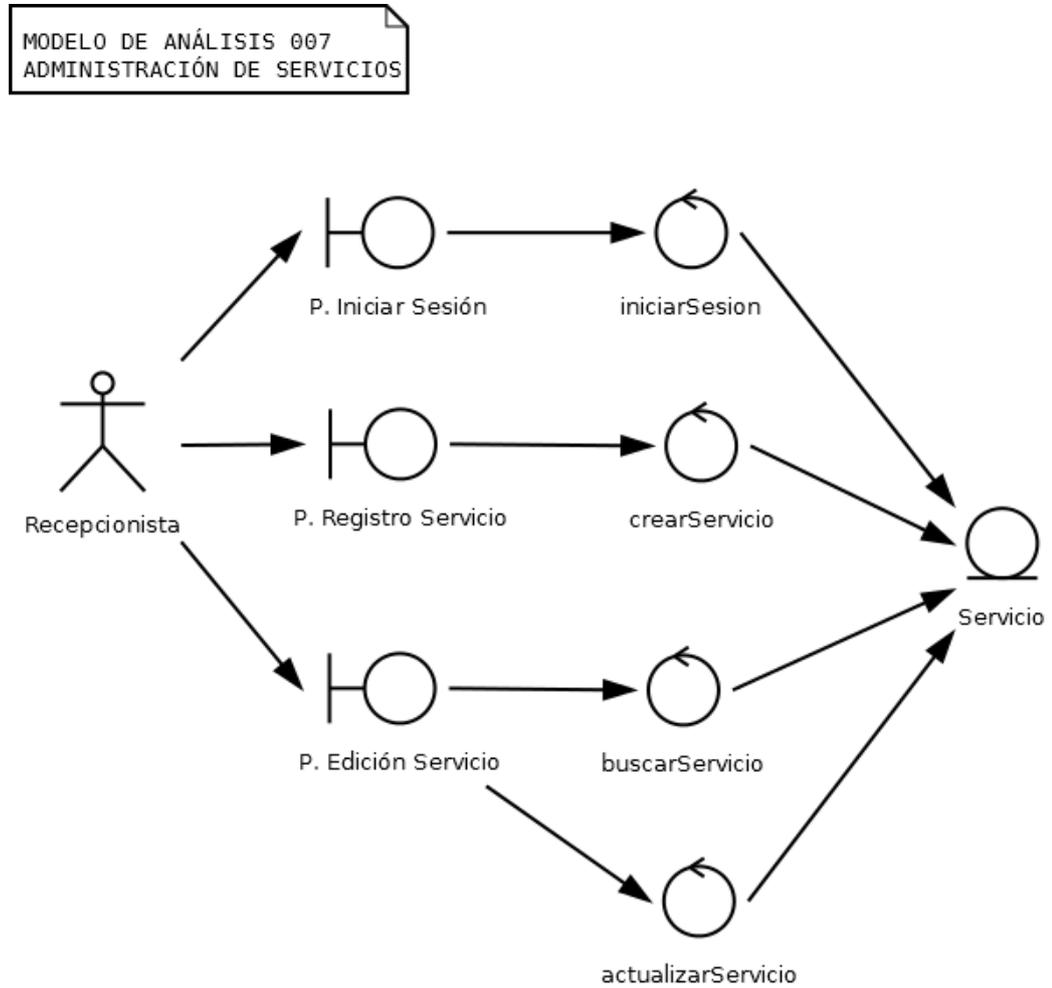


Elaborado por: Víctor Quimbiamba

La figura 26 describe el modelo de análisis del escenario de administración de tipo de material.

2.3.2.7 Administración de servicio

Figura 27. Modelo de Análisis: Administración de Servicio

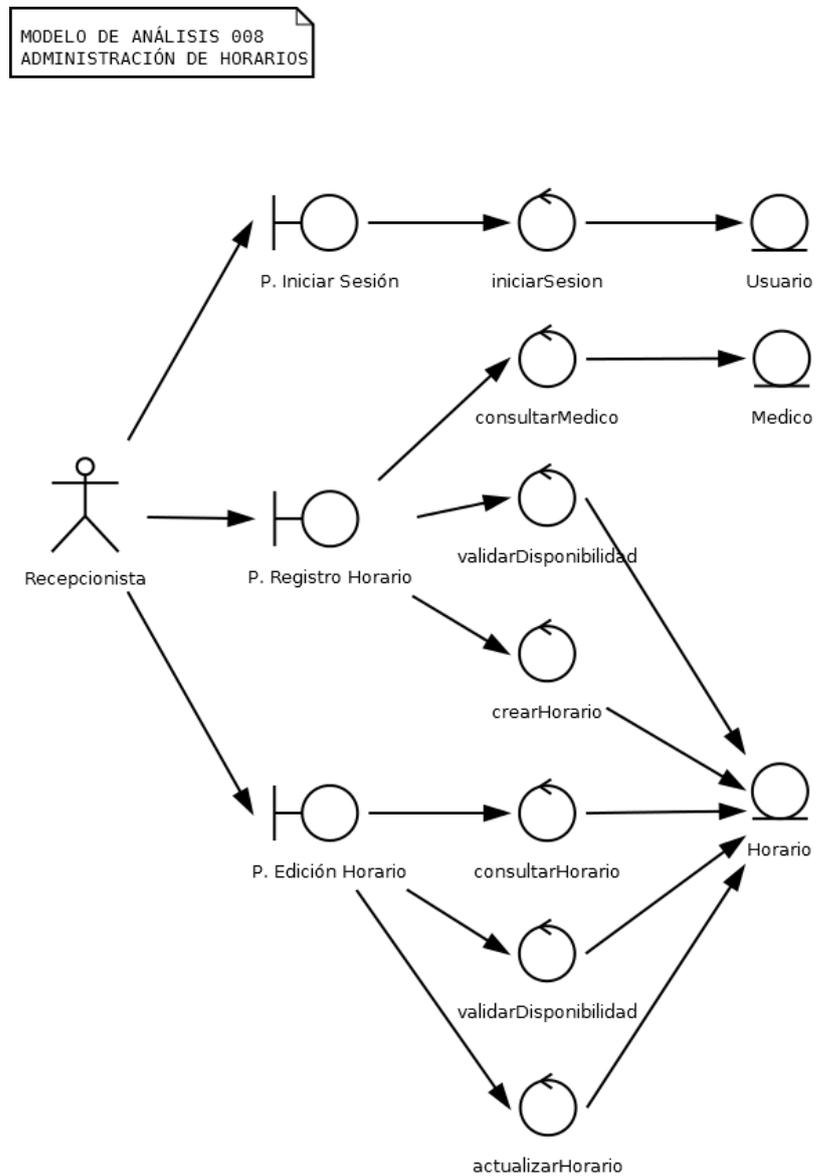


Elaborado por: Víctor Quimbiamba

La figura 27 describe el modelo de análisis del escenario de administración de servicio.

2.3.2.8 Administración de horario

Figura 28. Modelo de Análisis: Administración de Horarios

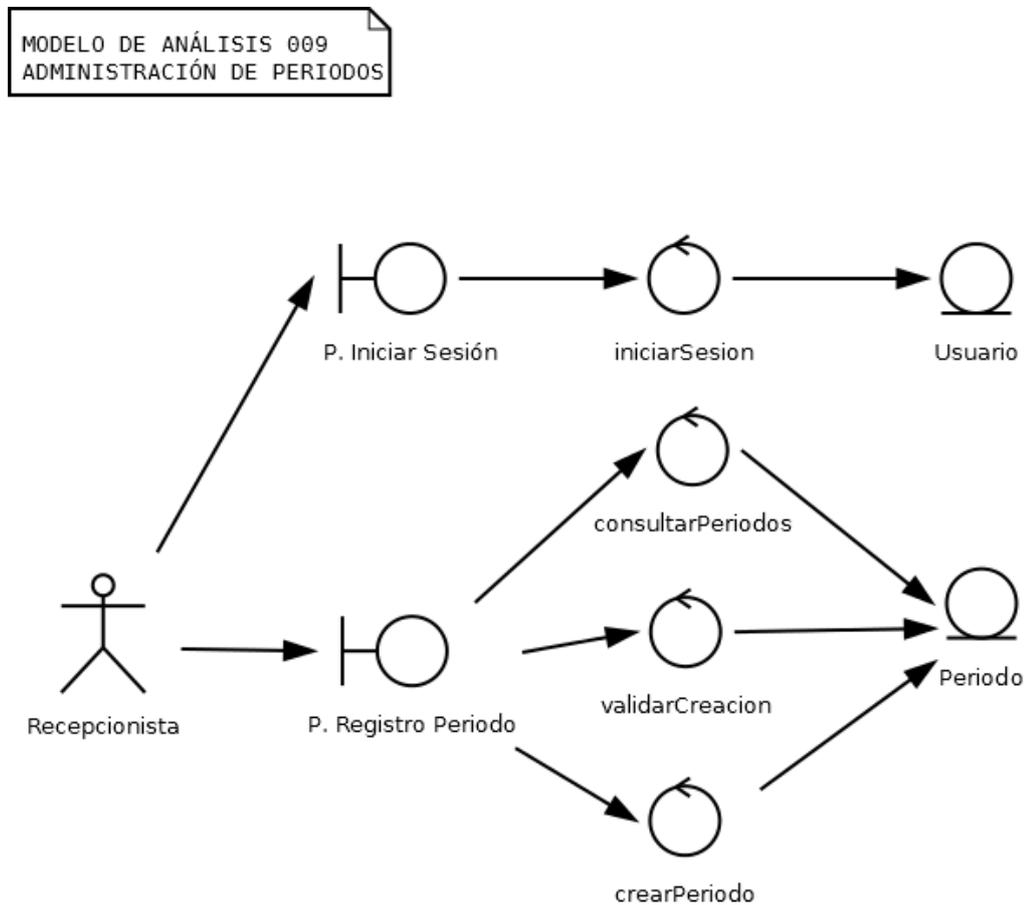


Elaborado por: Víctor Quimbiamba

La figura 28 describe el modelo de análisis del escenario de administración de horario.

2.3.2.9 Administración de periodo

Figura 29. Modelo de Análisis: Administración de Periodos



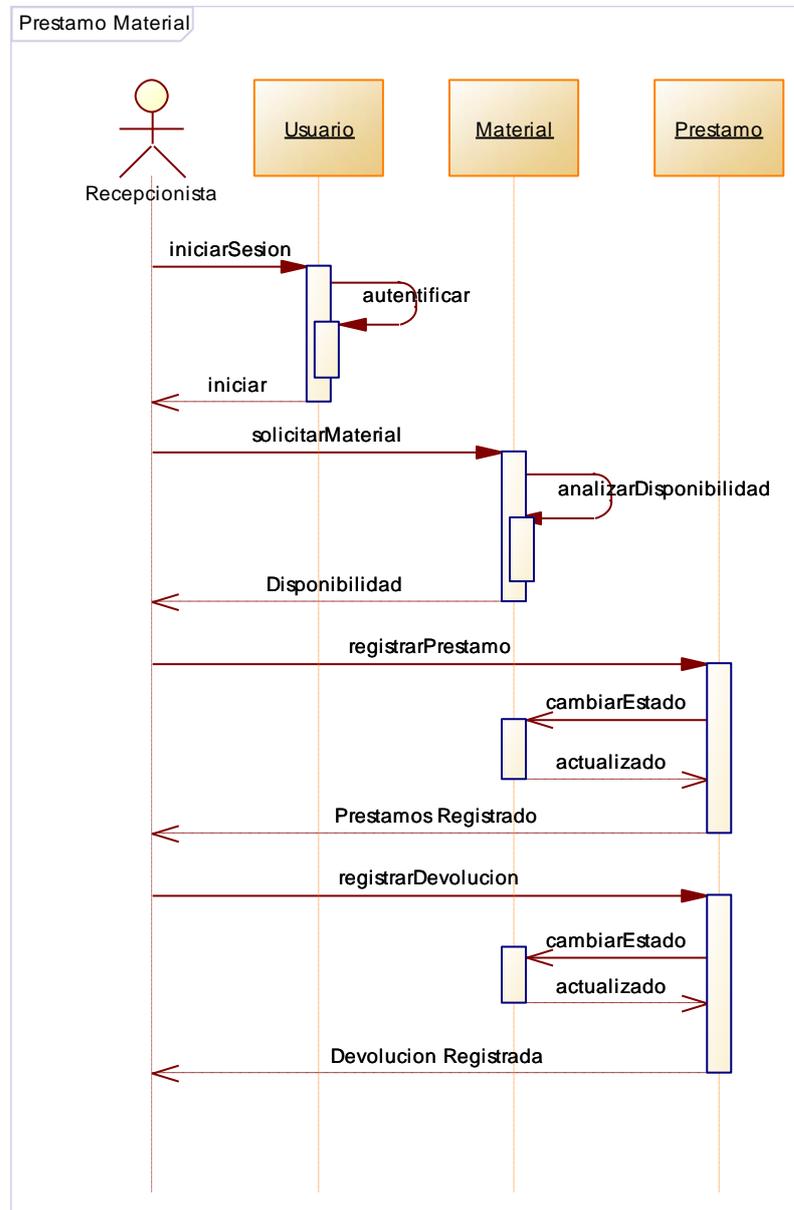
Elaborado por: Víctor Quimbiamba

La figura 29 describe el modelo de análisis del escenario de administración de periodo.

2.3.3 Diagrama de secuencia

2.3.3.1 Préstamo de materiales

Figura 30. Diagrama de Secuencia: Préstamo de Materiales

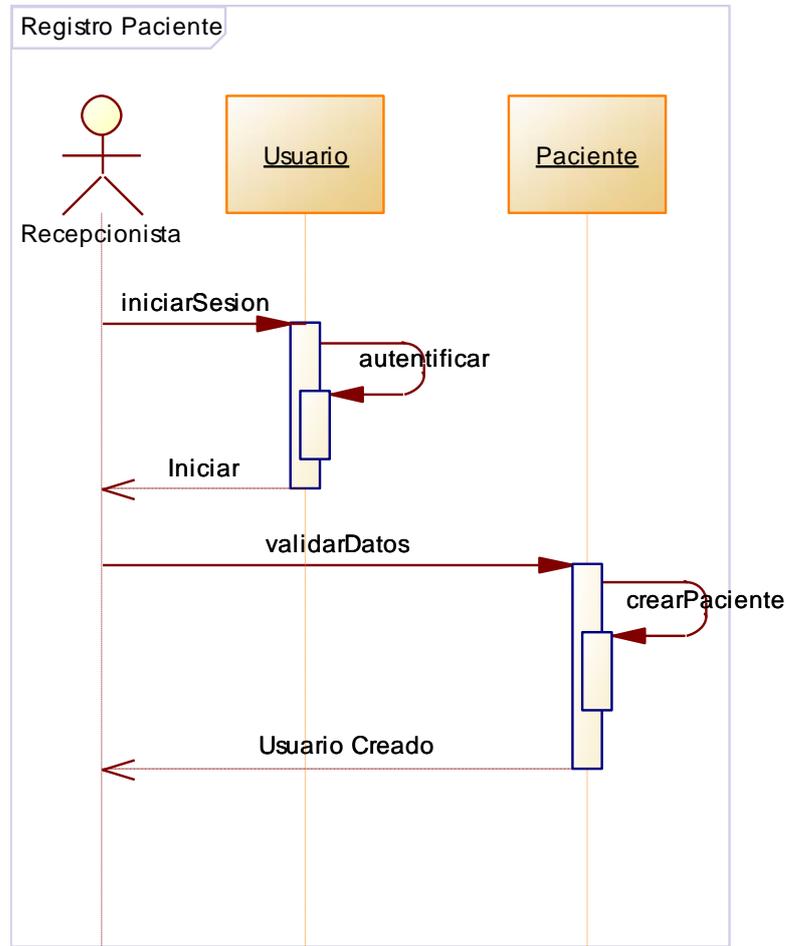


Elaborado por: Víctor Quimbiamba

La figura 30 muestra el diagrama de secuencia del escenario para el préstamo de materiales.

2.3.3.2 Registro paciente

Figura 31. Diagrama de Secuencia: Registro Paciente

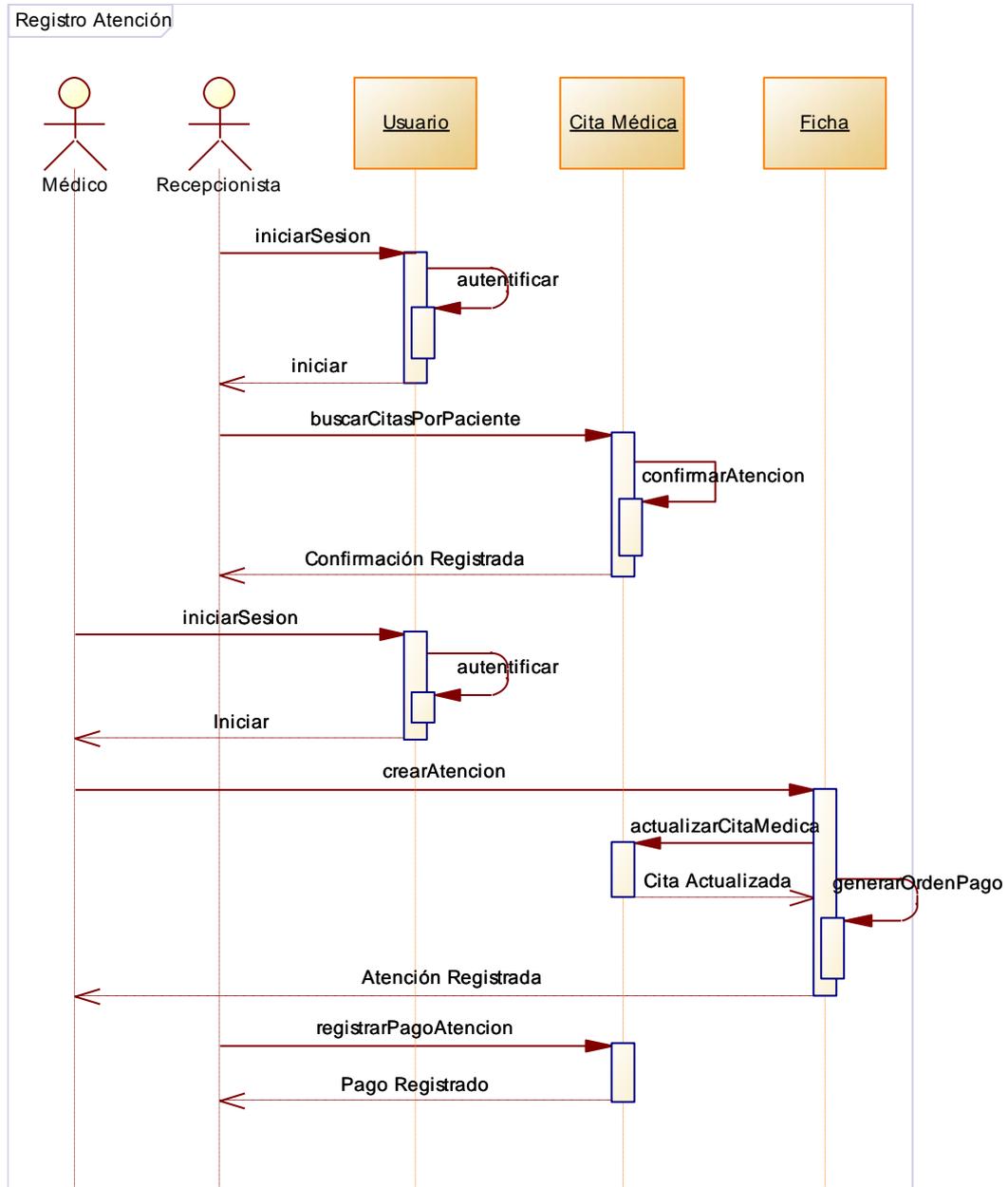


Elaborado por: Víctor Quimbiamba

La figura 31 muestra el diagrama de secuencia del escenario para el registro de paciente.

2.3.3.3 Registro atención

Figura 32. Diagrama de Secuencia: Registro Atención

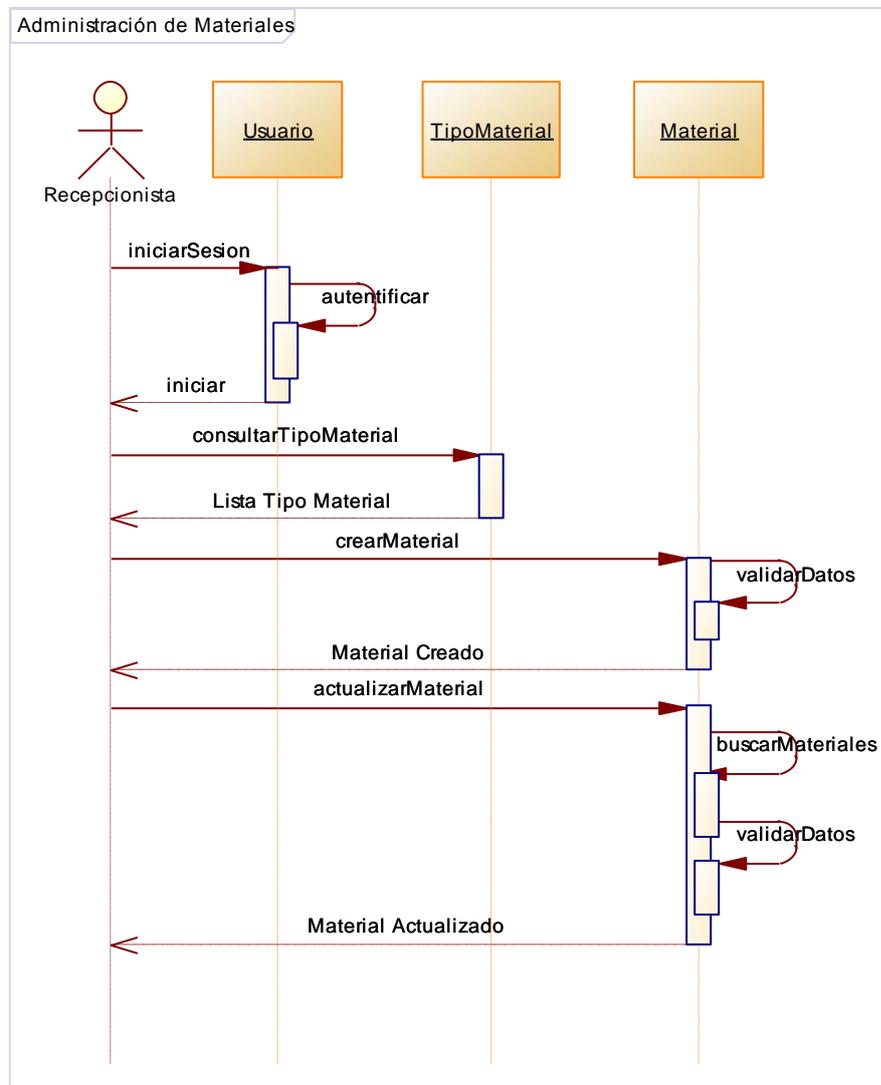


Elaborado por: Víctor Quimbiamba

La figura 32 muestra el diagrama de secuencia del escenario para el registro de atención.

2.3.3.4 Administración de material

Figura 2.22. Diagrama de Secuencia: Administración de Material

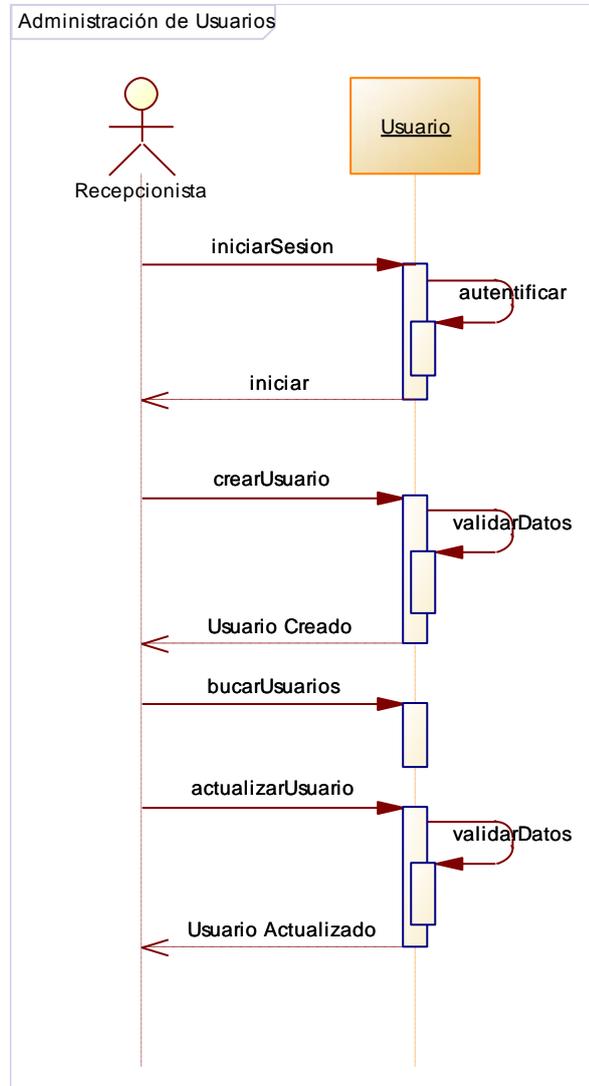


Elaborado por: Víctor Quimbiamba

La figura 33 muestra el diagrama de secuencia del escenario para la administración de material.

2.3.3.5 Administración de usuarios

Figura 2.23. Diagrama de Secuencia: Administración de Usuarios

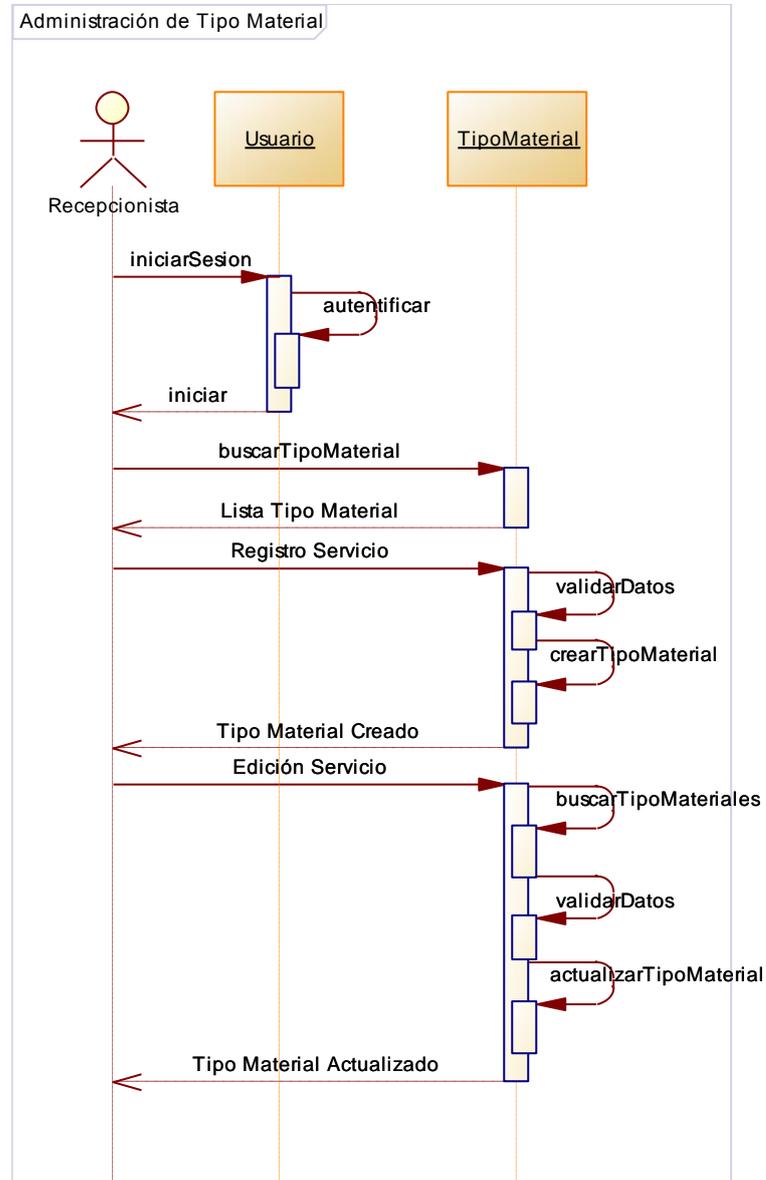


Elaborado por: Víctor Quimbiamba

La figura 34 muestra el diagrama de secuencia del escenario para la administración de usuarios.

2.3.3.6 Administración de tipo material

Figura 35. Diagrama de Secuencia: Administración Tipo Material

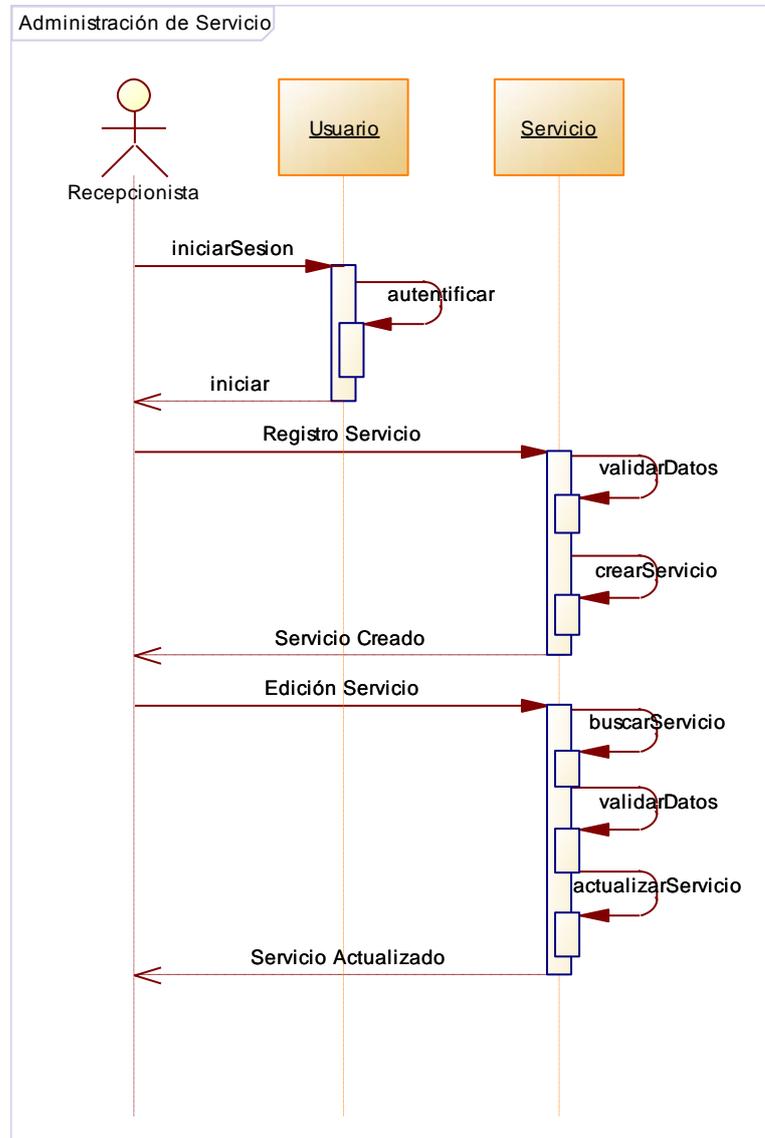


Elaborado por: Víctor Quimbiamba

La figura 35 muestra el diagrama de secuencia del escenario para la administración de tipo material.

2.3.3.7 Administración de servicio

Figura 36. Diagrama de Secuencia: Administración de Servicio

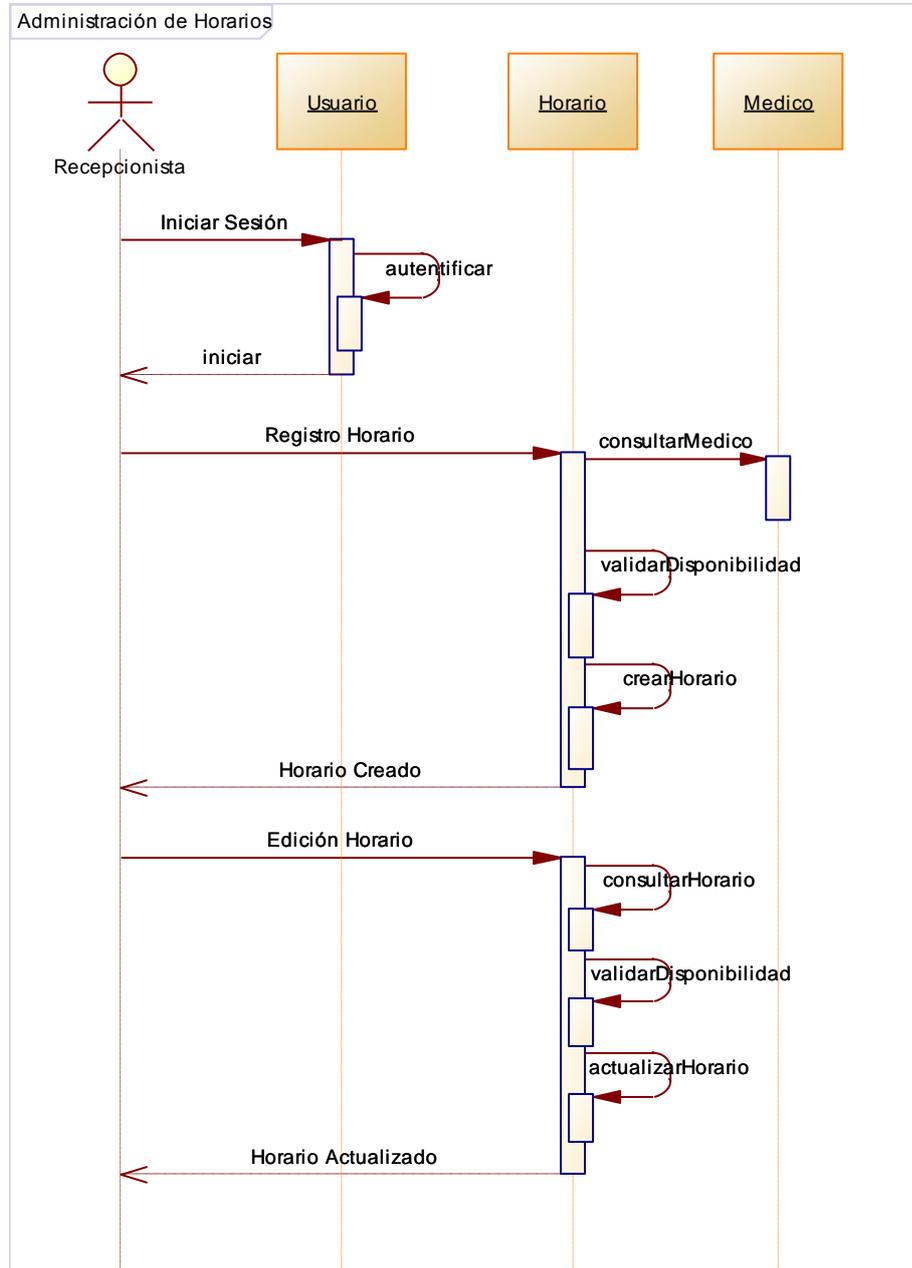


Elaborado por: Víctor Quimbiamba

La figura 36 muestra el diagrama de secuencia del escenario para la administración de servicio.

2.3.3.8 Administración de horarios

Figura 37. Diagrama de Secuencia: Administración de Horarios

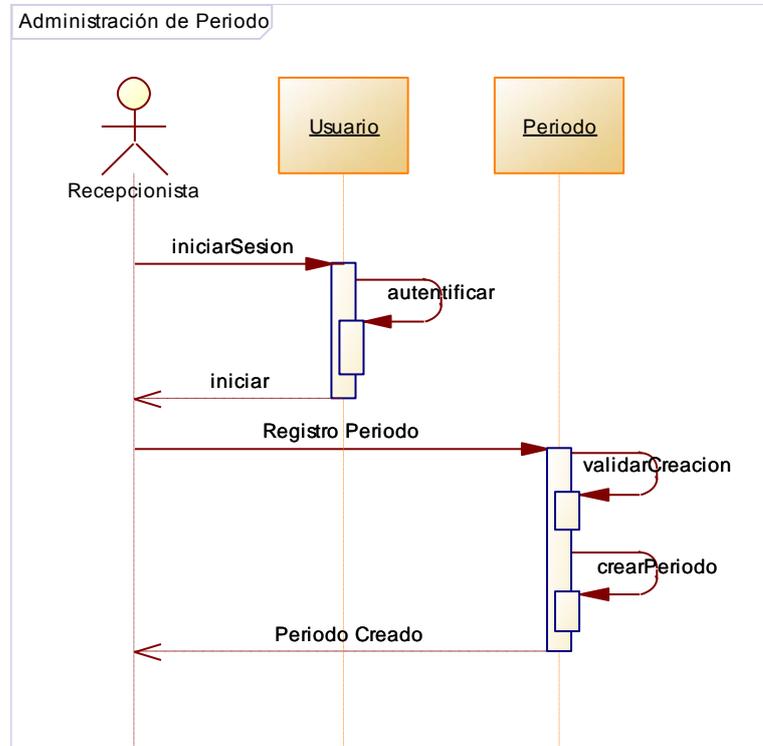


Elaborado por: Víctor Quimbiamba

La figura 37 muestra el diagrama de secuencia del escenario para la administración de horario.

2.3.3.9 Administración de periodo

Figura 38. Diagrama de Secuencia: Administración de Periodos

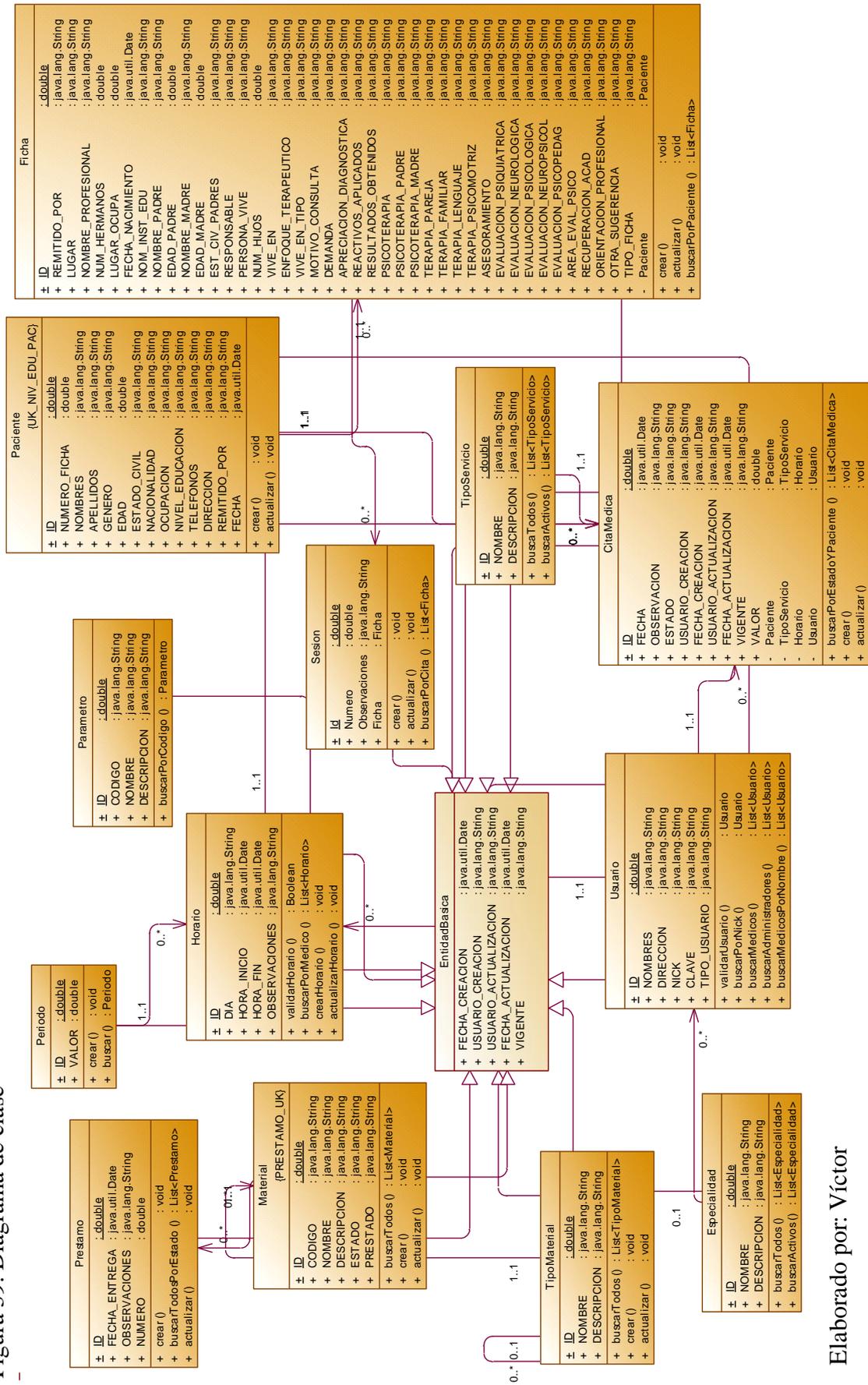


Elaborado por: Víctor Quimbiamba

La figura 38 muestra el diagrama de secuencia del escenario para la administración de periodo.

2.3.4 Diagrama de clases

Figura 39. Diagrama de clase

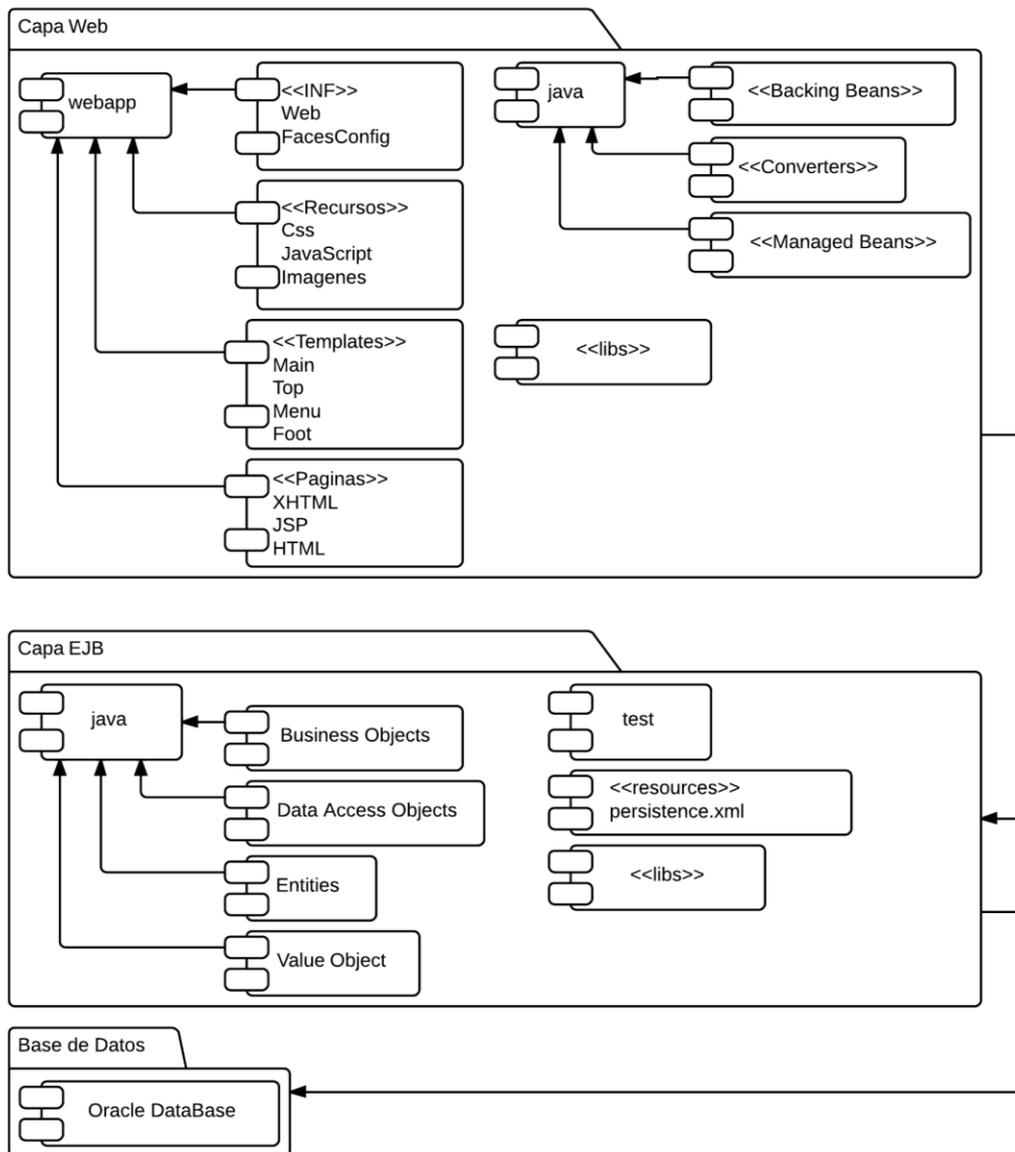


Elaborado por: Víctor

2.3.5 Diagrama de componentes

La descripción de la arquitectura del sistema se puede observar en la figura 2.29 en la cual se describen los paquetes, capas y organización.

Figura 40. Diagrama de Componentes



Elaborado por: Víctor Quimbiamba

CAPÍTULO 3

DISEÑO E IMPLEMENTACIÓN

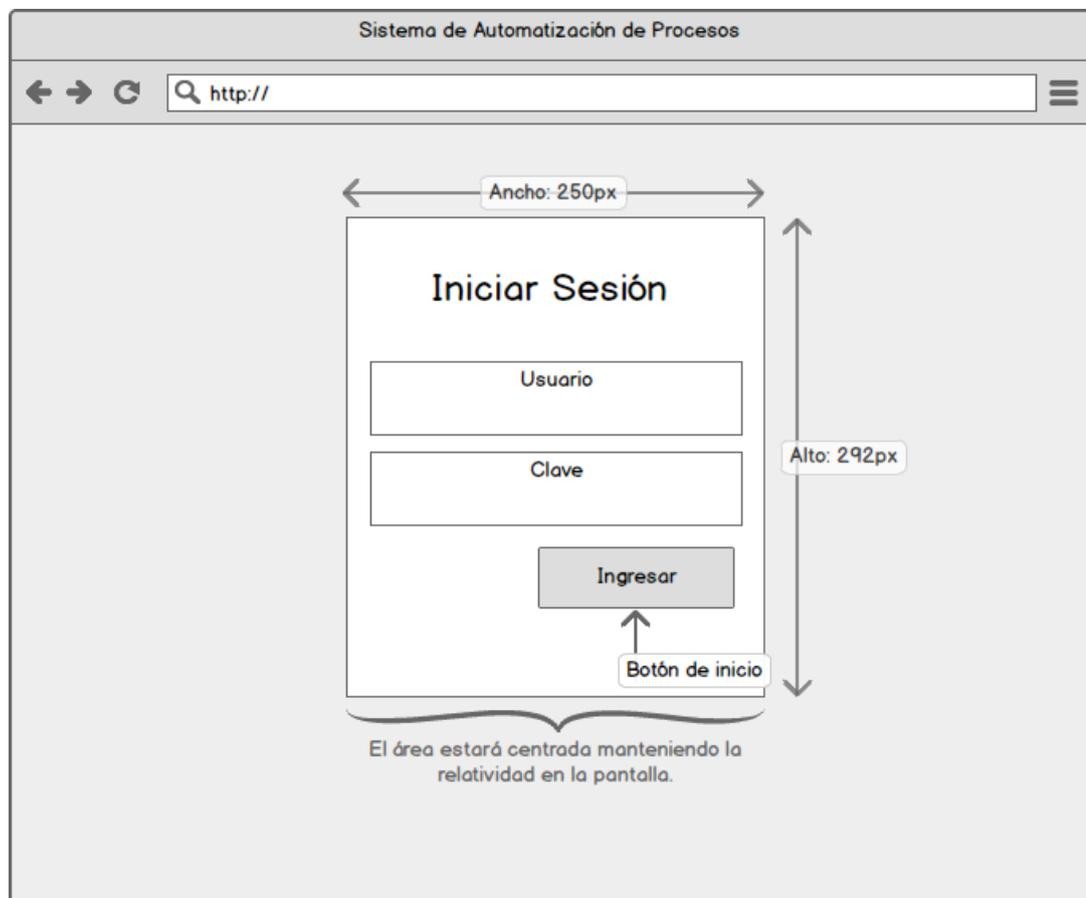
3.1 Diseño de interfaces

La siguiente es una presentación del estándar en las distintas interfaces que se presentan en el sistema utilizando la herramienta de diseño de interfaces Balsamiq Mockups.

Esta pantalla especifica la forma y dimensiones para el uso del sistema.

Inicio de sesión

Figura 41. Pantalla Inicio Sesión

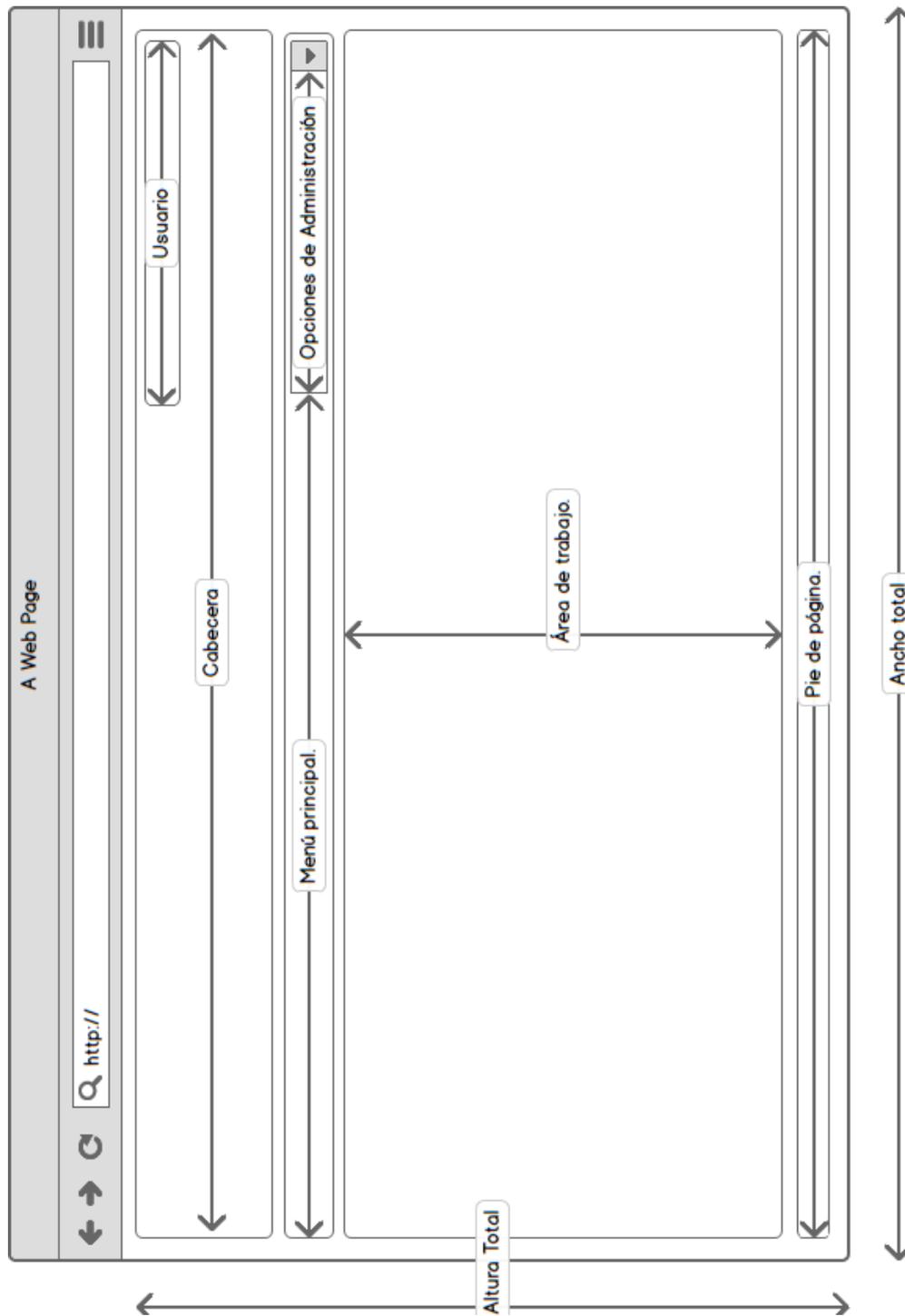


Elaborado por: Víctor Quimbiamba

Pantalla principal

La figura 42 muestra la plantilla genérica que se usa en el sistema, en determinados casos varia el área de trabajo, pero el esquema básico se mantiene.

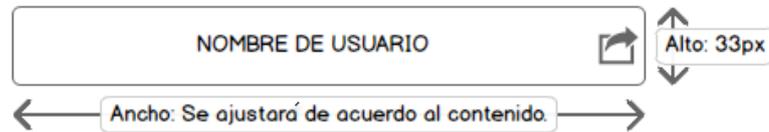
Figura 42. Plantilla principal



Elaborado por: Víctor Quimbiamba

Usuario

Figura 43. Área de usuario

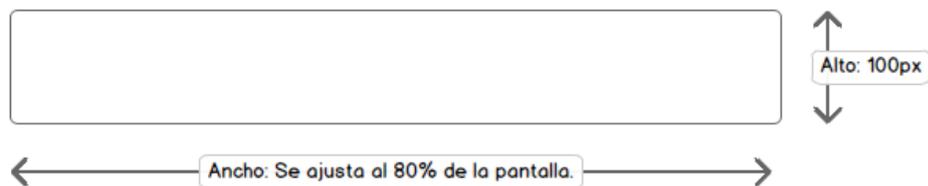


Elaborado por: Víctor Quimbiamba

La figura 43 muestra como el área está destinada para mostrar el nombre del usuario que ha ingresado al sistema, además junto al nombre se mostrará un ícono que se usará para salir del sistema.

Cabecera

Figura 44. Área de cabecera

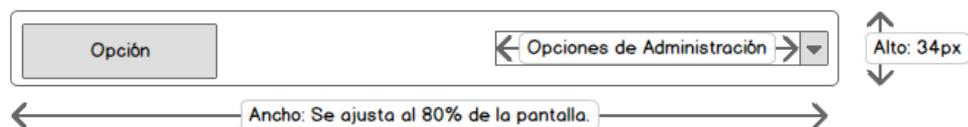


Elaborado por: Víctor Quimbiamba

La figura 44 muestra el espacio que está destinado para mostrar un diseño con el logotipo de la institución.

Menú principal

Figura 45. Área del menú principal

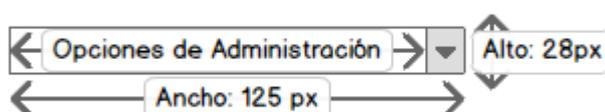


Elaborado por: Víctor Quimbiamba

La figura 45 muestra el espacio del menú principal proporciona un acceso rápido a los módulos principales del sistema de acuerdo al perfil, además aloja las opciones de administración como parte de un sub menú.

Opciones de administración

Figura 46. Área de las opciones de administración



Elaborado por: Víctor Quimbiamba

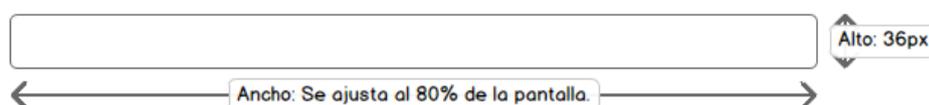
La figura 46 muestra el espacio de opciones de administración, las opciones se muestran de acuerdo al perfil.

Área de trabajo

El espacio de área de trabajo es donde se mostrarán los formularios de los diferentes módulos.

Pie de página

Figura 47. Área de pie de página



Elaborado por: Víctor Quimbiamba

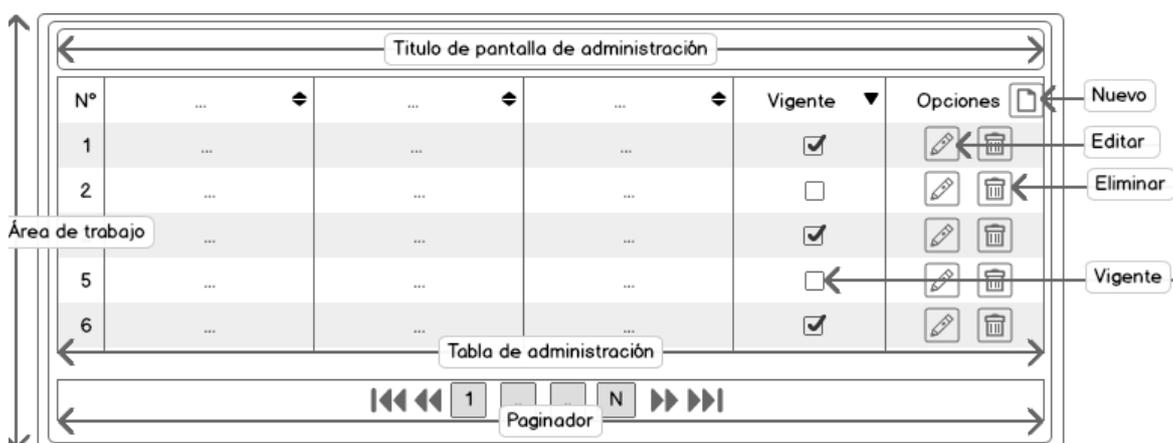
La figura 47 muestra el espacio de pie de página, es usado para mostrar información del sistema como preferencias de exploradores, dimensión recomendada, etc.

Altura total: La altura total de la pantalla es relativa, es decir el espacio se acoplará al contenido que se encuentre en el área de trabajo.

Ancho total: El ancho total de la pantalla está definido por un estándar de diseño.

Lista de administración

Figura 48. Lista de administración



Elaborado por: Víctor Quimbiamba

Descripción: La figura 48 muestra la lista en los módulos de administración.

Título de la pantalla: Muestra el título de la pantalla administrativa, ejemplo: “Administración de Materiales”, posee un ancho relativo del 80% de la pantalla y una altura de 25 pixeles.

Nuevo: Es una opción que inicia la creación de una opción de administración y cambia de pantalla al formulario correspondiente.

Editar: Es una opción que se presenta en cada registro de la tabla de administración, permite editar un registro.

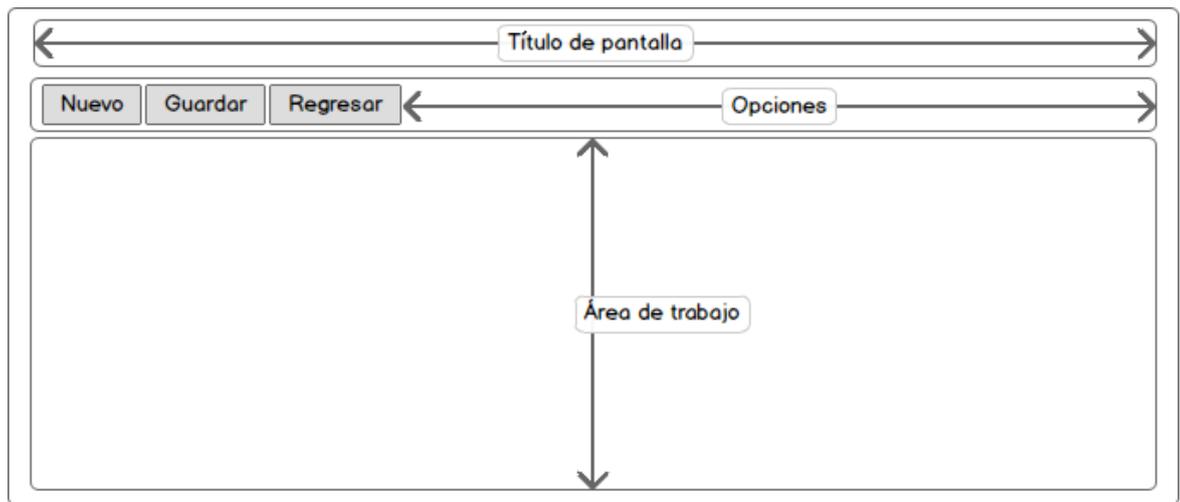
Eliminar: Es una opción que se presenta en cada registro de la tabla de administración, permite eliminar un registro.

Vigencia: Es un atributo que todo registro posee e indica el estado en el que se encuentra el registro.

Paginador: En un componente que se sitúa bajo las tablas de administración, el propósito del paginador es el de dar una acceso más cómodo a las tablas pues divide la información en páginas y permite navegar de forma rápida.

Pantalla de proceso

Figura 49. Pantalla de procesos



Elaborado por: Víctor Quimbiamba

Descripción: La figura 49 describe la estructura de las pantallas de procesos el contenido, las opciones y el título variarán de acuerdo al proceso o administración que se realice.

Mensajes

Figura 50. Icono de error



Elaborado por: Víctor Quimbiamba

La figura 50 muestra el icono de error, se mostrará junto al campo de ingreso que presente una validación, las validaciones serán de tipo obligatorio, mal formato, etc.

Figura 51. Mensaje de proceso exitoso



Elaborado por: Víctor Quimbiamba

La figura 51 muestra el estilo de los mensajes de éxito, este tipo de mensajes aparecerán en el centro de la pantalla e indicará un mensaje de éxito con la acción que se haya llamado.

Figura 52. Mensaje de proceso fallido



Elaborado por: Víctor Quimbiamba

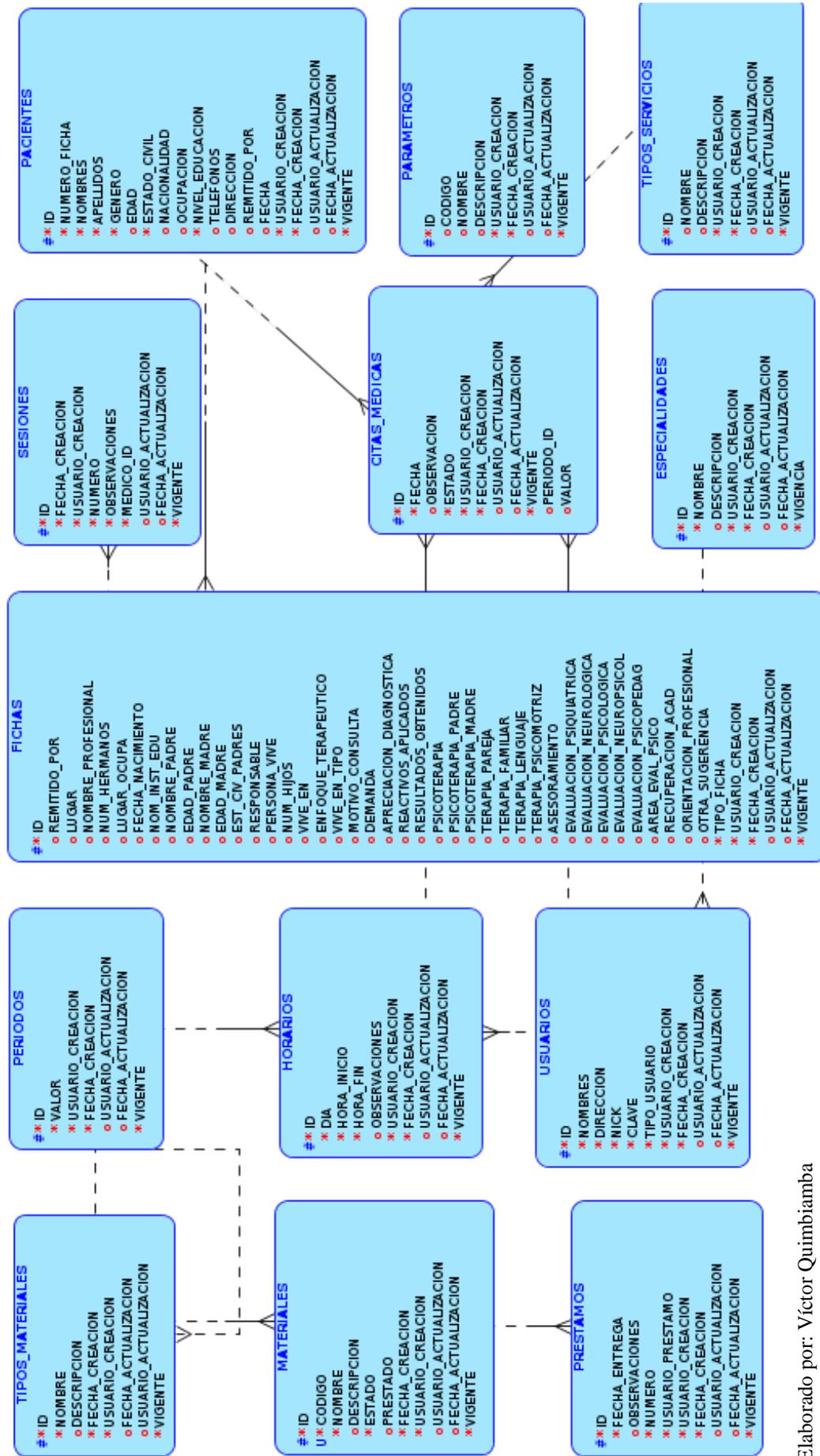
La figura 52 muestra el estilo de los mensajes de error, este tipo de mensajes aparecerán en el centro de la pantalla e indicará un mensaje de error con la acción que se haya llamado.

3.2 Diseño de base de datos

3.2.1 Modelo lógico

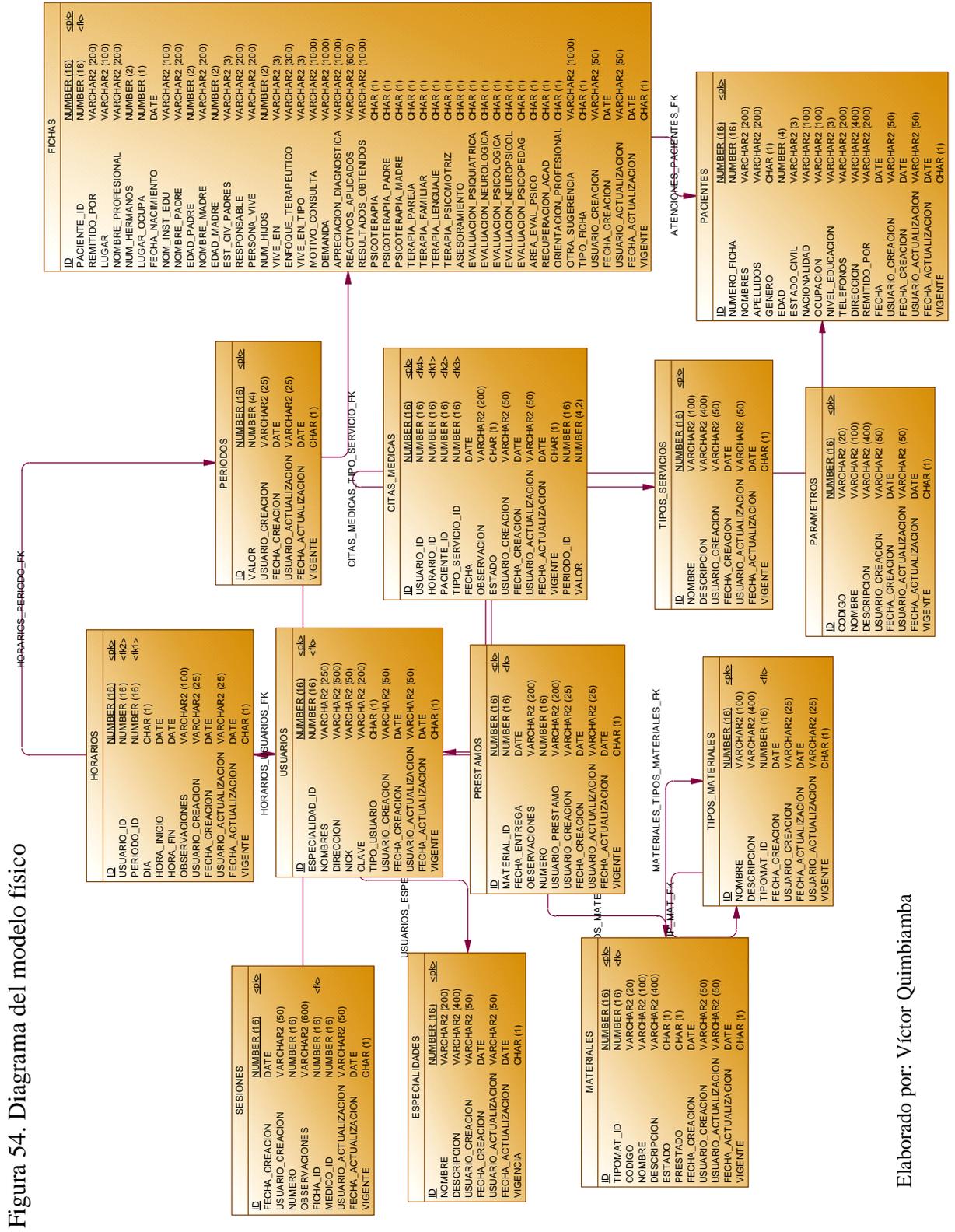
Debido a que la base de datos es Oracle Data Base se utilizó la herramienta Oracle SQL Developer Data Modeler para diseñar los diferentes modelos y el script.

Figura 53 – Modelo lógico



Elaborado por: Víctor Quimbamba

3.2.2 Modelo físico



Elaborado por: Víctor Quimbamba

3.2.3 Diccionario de datos

Tabla 11. Tabla TIPOS_MATERIALES

Campo	Tipo	Tamaño	Descripción
ID	NUMBER	16	Clave principal
NOMBRE	VARCHAR2	100	Nombre del tipo material
DESCRIPCION	VARCHAR2	400	Descripción del tipo material
TIPOMAT_ID	NUMBER	16	Clave foránea de TIPOS_MATERIALES
FECHA_CREACION	DATE		Fecha en la que el registro es creado.
USUARIO_CREACION	VARCHAR2	25	Usuario que crea el nuevo registro
FECHA_ACTUALIZACION	DATE		Fecha de la última edición del registro.
USUARIO_ACTUALIZACION	VARCHAR2	25	Usuario que realiza la última edición del registro.
VIGENTE	CHAR	1	Carácter que indica el estado de vigencia del registro.

Elaborado por: Víctor Quimbiamba

Relaciones:

TIPOS_MATERIALES con el campo TIPOMAT_ID

Campos claves:

ID, TIPOMAT_ID

Tabla 12. Tabla MATERIALES

Campo	Tipo	Tamaño	Descripción
ID	NUMBER	16	Clave principal
NOMBRE	VARCHAR2	100	Nombre del material
DESCRIPCION	VARCHAR2	400	Descripción del material
TIPOMAT_ID	NUMBER	16	Clave foránea de TIPOS_MATERIALES
CODIGO	VARCHAR2	20	Código único para el material
ESTADO	VARCHAR2	1	Describe el estado en que el material se encuentra el material.
PRESTADO	VARCHAR2	1	Carácter que indica si el materias está siendo prestado.
FECHA_CREACION	DATE		Fecha en la que el registro es creado.
USUARIO_CREACION	VARCHAR2	25	Usuario que crea el nuevo registro
FECHA_ACTUALIZACION	DATE		Fecha de la última edición del registro.
USUARIO_ACTUALIZACION	VARCHAR2	25	Usuario que realiza la última edición del registro.
VIGENTE	CHAR	1	Carácter que indica el estado de vigencia del registro.

Elaborado por: Víctor Quimbiamba

Relaciones:

TIPOS_MATERIALES con el campo TIPOMAT_ID

Campos claves:

ID, TIPOMAT_ID

Tabla 13. Tabla PRESTAMOS

Campo	Tipo	Tamaño	Descripción
ID	NUMBER	16	Clave principal
MATERIAL_ID	NUMBER	16	Clave foránea de la tabla MATERIALES
FECHA_ENTREGA	DATE		Fecha prevista para la entrega del material
OBSERVACIONES	VARCHAR2		Información adicional durante el proceso de préstamo.
NUMERO	NUMBER		Número secuencial para el registro del préstamo
USUARIO_PRESTAMO	VARCHAR2		Usuario al cual se le entrega el material.
FECHA_CREACION	DATE		Fecha en la que el registro es creado.
USUARIO_CREACION	VARCHAR2	25	Usuario que crea el nuevo registro
FECHA_ACTUALIZACION	DATE		Fecha de la última edición del registro.
USUARIO_ACTUALIZACION	VARCHAR2	25	Usuario que realiza la última edición del registro.
VIGENTE	CHAR	1	Carácter que indica el estado de vigencia del registro.

Elaborado por: Víctor Quimbiamba

Relaciones:

MATERIALES con el campo MATERIAL_ID

Campos claves:

ID, MATERIAL_ID

Tabla 14. Tabla PERIODOS

Campo	Tipo	Tamaño	Descripción
ID	NUMBER	16	Clave principal
VALOR	NUMBER	4	Almacena el valor numérico del periodo vigente
FECHA_CREACION	DATE		Fecha en la que el registro es creado.
USUARIO_CREACION	VARCHAR2	25	Usuario que crea el nuevo registro
FECHA_ACTUALIZACION	DATE		Fecha de la última edición del registro.
USUARIO_ACTUALIZACION	VARCHAR2	25	Usuario que realiza la última edición del registro.
VIGENTE	CHAR	1	Carácter que indica el estado de vigencia del registro.

Elaborado por: Víctor Quimbiamba

Campos claves:

ID

Tabla 15. Tabla USUARIOS

Campo	Tipo	Tamaño	Descripción
ID	NUMBER	16	Clave principal
ESPECIALIDAD_ID	NUMBER	16	Clave foránea de ESPECIALIDADES
NOMBRES	VARCHAR2	250	Nombre correspondiente al usuario.
DIRECCION	VARCHAR2	500	Dirección del usuario
NICK	VARCHAR2	50	Nombre único para el ingreso al sistema.
CLAVE	VARCHAR2	200	Clave para el ingreso al sistema.
TIPO_USUARIO	VARCHAR2	1	Identificador para los dos tipos de roles M=Médico, A=Administrador.
FECHA_CREACION	DATE		Fecha en la que el registro es creado.
USUARIO_CREACION	VARCHAR2	25	Usuario que crea el nuevo registro
FECHA_ACTUALIZACION	DATE		Fecha de la última edición del registro.
USUARIO_ACTUALIZACION	VARCHAR2	25	Usuario que realiza la última edición del registro.
VIGENTE	CHAR	1	Carácter que indica el estado de vigencia del registro.

Elaborado por: Víctor Quimbiamba

Relaciones:

ESPECIALIDADES con el campo ESPECIALIDAD_ID

Campos claves:

ID, ESPECIALIDAD_ID

Tabla 16. Tabla ESPECIALIDADES

Campo	Tipo	Tamaño	Descripción
ID	NUMBER	16	Clave principal
NOMBRES	VARCHAR2	200	Nombre de la especialidad.
DESCRIPCIÓN	VARCHAR2	400	Descripción de la especialidad.
FECHA_CREACION	DATE		Fecha en la que el registro es creado.
USUARIO_CREACION	VARCHAR2	25	Usuario que crea el nuevo registro
FECHA_ACTUALIZACION	DATE		Fecha de la última edición del registro.
USUARIO_ACTUALIZACION	VARCHAR2	25	Usuario que realiza la última edición del registro.
VIGENTE	CHAR	1	Carácter que indica el estado de vigencia del registro.

Elaborado por: Víctor Quimbiamba

Campos claves:

ID

Tabla 17. Tabla HORARIOS

Campo	Tipo	Tamaño	Descripción
ID	NUMBER	16	Clave principal
USUARIO_ID	NUMBER	16	Clave foránea de USUARIOS
PERIODO_ID	NUMBER	16	Clave foránea de PERIODOS
DIA	VARCHAR2	1	Valor simbólico para un día de la semana
HORA_INICIO	DATE		Hora de inicio del horario
HORA_FIN	DATE		Hora fin del horario
OBSERVACIONES	VARCHAR2	100	Observación adicional durante la creación.
FECHA_CREACION	DATE		Fecha en la que el registro es creado.
USUARIO_CREACION	VARCHAR2	25	Usuario que crea el nuevo registro
FECHA_ACTUALIZACION	DATE		Fecha de la última edición del registro.
USUARIO_ACTUALIZACION	VARCHAR2	25	Usuario que realiza la última edición del registro.
VIGENTE	CHAR	1	Carácter que indica el estado de vigencia del registro.

Elaborado por: Víctor Quimbiamba

Relaciones:

USUARIOS con el campo USUARIO_ID

PERIODOS con el campo PERIODO_ID

Campos claves:

ID, USUARIO_ID, PERIODO_ID

Tabla 18. Tabla TIPOS_SERVICIOS

Campo	Tipo	Tamaño	Descripción
ID	NUMBER	16	Clave principal
NOMBRE	VARCHAR2	100	Nombre del tipo de servicio
DESCRIPCION	VARCHAR2	400	Descripción del tipo de servicio
FECHA_CREACION	DATE		Fecha en la que el registro es creado.
USUARIO_CREACION	VARCHAR2	25	Usuario que crea el nuevo registro
FECHA_ACTUALIZACION	DATE		Fecha de la última edición del registro.
USUARIO_ACTUALIZACION	VARCHAR2	25	Usuario que realiza la última edición del registro.
VIGENTE	CHAR	1	Carácter que indica el estado de vigencia del registro.

Elaborado por: Víctor Quimbiamba

Campos claves:

ID

Tabla 19. Tabla PACIENTES

Campo	Tipo	Tamaño	Descripción
ID	NUMBER	16	Clave principal
NUMERO_FICHA	NUMBER	16	Número secuencia de identificación del cliente.
NOMBRES	VARCHAR2	200	Nombres del paciente
APELLIDOS	VARCHAR2	200	Apellidos del paciente
GENERO	VARCHAR2	1	Clave del genero del paciente
EDAD		4	Edad del paciente
ESTADO_CIVIL	VARCHAR2	3	Clave del estado civil del paciente
NACIONALIDAD	VARCHAR2	100	Nacionalidad del paciente
OCUPACION	VARCHAR2	100	Ocupación del paciente
NIVEL_EDUCACION	VARCHAR2	3	Nivel de educación del paciente
TELEFONOS	VARCHAR2	200	Teléfonos del paciente
DIRECCION	VARCHAR2	400	Dirección del paciente
REMITIDO_POR	VARCHAR2	200	Médico que lo remite.
FECHA	DATE		Fecha del sistema.
FECHA_CREACION	DATE		Fecha en la que el registro es creado.
USUARIO_CREACION	VARCHAR2	25	Usuario que crea el nuevo registro
FECHA_ACTUALIZACION	DATE		Fecha de la última edición del registro.
USUARIO_ACTUALIZACION	VARCHAR2	25	Usuario que realiza la última edición del registro.
VIGENTE	CHAR	1	Carácter que indica el estado de vigencia del registro.

Elaborado por: Víctor Quimbiamba

Campos claves:

ID

Tabla 20. Tabla CITAS_MEDICAS

Campo	Tipo	Tamaño	Descripción
ID	NUMBER	16	Clave principal
USUARIO_ID	NUMBER	16	Usuario medico al que se le asigna la cita
HORARIO_ID	NUMBER	16	Hora en la que la cita será programada
PACIENTE_ID	NUMBER	16	Paciente que solicita la cita médica.
TIPO_SERVICIO_ID	NUMBER	16	Tipo de servicio para la cita médica.
PERIODO_ID	NUMBER	16	Periodo en el cual se registra la cita médica.
FECHA	DATE		Fecha en la que la cita será programada.
OBSERVACION	VARCHAR2	200	Observación adicional de la cita medica
ESTADO	VARCHAR2	1	Estado de la cita médica.
VALOR	NUMBER	4,2	Valor de la cita médica.
FECHA_CREACION	DATE		Fecha en la que el registro es creado.
USUARIO_CREACION	VARCHAR2	25	Usuario que crea el nuevo registro
FECHA_ACTUALIZACION	DATE		Fecha de la última edición del registro.
USUARIO_ACTUALIZACION	VARCHAR2	25	Usuario que realiza la última edición del registro.
VIGENTE	CHAR	1	Carácter que indica el estado de vigencia del registro.

Elaborado por: Víctor Quimbiamba

Relaciones:

USUARIOS con el campo USUARIO_ID

HORARIOS con el campo HORARIO_ID

PACIENTES con el campo PACIENTE_ID

TIPOS_SERVICIOS con el campo TIPO_SERVICIO_ID

PERIODOS con el campo PERIODO_ID

Campos claves:

ID, USUARIO_ID, HORARIO_ID, PACIENTE_ID, TIPO_SERVICIO_ID, PERIODO_ID

Tabla 21. Tabla FICHA

Campo	Tipo	Tamaño	Descripción
ID	NUMBER	16	Clave principal
PACIENTE_ID	NUMBER	16	Clave foránea de PACIENTE
REMITIDO_POR	VARCHAR2	200	Descripción del remitente
LUGAR	VARCHAR2	100	Lugar de origen de la ficha
NOMBRE_PROFESIONAL	VARCHAR2	200	Nombre del médico
NUM_HERMANOS	NUMBER	2	Número de hermanos del paciente
LUGAR_OCUPA	NUMBER	1	Número de hermano
FECHA_NACIMIENTO	DATE		Fecha de nacimiento del paciente
NOM_INST_EDU	VARCHAR2	100	Institución del paciente
NOMBRE_PADRE	VARCHAR2	200	Nombre del padre del paciente
EDAD_PADRE	NUMBER	2	Edad del padre del paciente
NOMBRE_MADRE	VARCHAR2	200	Nombre del padre del paciente
EDAD_MADRE	NUMBER	2	Edad de la madre del paciente
EST_CIV_PADRES	VARCHAR2	30	Estado civil de los padres
RESPONSABLE	VARCHAR2	200	Responsable del paciente
PERSONA_VIVE	VARCHAR2	200	Perdona con la que vive el paciente
NUM_HIJOS	NUMBER	2	Número de hijos del paciente
VIVE_EN	VARCHAR2	3	Lugar en donde vive el paciente
ENFOQUE_TERAPEUTICO	VARCHAR2	300	Enfoque de la terapia
VIVE_EN_TIPO	VARCHAR2	3	Tipo de vivienda del paciente
MOTIVO_CONSULTA	VARCHAR2	1000	Motivo de la consulta
DEMANDA	VARCHAR2	1000	Dato específico de la ficha
APRECIACION_DIAGNOSTICA	VARCHAR2	1000	Dato específico de la ficha
REACTIVOS_APLICADOS	VARCHAR2	600	Dato específico de la ficha
RESULTADOS_OBTENIDOS	VARCHAR2	1000	Dato específico de la ficha
PSICOTERAPIA	CHAR	1	Dato específico de la ficha
PSICOTERAPIA_PADRE	CHAR	1	Dato específico de la ficha
PSICOTERAPIA_MADRE	CHAR	1	Dato específico de la ficha
TERAPIA_PAREJA	CHAR	1	Dato específico de la ficha
TERAPIA_FAMILIAR	CHAR	1	Dato específico de la ficha
TERAPIA LENGUAJE	CHAR	1	Dato específico de la ficha
TERAPIA PSICOMOTRIZ	CHAR	1	Dato específico de la ficha
ASESORAMIENTO	CHAR	1	Dato específico de la ficha
EVALUACION PSIQUIATRICA	CHAR	1	Dato específico de la ficha
EVALUACION NEUROLOGICA	CHAR	1	Dato específico de la ficha
EVALUACION PSICOLOGICA	CHAR	1	Dato específico de la ficha
EVALUACION NEUROPSICOL	CHAR	1	Dato específico de la ficha
EVALUACION PSICOPEDAG	CHAR	1	Dato específico de la ficha

AREA_EVAL_PSICO	CHAR	1	Dato especifico de la ficha
RECUPERACION_ACAD	CHAR	1	Dato especifico de la ficha
ORIENTACION_PROFESIONAL	CHAR	1	Dato especifico de la ficha
OTRA_SUGERENCIA	VARCHAR2	1000	Dato especifico de la ficha
TIPO_FICHA	CHAR	1	Dato especifico de la ficha
FECHA_CREACION	DATE	Date	Fecha en la que el registro es creado.
USUARIO_CREACION	VARCHAR2	25	Usuario que crea el nuevo registro
FECHA_ACTUALIZACION	DATE		Fecha de la última edición del registro.
USUARIO_ACTUALIZACION	VARCHAR2	25	Usuario que realiza la última edición del registro.
VIGENTE	VARCHAR2	1	Carácter que indica el estado de vigencia del registro.

Elaborado por: Víctor Quimbiamba

Relaciones:

PACIENTES con el campo PACIENTE_ID

Campos claves:

ID, PACIENTE_ID

3.3 Implementación

3.3.1 Manual de instalación

Requerimientos

- Software
 - Sistema Operativo, Linux o Windows.
 - Java
 - GlassFish 3.x
 - Oracle 9 o mayor
- Hardware
 - 1 PC con las siguientes características.
 - Procesador core i5 o superior
 - Memoria RAP de 4GB o superior
 - Espacio de disco superior a 100GB
 - Tarjeta de red
 - Periféricos de entrada y salida.
 - Cableado estructurado

Instalación

Sistema operativo

EL ambiente de producción debe estar instalado en un sistema operativo Linux, Windows o Mac, que soporte Java. El sistema operativo de producción para este caso será Ubuntu Server 11.10 de 64 bit. (Ver anexo 1)

Oracle

La instalación del servidor de base de datos no se especificará en este apartado pues el servidor será proporcionado por la institución.

Se agregará un anexo digital del esquema de la base de datos, para ello se crea un esquema en la base de datos Oracle y se ejecuta el script del anexo digital.

Glassfish

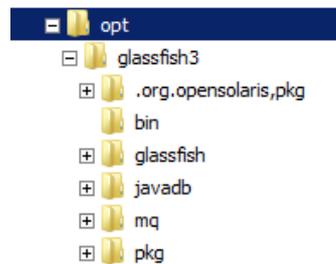
El prerequisite para la instalación del servidor de aplicaciones GlassFish es haber instalado la máquina virtual de Java. (Ver anexo 2)

Lo primero es descargar la versión 3 o superior del proyecto GlassFish desde la página de oficial (<https://glassfish.java.net/public/downloadsindex.html#top>), se recomienda descargar la versión en formato zip que es fácil de configurar y además compatible con los sistemas operativos más usados.

El archivo que se ha descargado anteriormente se debe descomprimir y copiar en un lugar sistema operativo para el caso de Linux se copia en el directorio raíz “/opt”, la dirección que se use no es importante puesto que el servicio asadmin del Glassfish será ejecutado desde el directorio en donde se ubique.

La figura 55 muestra la estructura del directorio en el servidor.

Figura 55. Directorio del Glassfish

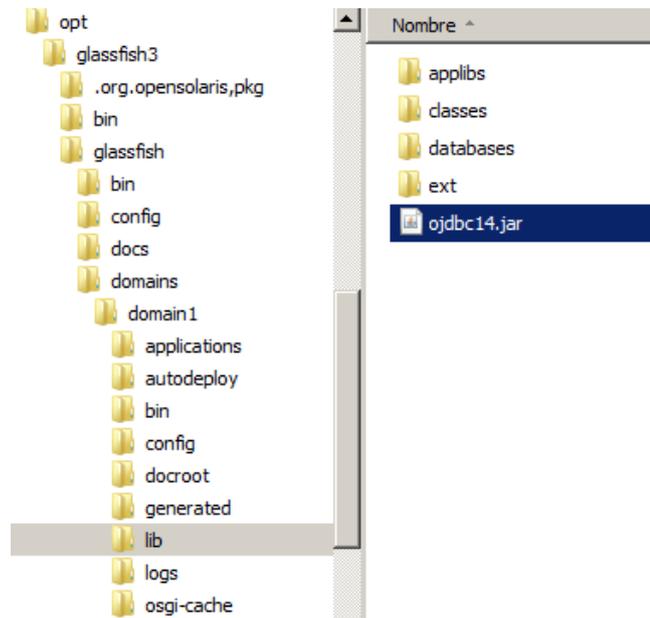


Elaborado por: Víctor Quimbiamba

El siguiente paso para el caso de Linux es darle permisos de ejecución, lectura y escritura a la carpeta del Glassfish, para esto se usa el comando “sudo chmod -R 777 /opt/glassfish3”

Las conexiones a bases de datos necesitan librerías, para este caso se necesita descargar la librería de conexión a la base de datos Oracle “ojdbc14.jar” desde la página de Oracle (<http://www.oracle.com/technetwork/database/enterprise-edition/jdbc-10201-088211.html>). Lo siguiente será copiar la librería en el directorio de la figura 56

Figura 56. Dirección de librería para el dominio 1



Elaborado por: Víctor Quimbiamba

Una vez realizados estos pasos, mediante una línea de comandos se ejecuta el comando que se ve en la figura 57

Figura 57. Comando para iniciar un dominio de Glassfish

```
ups@tesis-server:~$ /opt/glassfish3/bin/asadmin restart-domain domain1_
```

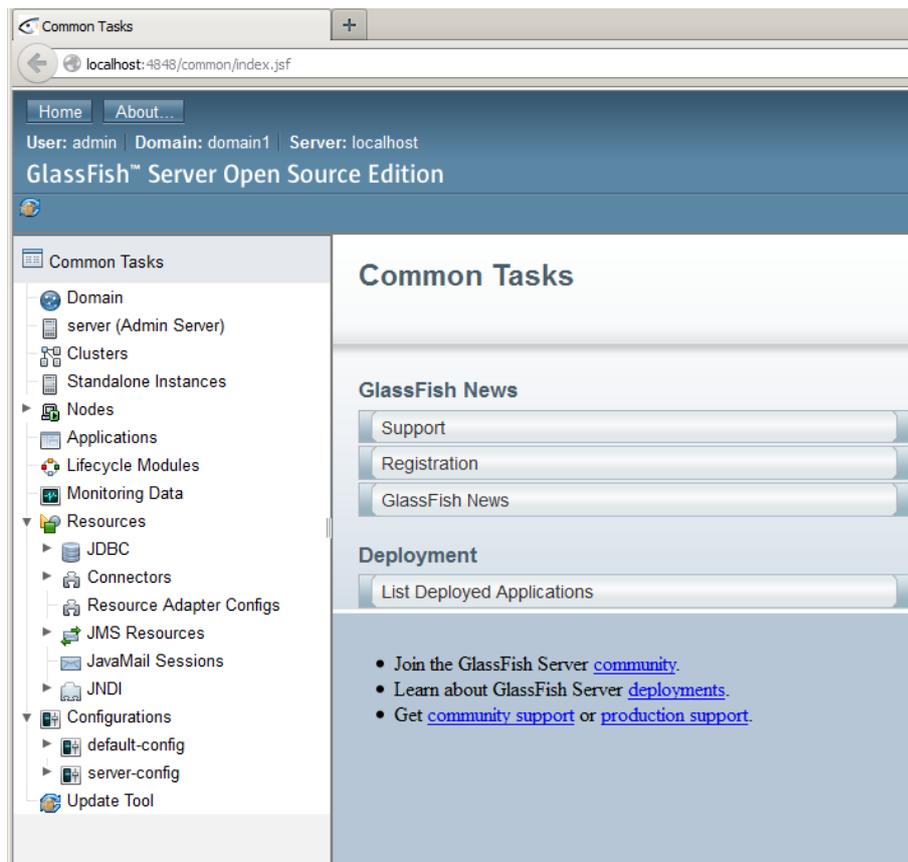
Elaborado por: Víctor Quimbiamba

Cada dominio tiene su propio Servidor de Administración de Dominios (DAS) y su propio puerto único de administración. El servidor Glassfish posee un dominio por defecto el dominio con nombre “domain1” que es el que se inició en la figura 57.

La consola de administración es una utilidad funcional que incorpora el servidor de aplicaciones Glassfish, el puerto de acceso por defecto es el 4848. La dirección URL por la cual se accederá a la consola de administración del servidor Glassfish estará compuesta por la dirección del servidor y el puerto 4848 o el puerto que se haya configurado.

El acceso a la consola de administración se puede observar en la figura 58.

Figura 58. Consola de administración de Glassfish



Elaborado por: Víctor Quimbiamba

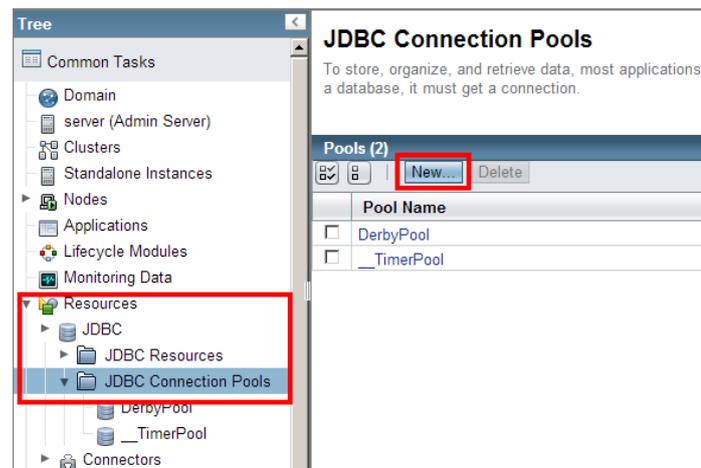
Configuración de conexión a la base de datos

La configuración de la conexión a la base tiene dos pasos, el primero es crear una nueva fuente de datos y el segundo es crear un recurso Java Database Connectivity (JDBC).

Creación de fuente de datos

La creación de fuentes de datos inicia con la selección de la opción “Resources / JDBC / JDBC Connection Pools” en el árbol de opciones de la consola de administración como se puede ver en la figura 59

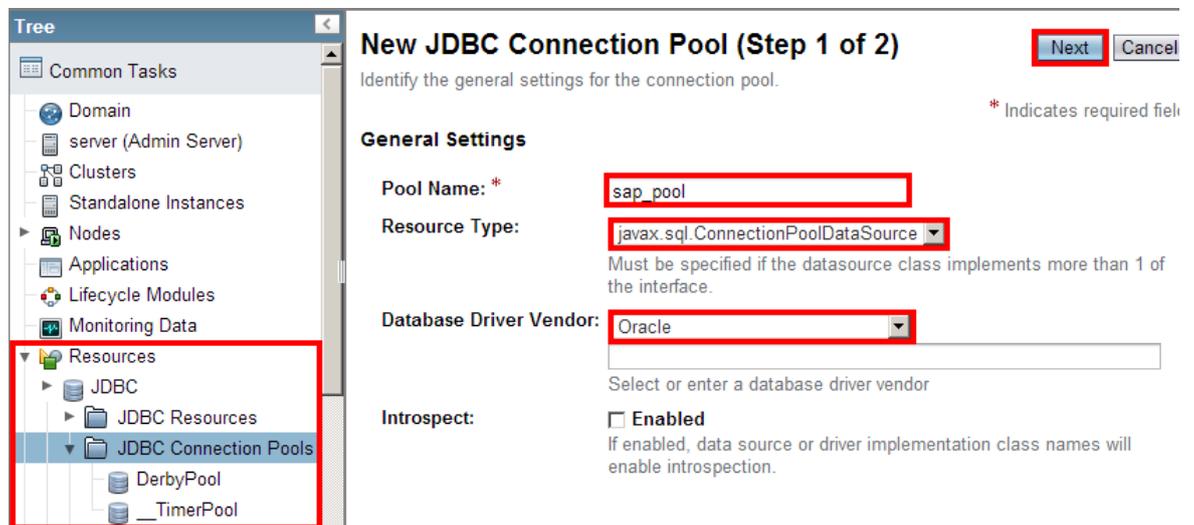
Figura 59. Creación de JDBC Connection Pools 1



Elaborado por: Víctor Quimbiamba

Una vez se presione el botón de nuevo se presentará el formulario con los datos básicos para la creación de un pool de conexiones como se puede ver en la figura 59.

Figura 60. Creación de JDBC Connection Pools 2



Elaborado por: Víctor Quimbiamba

Se debe ingresar un nombre cualquiera y seleccionar los datos correspondientes para la conexión a la base de datos Oracle, el tipo de recurso y el proveedor de manejo a la base de datos deben ser iguales a los seleccionados en la figura 60 y presionar el botón siguiente.

La librería de conexión que se utilizó en el apartado anterior proporciona al servidor Glassfish nuevas propiedades de conexión, para la conexión a la base de datos se requieren pocos datos los cuales se pueden ver en la figura 61

Figura 61. Propiedades de conexión

Name	Value	Description:
User		
Password		
DataSourceName	OracleDataSource	
ExplicitCachingEnabled	false	
URL	jdbc:oracle:thin:@IPSERVIDOR:PUERTO:SID	

Elaborado por: Víctor Quimbiamba

Los datos de la figura 61 serán proporcionados por el administrador de base de datos en donde se creó la base de datos, una vez ingresados los datos se presiona el botón de finalizar y se mostrará el nuevo pool en la tabla de pools como se puede ver en la figura 62

Figura 62. Tabla de pools de conexiones 1

Pool Name	Resource Type	Classname	Description
DerbyPool	javax.sql.DataSource	org.apache.derby.jdbc.ClientDataSource	
TimerPool	javax.sql.XADataSource	org.apache.derby.jdbc.EmbeddedXADataSource	
sap_pool	javax.sql.ConnectionPoolDataSource	oracle.jdbc.pool.OracleConnectionPoolDataSource	

Elaborado por: Víctor Quimbiamba

Una vez creado el pool de conexión su puede verificar si la conexión es exitosa presionando sobre el nombre del pool de la figura 62 y se presentará la edición del pool como se puede ver en la figura 63, para probar la conexión se presiona el botón “Ping” y se presentará un mensaje satisfactorio si la conexión se realiza con éxito caso contrario un mensaje de error, en cuyo caso se revisa la dirección el puerto y el Sistema de Identificación de Oracle u Oracle System ID (SID).

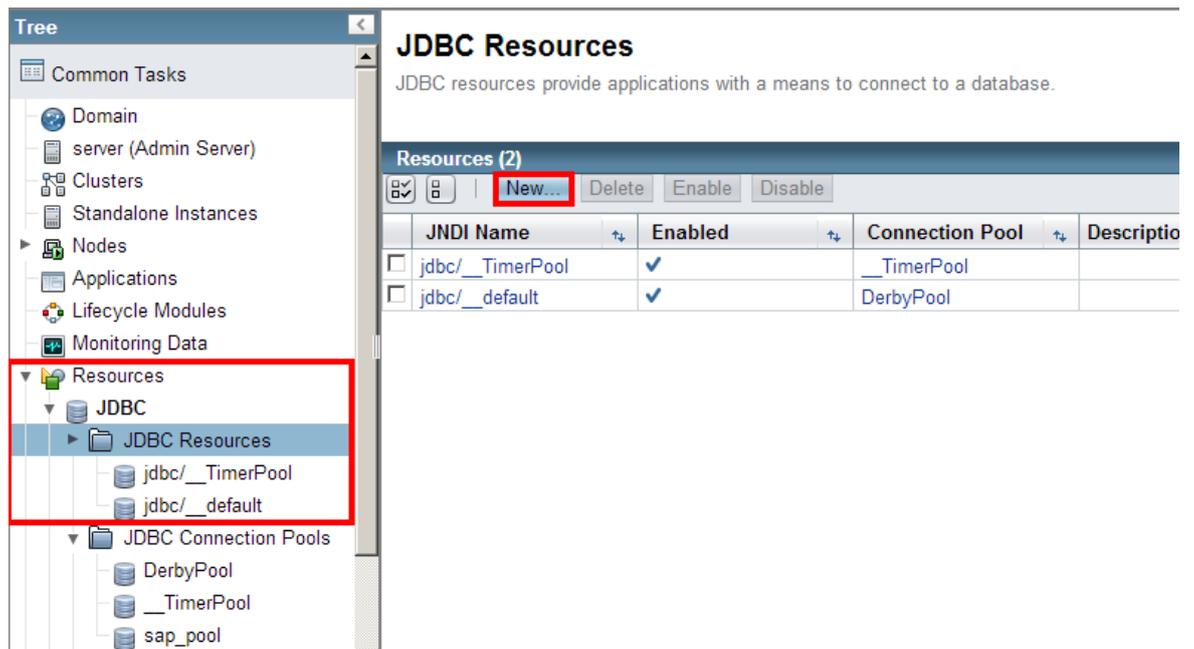
Figura 63 – Tabla de pools de conexiones 2



Creación del recurso JDBC

La creación de un recurso JDBC inicia con la selección de la opción “JDBC Resources” en el árbol de opciones en la consola de administración tal como se ve en la figura 64.

Figura 64. Tabla de pools de conexiones 3



Una vez presionado el botón de nuevo se procede a la creación del recurso JDBC lo cual es muy simple, pues consiste en poner un nombre y seleccionar el origen de datos que se creó en el apartado anterior.

El formulario de creación para el recurso JDBC se puede ver en la figura 65, la única restricción en la creación del recurso en el nombre pues debe ser exactamente “jdbc/sacps”, pues este nombre está especificado en el archivo de configuración del sistema.

Figura 65. Creación de recurso JDBC 1

New JDBC Resource OK Cancel

Specify a unique JNDI name that identifies the JDBC resource you want to create. The name must contain only alphanumeric, underscore, dash, or dot characters.

JNDI Name: *

Pool Name:

Use the JDBC Connection Pools page to create new pools

Description:

Status: Enabled

Additional Properties (0)

Name	Value	Description:
------	-------	--------------

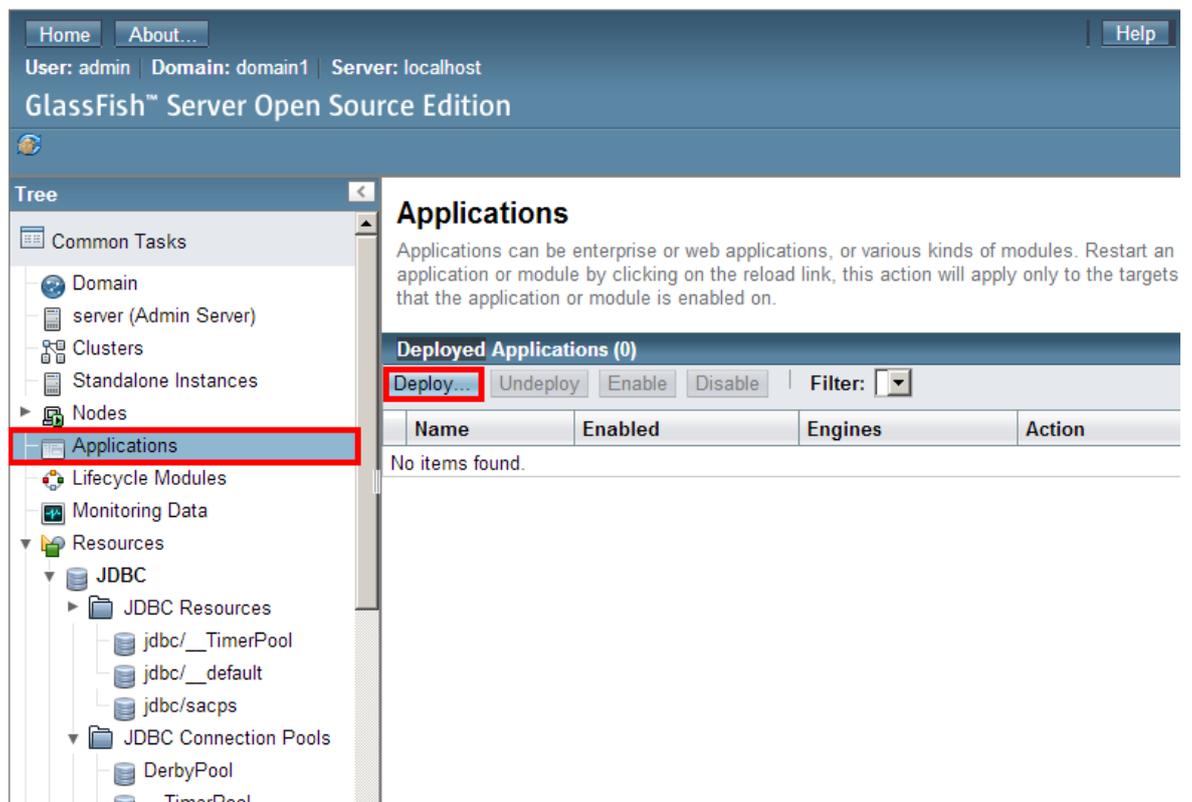
Elaborado por: Víctor Quimbiamba

Desplegar aplicación

Una vez configurado el recurso JDBC se debe desplegar el recurso Enterprise ARchive (EAR) que se proporciona en el anexo digital.

Lo primero que se debe hacer es seleccionar la opción “Applications” en el árbol de opciones de la consola de administración del servidor Glassfish para luego presionar el botón de “Deployed” como se puede ver en la figura 66.

Figura 66. Creación de recurso JDBC 2

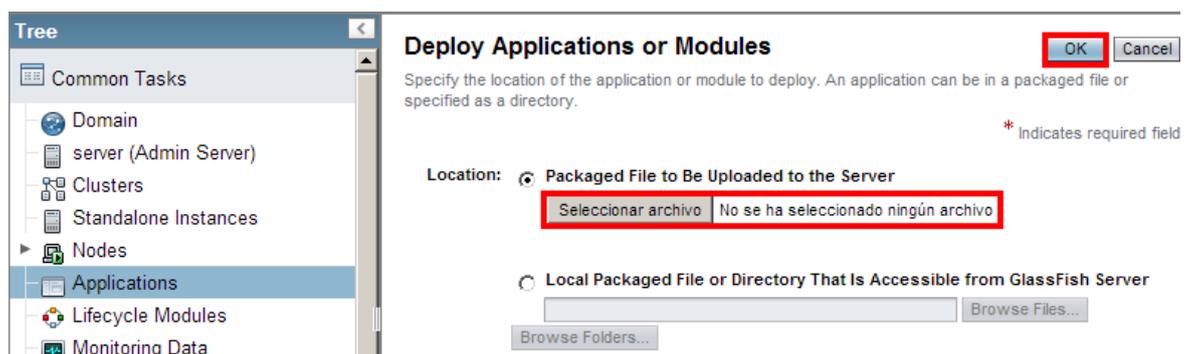


Elaborado por: Víctor Quimbiamba

El despliegue de la aplicación consiste en la selección de del archivo que empaqueta la aplicación web, en este caso será el archivo “sacps.ear” que estar en el anexo digital.

La figura 67 muestra el formulario de despliegue de aplicaciones o módulos.

Figura 67. Despliegue de aplicaciones o módulos – Selección de archivo



Elaborado por: Víctor Quimbiamba

Al presionar el botón “OK” el servidor Glassfish tomará el paquete EAR con todos los módulos y los desplegará, si el proceso de despliegue de aplicación es correcto se podrá ver el resultado como el de la figura 68.

Figura 68 – Lista de aplicación o módulos desplegados.

The screenshot shows the GlassFish Server Open Source Edition interface. The top navigation bar includes 'Home', 'About...', and 'Help' buttons. Below this, the user information is displayed: 'User: admin | Domain: domain1 | Server: localhost'. The main title is 'GlassFish™ Server Open Source Edition'. On the left, there is a 'Tree' view showing the server structure, with 'Applications' selected. The right pane is titled 'Applications' and contains a description: 'Applications can be enterprise or web applications, or various kinds of modules. Restart an application or module by clicking on the reload link, this action will apply only to the targets that the application or module is enabled on.' Below the description is a section for 'Deployed Applications (1)' with a table. The table has columns for 'Name', 'Enabled', 'Engines', and 'Action'. The row for 'sacps' is highlighted with a red border. The 'Enabled' column for 'sacps' contains a checkmark, and the 'Action' column contains '| Redeploy | Reload'.

Name	Enabled	Engines	Action
sacps	✓	ear, web, ejb	Redeploy Reload

Elaborado por: Víctor Quimbiamba

3.3.2 Manual de usuario

Introducción

Esta aplicación fue diseñada bajo las necesidades requeridas por el centro psicológico “DR. P. Emilio Gambirasio” del Campus Girón de la Universidad Politécnica Salesiana.

Este manual pretende ser una ayuda para el uso de la aplicación, y toma las funcionalidades más importantes así como también separa la explicación de la funcionalidad para los dos perfiles que maneja la aplicación.

Acceso a la aplicación

La aplicación utiliza de un navegador web y usando la siguiente dirección URL <http://190.15.136.5:8080/sacps>. El usuario debe ingresar esta dirección, al cargar el contenido se mostrará una pantalla de bienvenida como el de la figura 69, inmediatamente a esto se abrirá una nueva ventana igual a la de la figura 70

Figura 69. Pantalla de Presentación de la aplicación



Elaborado por: Víctor Quimbiamba

Por lo general al iniciar la aplicación el navegador advertirá que se usará una ventana emergente, se recomienda presionar el botón “Opciones” y seleccionar la primera

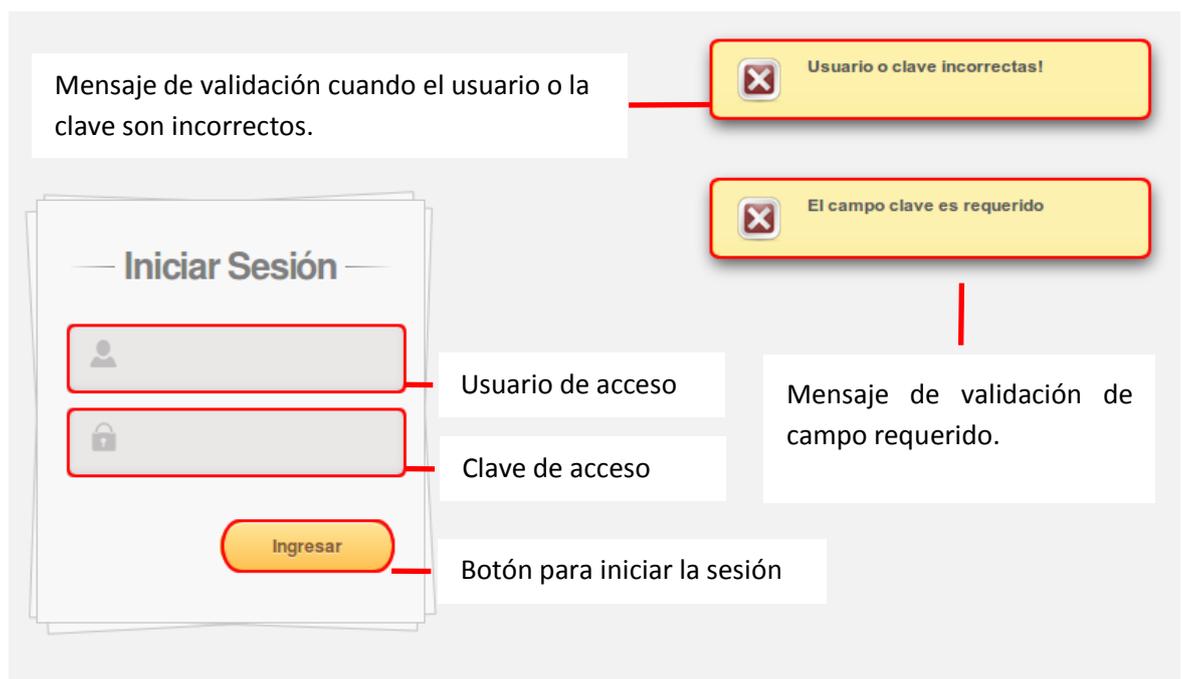
opción, con esto la próxima vez que se acceda a la aplicación la ventana emergente de presentará automáticamente, u opcionalmente existe un acceso presionando el link “aquí” que se encuentra en la parte inferior de la pantalla de acceso a la aplicación.

Ingreso a la aplicación

El acceso a la aplicación para los diferentes perfiles es único y sencillo como se muestra en la figura 70. Basta con ingresar el usuario, la clave y presionar el botón 'Ingresar'.

En el caso de ser la primera vez que se utilice el sistema el usuario y clave por defecto serán “admin”.

Figura 70. Pantalla de ingreso a la aplicación



Elaborado por: Víctor Quimbiamba

El botón con el texto 'Ingresar' procesa y valida el ingreso, de ser correcta la validación el ingreso se realizará según el perfil y presentará la pantalla principal.

Ayuda

Al pasar el cursor por cada campo se presenta una ayuda con la descripción del mismo.

Los campos de usuario y clave son sensibles a mayúsculas y minúsculas.

Importante

Después de ingresar por primera vez al sistema se debe crear un nuevo usuario (Ver cómo crear nuevo usuario) y un nuevo periodo (Ver cómo crear nuevo periodo) puesto que el usuario “admin” será deshabilitado, luego de esto se debe salir del sistema e ingresar con el usuario anteriormente creado.

Perfil administrativo

El perfil administrativo está encargado de la parte operativa de la aplicación esto incluye:

- Registro de pacientes

- Registro de médicos

- Registro de citas médicas

- Cobro y cancelación de citas médicas

- Registro de préstamos y devoluciones

- Administración (Tipos de Materiales, Materiales, Tipo de Servicios, Periodo, Reportes)

La figura 71 muestra la pantalla principal del perfil administrativo la cual posee las opciones de administración y accesos rápidos en el menú a las opciones principales de la aplicación.

Acceso

El acceso a todas las opciones de administración es mediante el menú de administración como se puede ver en la figura 71.

El resto de opciones están en el menú de acceso rápido.

Figura 71. Pantalla principal del perfil Administrativo



Elaborado por: Víctor Quimbiamba

Administración de usuarios

La funcionalidad de este caso es la misma para las pantallas de administración de Médicos, Materiales, Especialidades y Tipos de Servicio.

Lista de usuarios

La lista de usuarios es un formulario muy útil y se presenta como se muestra en la figura 72, esta lista presentará los usuarios de perfil administrativo del sistema, además es la pantalla inicial al momento de crear un nuevo usuario pues el botón de creación se posiciona en la cabecera de las opciones.

Figura 72. Lista de administración de usuarios

N°	Nombre	Teléfono	Correo Electrónico	Vigente	Opciones
1	GLORIA CHAMORRO	3962880	gchamorro@ups.edu.ec	<input checked="" type="checkbox"/>	
2	ADMIN	3441499	vquimbiamba@est.ups.edu.ec	<input checked="" type="checkbox"/>	

Elaborado por: Víctor Quimbiamba

Botones

Nuevo: Al presionar el botón de nuevo que está posicionado en la cabecera de la columna de las opciones se cambiará de formulario al de creación de usuarios de la Figura 73.

Editar: Cada registro de la tabla de usuarios poseerá en la columna de opciones el botón de editar, al presionar el botón editar se cargará el formulario de la figura 73 con los datos pre-cargados del usuario que se haya seleccionado para proceder con la edición.

Creación y edición de usuarios

La figura 73 muestra el formulario de ingreso y edición de usuarios, el tipo de usuarios que se crea en esta opción son usuarios de tipo administrativo.

Condiciones

- Todos los campos son requeridos.
- La longitud del usuario de acceso a la aplicación debe superar los 4 caracteres.
- La longitud de la clave debe superar los 5 caracteres.
- El usuario de acceso a la aplicación debe ser único.

Figura 73. Formulario de creación y edición de usuarios

CENTRO PSICOLÓGICO SALESIANO EMILIO GAMBIRASIO

ADMIN

UNIVERSIDAD POLITÉCNICA SALESIANA ECUADOR

Nueva Cita Médica Registrar Paciente Citas Pendientes Prestamos Administración

ADMINISTRACIÓN DE USUARIOS

Nuevo Guardar Regresar

Nombres:

Correo electrónico:

Teléfonos:

Dirección:

Usuario:

Clave:

Vigencia:

Fecha Creación: 27/05/2013

Usuario Creación: admin

Fecha Actualización:

Usuario Actualización:

Limpia el formulario e inicia la creación de un nuevo usuario.

Procesa las validaciones y guarda el usuario con los datos ingresados

Para una mejor experiencia recomendamos: Microsoft IE 9 + Mozilla Firefox 4.0 + Google Chrome 10 Apple Safari 5 Página optimizada (1024x768)

Elaborado por: Víctor Quimbiamba

Acciones:

- Nuevo: Al presionar el botón nuevo el formulario se limpiará dejándolo listo para la creación de una nuevo registro.
- Guardar: Al presionar el botón guardar se validarán los campos del formulario. En el caso de que se produzca un error en el momento de guardar el usuario se mostrará un mensaje de acuerdo al estándar indicando el error. En el caso de la edición realizará el mismo proceso y se procederá a actualizar el registro.
- Regresar: Al presionar el botón regresar se abandona la pantalla y se llama a la pantalla de la lista de usuarios correspondiente a la figura 72.

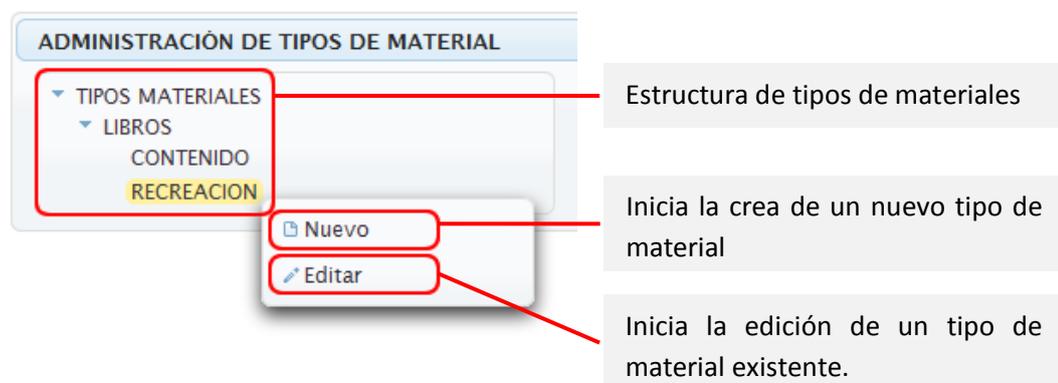
Importante

Al editar un usuario y desactivar la vigencia, esto implicará que el usuario no tendrá acceso a la aplicación.

Administración de tipos de materiales

El comportamiento de la administración de los tipos de materias es diferente a la administración de los demás catálogos pues está estructurada a manera de un árbol como se puede ver en la figura 74.

Figura 74. Lista de tipos de Materiales



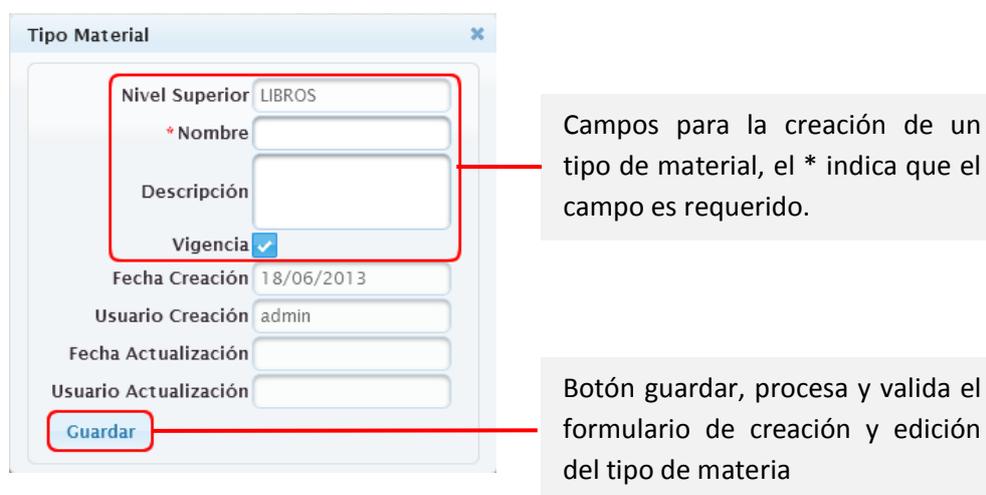
Elaborado por: Víctor Quimbiamba

Creación y edición

La figura 75 muestra el formulario de creación y edición de un tipo de material, el formulario se presenta en una ventana emergente.

Al igual que todo registro si el tipo de material tiene la vigencia desactivada el registro no aparecerá en la administración de materiales.

Figura 75. Formulario de creación y edición de tipo material



The image shows a screenshot of a web application window titled "Tipo Material". The window contains a form with the following fields and controls:

- Nivel Superior:** A dropdown menu with the value "LIBROS".
- * Nombre:** A text input field. A red box highlights this field, and a callout box points to it with the text: "Campos para la creación de un tipo de material, el * indica que el campo es requerido."
- Descripción:** A larger text input field.
- Vigencia:** A checkbox that is checked.
- Fecha Creación:** A date input field with the value "18/06/2013".
- Usuario Creación:** A text input field with the value "admin".
- Fecha Actualización:** A date input field.
- Usuario Actualización:** A text input field.
- Guardar:** A blue button at the bottom left. A red box highlights this button, and a callout box points to it with the text: "Botón guardar, procesa y valida el formulario de creación y edición del tipo de materia".

Elaborado por: Víctor Quimbiamba

Registro de pacientes

El formulario para la creación de un nuevo paciente se puede ver en la figura 76

Figura 76. Formulario de creación y edición de paciente

The image shows a web-based form titled "REGISTRO DE PACIENTE". At the top left, there is a button labeled "GUARDAR" (Save) with a red border. To its right, a text box explains: "Procesa y valida el formulario para guardar y generar un número de historia." Below this is a section titled "Información General" containing several fields:

- * Remitido por: [text input]
- * Lugar: Quito
- * Fecha: 18/06/201 [calendar icon]
- * Nombre del profesional: [text input]
- * Nombres y Apellidos: [text input]
- * Fecha de nacimiento: [calendar icon]
- * Edad: [text input]
- * Estado civil: Seleccione... [dropdown menu]
- * Nacionalidad: [text input]
- * Ocupación: [text input]
- * Género: Masculino [dropdown menu]
- * Nivel de educación: Seleccione... [dropdown menu]
- * Teléfonos: [text input]
- * Dirección: [text input]

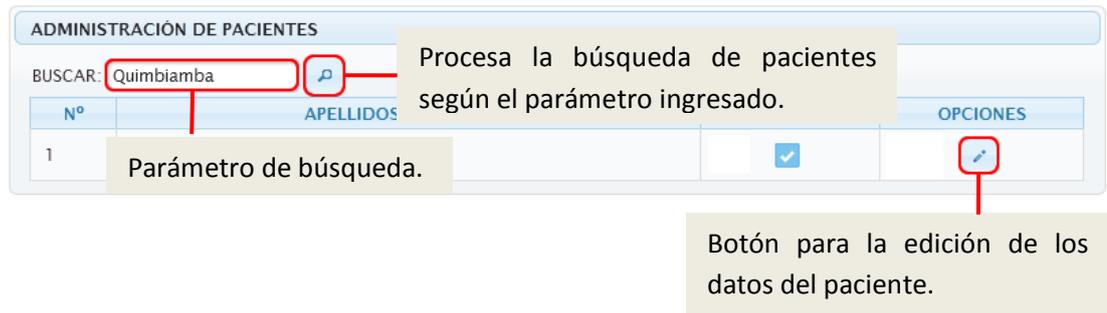
Three callout boxes provide additional instructions:

- A box pointing to the "Remitido por" field: "Selecciona un médico al que se le asigna el caso inicial."
- A box pointing to the "Fecha de nacimiento" field: "Debe ser una fecha en el pasado"
- A box pointing to the "Edad" field: "La edad no es un dato que se ingresa, este dato se calcula cuando se ingresa la fecha de nacimiento"

Elaborado por: Víctor Quimbiamba

La figura 76 muestra el formulario de búsqueda de pacientes, en este formulario se ingresa el parámetro que es el nombre o apellido para luego buscar el paciente, de esta manera se evita la carga masiva de datos. La única opción que posee cada registro del resultado de la búsqueda es la de edición de los datos del paciente, al presionar en el botón de edición se muestra el formulario de la figura 77 con los datos correspondientes al registro seleccionado.

Figura 77. Lista de pacientes

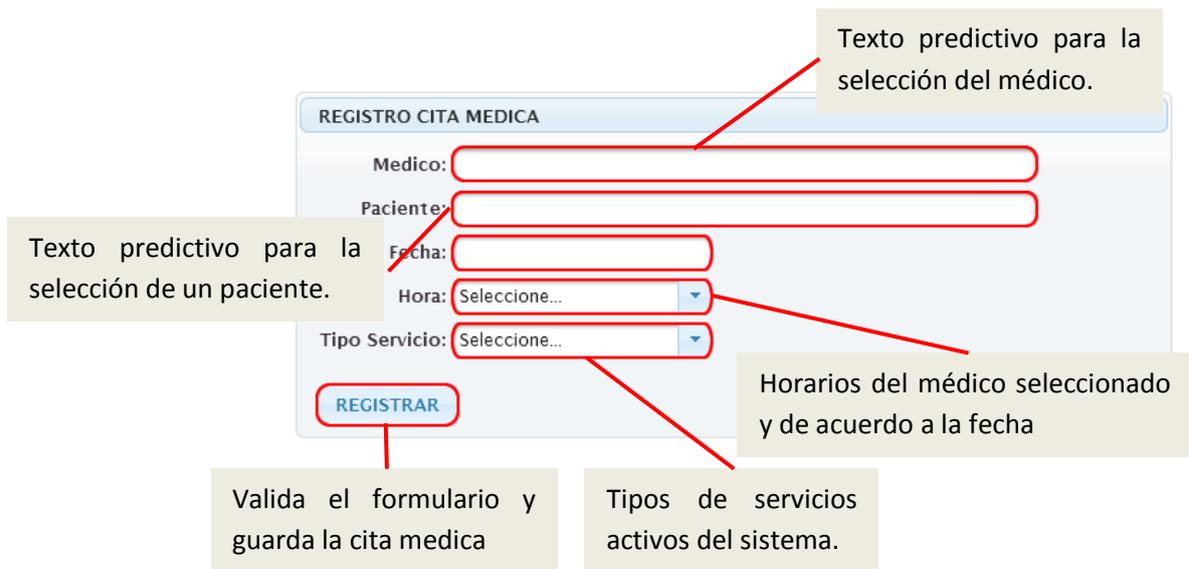


Elaborado por: Víctor Quimbiamba

Creación de cita médica

El formulario para la creación se muestra en la figura 78 y consta de datos simples.

Figura 78. Creación de cita médica



Elaborado por: Víctor Quimbiamba

La pantalla de registro de cita médica es muy funcional pues cuando se elige el médico y el horario existe una validación para calcular el día seleccionado y mostrar solo los horarios no usados en otras citas.

Manejo de citas pendientes

El manejo de citas pendientes se muestra en la figura 79, es una parte muy importante en la aplicación pues esta funcionalidad permite revisar citas pendientes, anular las citas o finalizar el proceso con el pago de la cita.

Figura 79. Lista de citas

The screenshot shows a web application interface for managing medical appointments. At the top, there is a search bar with a date filter set to '27/05/2013' and a status dropdown menu set to 'Pendiente'. Below the search bar is a table with columns: N°, MÉDICO, FECHA, HORA, ESTADO, and Opcion. The first row of the table shows appointment number 1, doctor MERCHAN HERRERA, date 27/05/2013, time 09:13 - 10:20, and status Pendiente. The 'Opcion' column for this row contains two buttons: 'x Cancelar' and 'Ver'. A dropdown menu is open over the 'Estado' column, showing options: 'Pendiente' (highlighted), 'Atendida', 'Finalizada', and 'Cancelada'. Callouts with red lines point to various elements: 'Filtros para la búsqueda' points to the search bar; 'Estados en los que puede estar una cita.' points to the dropdown menu; 'Toma los valores de los filtros, busca y muestra los resultados en la tabla.' points to the search bar and the first row of the table; and 'Opciones para la cita, estas opciones aparecen de acuerdo al estado de la cita, la opción 'Ver' es la única que se muestra en todos los estados' points to the 'Opcion' column.

N°	MÉDICO	FECHA	HORA	ESTADO	Opcion
1	MERCHAN HERRERA	27/05/2013	09:13 - 10:20	Pendiente	x Cancelar Ver

Elaborado por: Víctor Quimbiamba

Para cada opción de la lista de citas médicas de la figura 79 se presenta una ventana emergente de acuerdo al caso y puede cambiar el estado de la ficha y la disponibilidad de horarios en el caso de anular la cita médica.

Préstamos

El proceso de préstamos es una funcionalidad que no interviene con el proceso médico más bien es una opción de administración con funcionalidad extra, la pantalla para el registro de un préstamo se muestra en la figura 80.

Figura 80. Formulario de administración de préstamos

The image shows a web application interface for managing loans. The form is titled 'PRESTAMOS' and has two tabs: 'Préstamo' (selected) and 'Devolución'. Below the tabs are two buttons: 'Nuevo' and 'Guardar'. The form fields include: 'Material:' (with a dropdown menu), 'Usuario:' (with a text input), 'Fecha Entrega:' (with a date picker), 'Vigencia:' (with a checked checkbox), 'Fecha Creación:' (with a date input showing '19/06/2013'), and 'Usuario Creación:' (with a text input showing 'admin').

Annotations in the image:

- Opciones de administración de préstamo.** (Points to the 'Préstamo' and 'Devolución' tabs)
- Valida el formulario y guarda el préstamo, generando un nuevo número de préstamo útil para devolución del mismo.** (Points to the 'Nuevo' and 'Guardar' buttons)
- Texto predictivo del catálogo de materiales** (Points to the 'Material:' dropdown menu)
- Nombre de usuario al que se presta el material, el usuario no tiene que estar registrado en la aplicación.** (Points to the 'Usuario:' text input)
- Fecha programada de entrega, siempre será una fecha en el futuro.** (Points to the 'Fecha Entrega:' date picker)

Elaborado por: Víctor Quimbiamba

Para la devolución del material se usa el número de préstamo que se genera al registrar el préstamo, si existe algún comentario se ingresa caso contrario se registra la devolución directamente en el formulario de la figura 81.

Figura 81. Formulario de administración de devolución

PRESTAMOS

Valor del número de préstamo a buscar.

Numero prestamo:

Toma el valor del número de préstamo y procesa la búsqueda para muestra el resultado.

Material:

Usuario:

Fecha Entrega:

Resultado de la búsqueda.

Observaciones:

Observación para la devolución del material, no es obligatorio.

Registrar Devolución

Registra la devolución del material y lo cambia de estado.

Elaborado por: Víctor Quimbiamba

Horario médico

La administración de horarios inicia en la admiración de médicos como se puede ver en la figura 82 la opción de configuración de horario presentan el formulario de administración de horario como el de la figura 83. La configuración del horario se debe realizar cada vez que se crea un periodo y es personalizable para cada médico.

Figura 82. Lista médicos

ADMINISTRACIÓN DE MÉDICOS

Nº	Nombre	Especialidad	Vigente	Opciones
1	JUAN ANGLE DE LA CRUZ	CLINICO	<input checked="" type="checkbox"/>	<input type="button" value="Editar"/> <input type="button" value="Calendario"/>

Botón para la configuración de horarios.

Elaborado por: Víctor Quimbiamba

Figura 83. Lista de configuración de horarios

Nombre del médico al que se le configura el horario.

Toma el valor de día seleccionado y muestra una ventana emergente para la creación de un nuevo horario.

Regresa a la lista de médicos.

Registro de horario configurado.

Día para la configuración de horario, la selección de uno de los días actualiza la tabla de registros.

Inicia la edición de un horario existente.

Elaborado por: Víctor Quimbiamba

EL formulario para la creación y edición de un horario es el que se muestra en la figura 84, es una ventana emergente en la cual se ingresan los datos necesarios mediante un componente de selección de hora y minutos. Cuando el horario va a ser guardado se valida que no exista un cruce con otro registro, si la validación es exitosa el horario se registra caso contrario se presenta un mensaje de validación.

Figura 84. Formulario de creación y edición de horarios

Valida los datos ingresados y crea un nuevo horario.

Componente para la selección de hora y minutos

Elaborado por: Víctor Quimbiamba

Periodo

La administración de periodo es una parte muy importante de la aplicación pues esto define los horarios y los reportes, cada vez que se crea un periodo los horarios de cada médico son desactivados, pues cada periodo posee un horario diferente, además los reportes son generados con respecto al periodo.

La presentación de la administración del periodo es muy simple como se muestra en la figura 85, consta de una lista y para la creación se usa una ventana emergente, el periodo no puede ser editado ni eliminado.

Figura 85. Lista de periodos

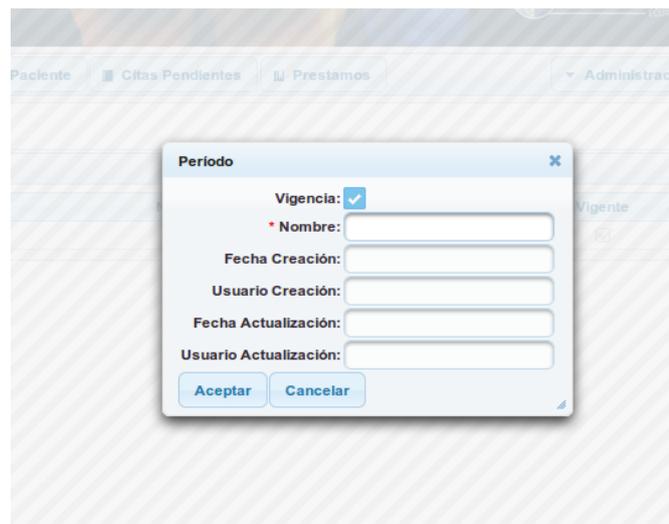


N°	Nombre	Vigente
1	2013	<input checked="" type="checkbox"/>

Elaborado por: Víctor Quimbiamba

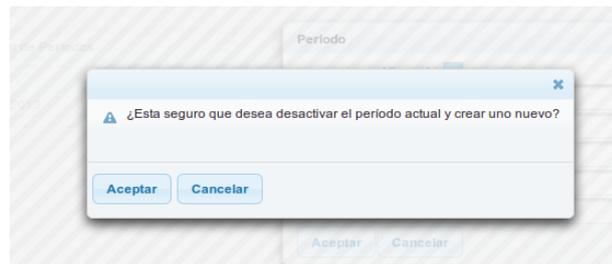
El formulario de creación de un nuevo periodo se puede observar en la figura 86. Cada vez que se crea un periodo se lanza una advertencia como se puede ver en la figura 3.3.30, con esto se confirma el proceso de creación de un nuevo periodo.

Figura 86. Ventana emergente de creación de nuevo periodo.



Elaborado por: Víctor Quimbiamba

Figura 87. Mensaje de confirmación para la creación de un nuevo periodo.



Elaborado por: Víctor Quimbiamba

Importante

La primera vez que se accede a la aplicación se debe crear un periodo, después de esto cada vez que se vaya a crear un nuevo periodo se debe tener en cuenta que no hay proceso de reverso y la información se actualizará al nuevo periodo.

Reportes

La generación de reportes inicia con la lista de reportes como se muestra en la figura 88. El valor por defecto del filtro para el periodo será el periodo actual pero podrá seleccionar los periodos anteriores, además se podrá realizar las consultas en un rango de fechas, una vez seleccionados los filtros se selecciona uno de los reportes

Figura 88. Formulario de generación de reportes.



Elaborado por: Víctor Quimbiamba

Perfil médico

El perfil de médico tiene una funcionalidad muy especial pues es aquí en donde se registran las atenciones de las citas médicas y una vez finalizada la sesión el médico guarda la observación de la cita y cambia al estado correspondiente.

La atención de una cita médica programada inicia con la lista de citas pendientes para la fecha actual como se muestra en la figura 89. Esta pantalla también es la pantalla principal del médico.

Figura 89. Pantalla principal del perfil médico

Muestra la pantalla principal cargando las citas pendientes del día.

Lanza una ventana emergente en la cual se puede editar información del médico.

Editar Perfil

CITAS MÉDICAS				
Nº	PACIENTE	FECHA	HORA	OPCIONES
1	ORTEGA CALVOPIÑA DANIEL ALEJANDRO	19/06/2013	09:00 - 10:00	

Cita pendiente del día.

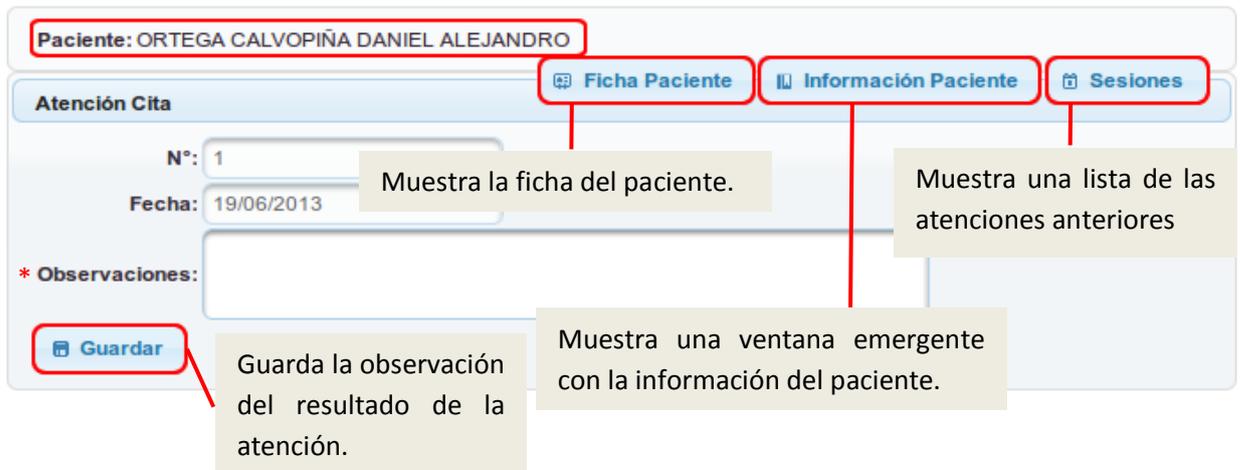
Inicia la atención de la cita médica.

Elaborado por: Víctor Quimbiamba

Atención de cita médica

La atención de una cita médica inicia en la pantalla principal del médico en la cual se muestra una lista de las citas médicas pendientes según la fecha actual como se muestra en la figura 89. Una vez presionado el botón de atención se cambia a la pantalla de la figura 90. El registro de una atención solo consta de una observación, puesto que cada caso consta de un seguimiento, además de esto se puede cambiar la información de la ficha del paciente como se puede ver en la figura 91.

Figura 90. Pantalla principal del perfil médico.



Elaborado por: Víctor Quimbiamba

La ficha del paciente se muestra en la figura 91. Es un formulario estándar proporcionado por el Centro Psicológico y se actualizan los datos de acuerdo al caso.

Figura 91. Ficha de paciente

Paciente: ORTEGA CALVOPIÑA DANIEL ALEJANDRO

FICHA

Vive en: Casa Departamento Pieza
 propia arrendada prestamo

Enfoque terapéutico:

Motivo de Consulta:

Demanda:

Apreciación Diagnostica:

Reactivos aplicados:

Resultados obtenidos:

Se sugiere:

Proceso Psicológico:

Psicoterapia: Psicoterapia para el padre: Psicoterapia para la madre:
Terapia de pareja (padres): Terapia familiar:
Terapia de lenguaje: Terapia psicomotriz: Asesoramiento:

Evaluaciones:

Evaluación Psiquiátrica: Evaluación Neurológica:
Evaluación Neuropsicológica:

Área Psicopedagógica:

Evaluación Psicopedagógica: Recuperación Académica:
Orientación Profesional:

Otros:

Elaborado por: Víctor Quimbiamba

CAPITULO 4

PRUEBAS

4.1 Pruebas funcionales

Para la realización de las pruebas funcionales se utilizó la herramienta Selenium (Ver anexo 3) que es un entorno de pruebas de software para aplicaciones basadas en la web. Selenium permite grabar y reproducir la interacción con una aplicación para crear pruebas sin usar un lenguaje de scripting.

El resultado de las siguientes pruebas es una muestra del correcto funcionamiento de las principales funcionalidades de la aplicación.

Prueba – Ingreso a la aplicación

Tabla 22. Especificación Ingreso a la Aplicación

Opción	Inicio de sesión
Rol	Administrador, Médico
Detalle	La aplicación presentará el formulario de inicio de sesión como el de la figura 3.3.13, solicita el ingreso de la siguiente información: <ul style="list-style-type: none">- Nombre de usuario de acceso- Clave para el usuario de acceso
Resultado esperado	Una vez ingresados los datos y presionado en el botón de ingreso, se actualiza la pantalla y se mostrará la pantalla principal y las opciones del menú cambiarán de acuerdo al perfil.

Elaborado por: Víctor Quimbiamba

Resultado

Luego de varias pruebas en el ingreso a la aplicación se concluyó que el proceso funciona correctamente.

Prueba – Registro de pacientes

Tabla 23. Especificación de Registro de Pacientes

Opción	Registro de pacientes
Rol	Administrador
Detalle	<p>La aplicación presentará el formulario de registro de pacientes como el de la figura 3.3.19, solicita el ingreso de la siguiente información:</p> <ul style="list-style-type: none">- Remitido por- Lugar- Fecha- Nombre del profesional- Nombres y Apellidos- Fecha de nacimiento- Edad- Estado Civil- Nacionalidad- Ocupación- Género- Nivel de educación- Teléfonos- Dirección
Resultado esperado	Una vez ingresados los datos y presionado en el botón guardar se procesa el formulario y si las validaciones son correctas el paciente se registra y muestra un mensaje de proceso exitoso.

Elaborado por: Víctor Quimbiamba

Resultado

Luego de haber realizado varias pruebas con diferentes datos y casos se verificó el correcto funcionamiento del registro de pacientes.

Prueba – Registro de cita médica

Tabla 24. Especificación de Registro de Cita Médica

Opción	Registro de Cita Médica
Rol	Administrador
Detalle	La aplicación presentará el formulario de registro de cita médica como el de la figura 3.3.21, solicita el ingreso de la siguiente información: <ul style="list-style-type: none">- Médico- Paciente- Fecha- Horario- Tipo de servicio
Resultado esperado	Una vez ingresados los datos y presionado en el botón registrar se procesa el formulario y si las validaciones son correctas la cita se registra y muestra un mensaje de proceso exitoso.

Elaborado por: Víctor Quimbiamba

Resultado

Luego de haber probar la correcta funcionalidad del proceso de registro de citas médicas y hacer varias pruebas con los horarios y médicos, el resultado reveló el correcto funcionamiento.

Prueba – Préstamos y devoluciones

Tabla 25. Especificación de Préstamos y Devoluciones

Opción	Préstamos y devoluciones
Rol	Administrador
Detalle	<p>Préstamo</p> <p>La aplicación presentará el formulario de registro préstamo como el de la figura 3.3.23, solicita el ingreso de la siguiente información</p> <ul style="list-style-type: none">- Materia- Usuario- Fecha entrega <p>Devolución</p> <p>La aplicación presentará el formulario de devolución como el de la figura 3.3.24, solicita el ingreso de la siguiente información</p> <ul style="list-style-type: none">- Número de préstamo
Resultado esperado	<p>Préstamo</p> <p>Una vez ingresados los datos y validados muestra un mensaje de éxito.</p> <p>Devolución</p> <p>Se ingresa el número de préstamo y el material está disponible para nuevos préstamos.</p>

Elaborado por: Víctor Quimbiamba

Resultado

Una vez realizada las pruebas con los casos de materias entre préstamos y devoluciones, se verificó que la funcionalidad se ejecuta correctamente.

Prueba – Registro paciente

Tabla 26. Especificación de Registro de Pacientes

Opción	Atención Cita
Rol	Médico
Detalle	<p>La aplicación presentará el formulario inicial para la atención de citas que corresponde a la figura 3.3.32 y se selecciona una cita por atender.</p> <p>El formulario para la atención solicita una observación del seguimiento del caso.</p> <p>Además de funcionalidades extras como edición de ficha y visualización de información.</p>
Resultado esperado	Una vez ingresada la observación se guarda y el estado de la cita cambia.

Elaborado por: Víctor Quimbiamba

Resultado

Luego de haber realizado varias pruebas con diferentes datos y casos se verificó el correcto funcionamiento de la atención de citas médicas y sus diferentes opciones.

4.2 Pruebas no funcionales

Las pruebas no funcionales se realizaron en la herramienta de desarrollo FireBug (Ver anexo 4) que posee entre otras la funcionalidad de monitorear las peticiones que se realizan en el navegador. Se determina el tamaño total de información enviada y recibida, así como el tiempo total que dura cada proceso.

Proceso – Inicio de sesión

El proceso inicia con la petición al servidor de la pantalla de autenticación y tras ingresar los datos correctos se muestra la pantalla principal, el proceso fue replicado y monitoreado logrando el resultado que se muestra en la figura 92.

Figura 92. Resultados de las pruebas al inicio de sesión

URL	Status	Domain	Size	Local IP	Timeline
POST index.jsf	302 Moved Temporarily	190.15.136.5:8080	213 B	192.168.1.5:49293	195ms
GET index.jsf	200 OK	190.15.136.5:8080	12,5 KB	192.168.1.5:49293	959ms
GET design.css	304 Not Modified	190.15.136.5:8080	4,6 KB	192.168.1.5:49296	142ms
GET calendar.js	304 Not Modified	190.15.136.5:8080	938 B	192.168.1.5:49297	191ms
GET autolumeri	304 Not Modified	190.15.136.5:8080	32,6 KB	192.168.1.5:49314	366ms
GET logout.png	304 Not Modified	190.15.136.5:8080	1,6 KB	192.168.1.5:49314	557ms
GET ajaxloadingl	304 Not Modified	190.15.136.5:8080	10,6 KB	192.168.1.5:49297	557ms
GET banner.jpg	304 Not Modified	190.15.136.5:8080	434 KB	192.168.1.5:49296	250ms
GET logomain.jp	304 Not Modified	190.15.136.5:8080	52,4 KB	192.168.1.5:49293	156ms
9 requests			549,3 KB	(536,5 KB from cache)	1.41s (onload: 2.23s)

Elaborado por: Víctor Quimbiamba

Análisis de resultados

Tabla 27. Resultados de pruebas inicio de sesión

Proceso	Inicio de sesión
URL Solicitud	/sacps/index.jsf
Tiempo total	Proceso (1,41 s) Carga (2,23 s)
Información transferida	549,3 KB (536.5 KB en cache)

Elaborado por: Víctor Quimbiamba

Proceso – Creación de usuario

El proceso inicia en formulario de creación de usuario, después de ingresar los datos se procesa el formulario y el resultado de este proceso se muestra en la figura 93.

Figura 93. Resultados de las pruebas a la creación de usuario.

URL	Status	Domain	Size	Local IP	Timeline
GET usuario.jsf	200 OK	190.15.136.5:8080	21,8 KB	192.168.1.5:49523	775ms
POST usuario.jsf	200 OK	190.15.136.5:8080	7,5 KB	192.168.1.5:49523	1.86s
2 requests			29,3 KB	2.63s (onload: 1.52s)	

Elaborado por: Víctor Quimbiamba

Análisis de resultados

Tabla 28. Resultados de pruebas creación de usuario.

Proceso	Creación de usuario
URL Solicitud	/sacps/paginas/administracion/usuario/usuario.jsf
Tiempo total del proceso	Proceso (263 ms) Carga (1.52 s)
Información transferida	29,3 KB

Elaborado por: Víctor Quimbiamba

Proceso – Registro cita médica

El proceso inicia en formulario de creación de cita médica, después de ingresar los datos se procesa el formulario y el resultado de este proceso se muestra en la figura 94

Figura 94. Resultados de las pruebas al Registro de Cita Médica

URL	Status	Domain	Size	Local IP	Timeline
GET cita_medica.j	200 OK	190.15.136.5:8080	20 KB	192.168.1.5:49584	689ms
POST cita_medica	200 OK	190.15.136.5:8080	7,7 KB	192.168.1.5:49584	274ms
POST cita_medica	200 OK	190.15.136.5:8080	1,6 KB	192.168.1.5:49584	305ms
POST cita_medica	200 OK	190.15.136.5:8080	543 B	192.168.1.5:49584	195ms
POST cita_medica	200 OK	190.15.136.5:8080	543 B	192.168.1.5:49584	215ms
POST cita_medica	200 OK	190.15.136.5:8080	1,4 KB	192.168.1.5:49584	243ms
POST cita_medica	200 OK	190.15.136.5:8080	523 B	192.168.1.5:49584	233ms
7 requests			32,3 KB	2.75s (onload: 1.38s)	

Elaborado por: Víctor Quimbiamba

Análisis de resultados

Tabla 29. Resultados de las pruebas al Registro de Cita Médica

Proceso	Registro de Cita Médica
URL Solicitud	/sacps/paginas/procesos/cita_medica/cita_medica.jsf
Tiempo total del proceso	Proceso (2.75) Carga (1.38 s)
Información transferida	32,3 KB

Elaborado por: Víctor Quimbiamba

Proceso – Registro paciente

El proceso inicia en formulario de registro de paciente, después de ingresar los datos se procesa el formulario y el resultado de este proceso se muestra en la figura 95

Figura 95. Resultados de las pruebas al Registro de Paciente

URL	Status	Domain	Size	Local IP	Timeline
GET paciente.jsf	200 OK	190.15.136.5:8080	32,6 KB	192.168.1.5:49721	1.28s
POST paciente.jsf	200 OK	190.15.136.5:8080	544 B	192.168.1.5:49721	200ms
POST paciente.jsf	200 OK	190.15.136.5:8080	2,4 KB	192.168.1.5:49721	250ms
POST paciente.jsf	200 OK	190.15.136.5:8080	19,4 KB	192.168.1.5:49721	613ms
4 requests			54,9 KB	2.34s (onload: 1.78s)	

Elaborado por: Víctor Quimbiamba

Análisis de resultados

Tabla 30. Resultados de las pruebas al Registro de Paciente

Proceso	Registro de Paciente
URL Solicitud	/sacps/paginas/administracion/paciente/paciente.jsf
Tiempo total del proceso	Proceso (2.34) Carga (1.78 s)
Información transferida	54,9 KB

Elaborado por: Víctor Quimbiamba

Conclusiones

- El desarrollo de un proyecto real ayuda a comprender el uso de metodologías para administrar y desarrollar software, pues en la realidad si un proyecto no está debidamente organizado fracasará.
- El uso de una arquitectura de negocio basado en componentes como el de los EJB, proporciona una mejor organización y un manejo más fácil, pues es el servidor que se encarga del administrar los componentes y el desarrollador solo los usa.
- La plataforma de desarrollo y ejecución JEE resulta ser muy útil pues sigue un estándar de desarrollo definido por N capas, con esto el desarrollo de la aplicación siempre está claramente separado de acuerdo a la funcionalidad y esto hace posible que el desarrollo sea más rápido.
- La realización de este proyecto de tesis muestra una visión clara de la necesidad de planificación y gestión de un producto informático, ya que si un proyecto está bien administrado y organizado el proyecto podrá salir adelante con un mínimo de problemas.
- Después de un proceso completo para el desarrollo de un producto de software surgen las conclusiones de cómo se hizo un proceso y la propuesta de una mejora en su realización.

Recomendaciones

- En cuanto al tiempo de desarrollo es muy importante estimar el tiempo real libre que se tiene para trabajar sobre las tareas de un tema de tesis, y no planificar el tiempo sin tomar en cuenta las actividades como el trabajo, etcétera.
- Analizar el tiempo de los involucrados en el tema, como el usuario final o el tutor, el tiempo disponible que los involucrados posean será el que se debe estimar además tomar en cuenta posibles eventualidades pues esto es un riesgo para el desarrollo del trabajo.
- Obtener información sobre el tema seleccionado, así como el proceso completo de desarrollo es muy útil tener una idea inicial de lo que se va a tratar, para luego esclarecer la base de conocimiento y obtener un resultado positivo.
- Correr un riesgo y utilizar herramientas que estén a la vanguardia en el mercado, así se recogerá más conocimiento y se enriquecerá el proyecto.
- A la institución a apoyar este tipo de proyectos pues es una ayuda mutua, entre estudiante que obtiene una experiencia real y la institución que puede beneficiarse con el resultado de una tesis.
- Incentivar el uso de nuevas tecnologías y más investigación pues esto ayudará en el proceso de elección de un tema así como en su realización.

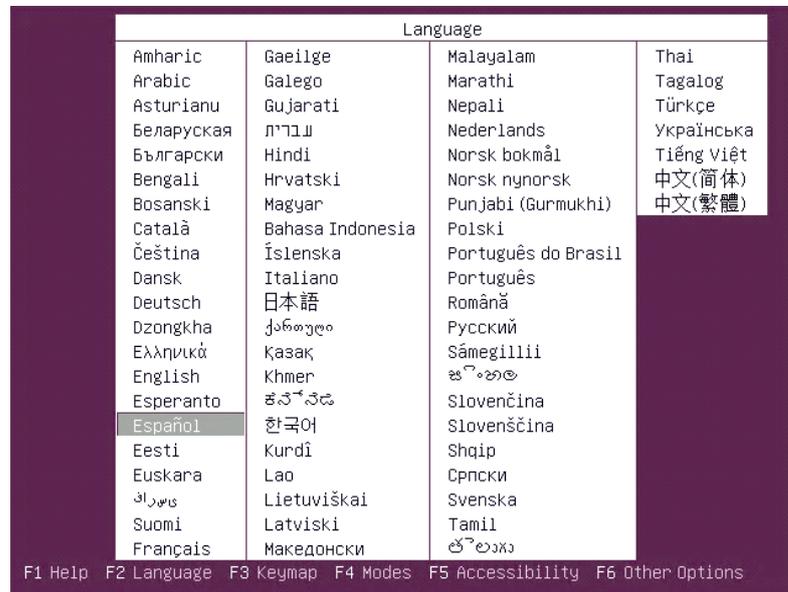
LISTA DE REFERENCIAS

- Burns , Schalk , Griffin (2009). JavaServer Faces 2.0: the Complete Reference. U.S.A.: Osborne/McGraw-Hill.
- Goncalves (2010). Beginning Java EE 6 with GlassFish 3. U.S.A. : Apress.
- Hans Bergsten (2004). JavaServer Faces. U.S.A.: O'Reilly Media
- Keith, Schinariol (2006). Pro EJB 3 - Java Persistence API. U.S.A.: apress.
- Kodali , Wetherbee , Zadrozny (2006). Beginning EJB 3 Application Development: From Novice to Professional. U.S.A. : Apress.
- Kou (2010). GlassFish Administration U.S.A. : Packt.
- Monson-Haefel, Burke (2006). Enterprise JavaBeans 3.0, 5th Edition. U.S.A. : O'Reilly Media.
- Pollice, Augustine , Lowe , Madhur (2003). Software Development for Small Teams : A RUP-Centric Approach. U.S.A. : Addison-Wesley.
- Robert V. Binder (1999). Testing Object-Oriented Systems: Models, Patterns, and Tools. U.S.A. : Addison-Wesley.
- Surhone, Tennoe, Henssonow (2010). Eclipselink. U.S.A. : Paperback.
- Varaksin, Caliskan (2013) - PrimeFaces Cookbook. U.S.A. : Packt Publishing

ANEXOS

Anexo 1 – Instalación Ubuntu Server

Figura 96. Selección de idioma de instalación.



Elaborado por: Víctor Quimbiamba

La figura 96 muestra la pantalla de selección de idioma de idioma para la instalación.

Figura 97. Selección de opción de instalación.



Elaborado por: Víctor Quimbiamba

La figura 97 muestra la pantalla principal de la instalación del sistema operativo y se debe seleccionar la primera opción es decir Instalar Ubuntu Server.

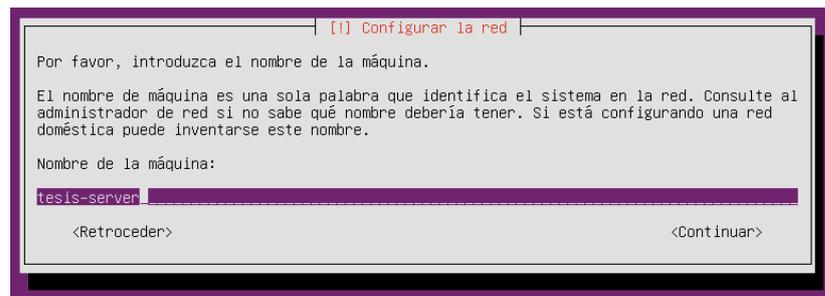
Figura 98. Descarga de componentes adicionales



Elaborado por: Víctor Quimbiamba

La figura 98 muestra el progreso de descarga de componentes adicionales, esto ocurre cuando el ordenador está conectado a una red con acceso a Internet, caso contrario se descargarán e instalarán al conectarse a Internet.

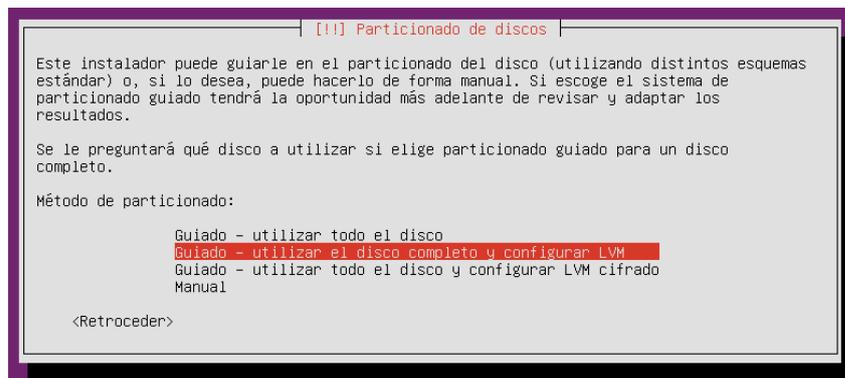
Figura 99. Configuración de Red



Elaborado por: Víctor Quimbiamba

La figura 99 muestra el diálogo que solicita el nombre que se asignará a la máquina para identificarla en la red.

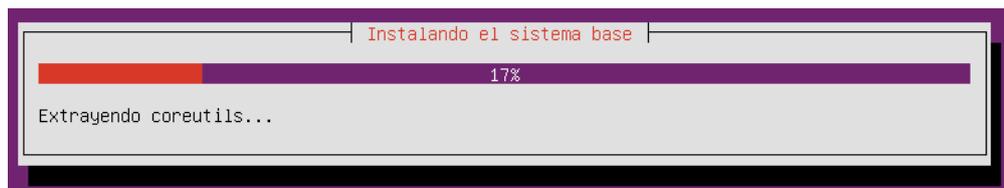
Figura 100. Particionado de discos



Elaborado por: Víctor Quimbiamba

La figura 100 muestra el dialogo para la selección del método de particionado que se realizará al disco, si el nivel de experticia es medio alto se puede seleccionar la opción manual para realizar una configuración de partición personalizada, luego de esto se presenta un cuadro de diálogo para la confirmación del particionado, este proceso toma un poco de tiempo para luego iniciar con la instalación del sistema base y el progreso de este se puede observar en la figura 101.

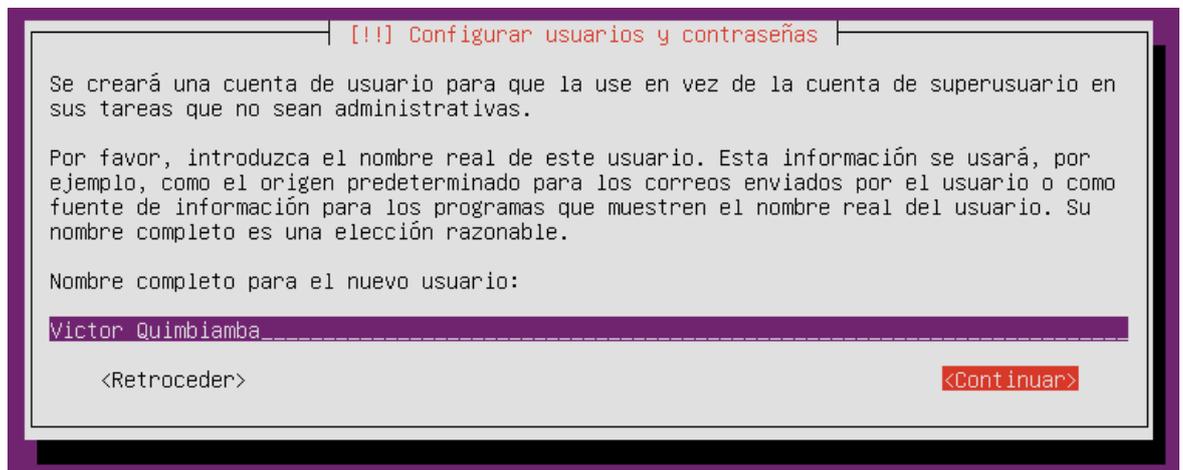
Figura 101. Progreso de instalación del Sistema Base



Elaborado por: Víctor Quimbiamba

La figura 102 muestra el cuadro de dialogo que solicita el nombre completo del usuario como referencia del usuario que se creará a continuación.

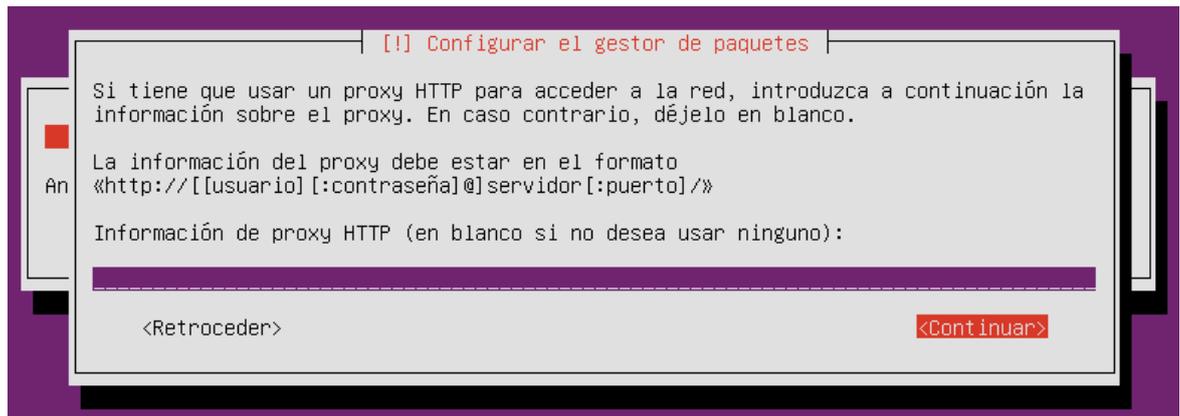
Figura 102. Configuración de usuario y contraseña - Nombre real



Elaborado por: Víctor Quimbiamba

Después de haber ingresado el nombre real del usuario se presentan dos cuadros de diálogo parecidos en donde se ingresa el nombre del usuario para el inicio de sesión, la clave y la confirmación de la misma.

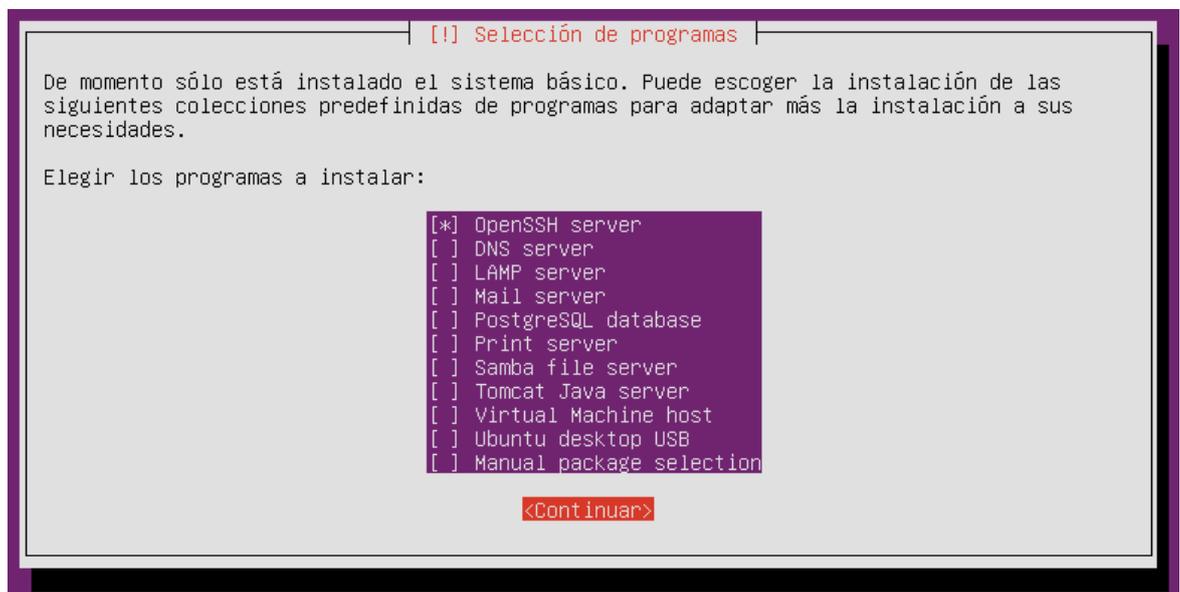
Figura 103. Configuración del gestor de paquetes



Elaborado por: Víctor Quimbiamba

La figura 103 muestra la configuración del gestor de paquetes en donde se puede configurar el proxy en caso de que el acceso a la red lo solicite.

Figura 104. Selección de programas

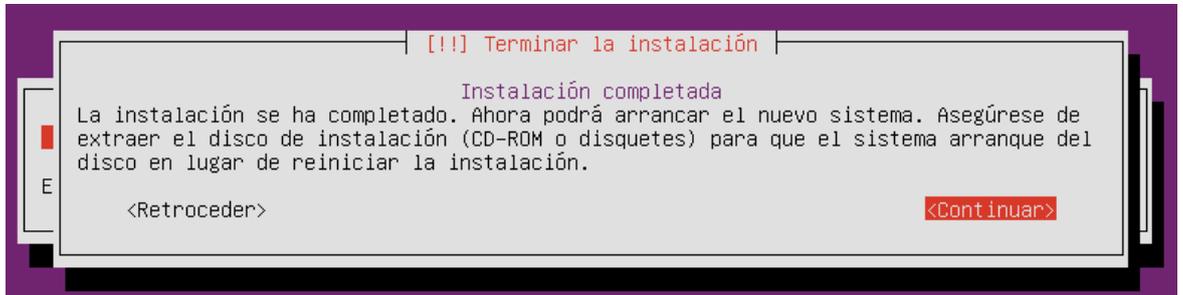


Elaborado por: Víctor Quimbiamba

La figura 104 muestra el dialogo de selección de programas disponibles, la lista y versión de los programas disponibles dependerá de la versión del sistema operativo que se esté instalando. Para este caso en específico el único programa extra que se

necesita es el “OpenSSH Server”, este programa permite realizar conexiones Secure Shell (SSH) lo cual es muy útil para el manejo remoto de equipos.

Figura 105. Descarga de componentes adicionales



Elaborado por: Víctor Quimbiamba

La figura 105 muestra el dialogo final de la instalación el cual solicita que se retire el CD de instalación para proceder a reiniciar el equipo. Una vez terminado esto el sistema se reiniciará y solicitará el usuario y clave que se crearon anteriormente en la instalación.

Anexo 2 – Instalación de la Máquina Virtual de Java

Para el caso de Linux, Windows u otro sistema operativo se necesita instalar los dos tipos de paquetes es decir el Java Development Kit (JDK) y Java Runtime Environment (JRE), estos paquetes pueden ser obtenidos desde la página oficial de Oracle (<http://www.oracle.com/technetwork/java/index.html>)

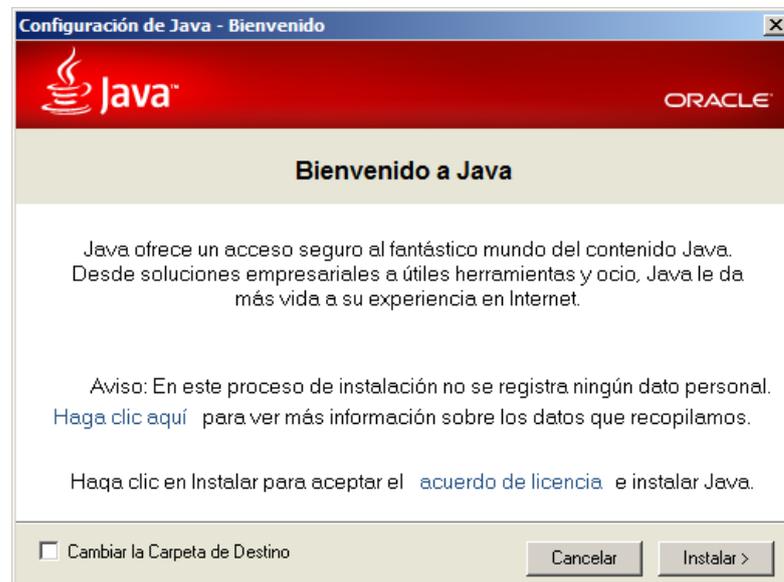
Instalación para Windows

Una vez descargadas los dos paquetes de la versión correspondiente para la arquitectura del sistema operativo que se use se procede a ejecutarlo como cualquier programa para Windows.

El primer paso será ejecutar el archivo ejecutable correspondiente a la versión JRE y posteriormente la versión JDK, una vez ejecutada aceptamos la solicitud de confirmación para realizar cambios en el equipo.

La figura 106 muestra la pantalla de bienvenida para la instalación del JRE.

Figura 106. Pantalla de bienvenida de instalación JRE



Elaborado por: Víctor Quimbiamba

La figura 107 muestra la pantalla de bienvenida de la instalación del JDK

Figura 107. Pantalla de bienvenida de instalación JRE



Elaborado por: Víctor Quimbiamba

Para los dos tipos de paquetes la instalación es simple, bastará con presionar el botón siguiente tantas veces se solicite, hasta que la instalación termine.

Instalación para Linux

En el caso específico de la distribución Debian se puede utilizar el siguiente comando:

```
sudo apt-get install openjdk-6-jdk openjdk-6-jre
```

Dependiendo de la distribución del Linux el gestor cambiara pero existen muchas fuentes en el origen de datos para cada distribución.

Anexo 3 – SELENIUM

Selenium es un set de herramientas en forma de extensión para Firefox que permiten la grabación y reproducción de la interacción con el explorador, además construir scripts para pruebas de aplicaciones web en diversos lenguajes como java, ruby, python, perl, .net o php. Es un producto open source que está permanentemente siendo optimizado.

La instalación del complemento para el explorador es simple, basta con buscar el complemento con el nombre de “selenium ide” como se muestra en la figura 108

Figura 108. Pantalla de instalación nuevo componente en Firefox



Elaborado por: Víctor Quimbiamba

El proceso de instalación solicitara reiniciar el explorado y posteriormente a esto en la barra de herramientas se agregará un nuevo botón como se muestra en la figura 108

Figura 109. Barra de herramientas con ícono Selenium IDE



Elaborado por: Víctor Quimbiamba

Cuando se presiona el ícono de Selenium IDE se presenta la ventana de la figura 109 con la cual ya se puede iniciar directamente hacer pruebas solo basta con poner el URL que se va a analizar presionar el botón de inicio para la grabación e interactuar con el explorador, para finalizar la grabación se presiona el mismo botón finalizando la grabación. La interacción con el navegador se registra en la tabla y se puede modificar si se necesita, para reproducir la grabación se presiona el botón de reproducción.

Figura 110. Ventana de Selenium IDE

URL para grabar la interacción

Base URL

Fast Slow

Test Case
Login *

Runs: 1
Failures: 0

Table

Command	Target	Value
open	/sacps/index.jsf	
type	id=j_idt8:username	admin
type	id=j_idt8:password	admin
clickAndWait	id=j_idt8:j_idt12	

Command: clickAndWait
Target: id=j_idt8:j_idt12
Value:

Log Reference Expert UI-Element Rollup

```
[debug] createForWindow
[debug] browserName: Netscape
[debug] userAgent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:21.0)
Gecko/20100101 Firefox/21.0
[debug] modifyWindow seleniumMarker1372997916417:undefined
[debug] _getFrameElement: frameElement=null
[debug] isPollingForLoad false, this.pollingForLoad[1]: undefined
[debug] Starting pollForLoad (selenium1372997916420):
http://190.15.136.5:8080/sacps/paginas/administracion/index.jsf
[debug] _getFrameElement: frameElement=null
[debug] pollForLoad original (selenium1372997916420):
http://190.15.136.5:8080/sacps/paginas/administracion/index.jsf
[debug] _isSamePage: currentHref: http://190.15.136.5:8080/sacps/paginas
/administracion/index.jsf
[debug] _isSamePage: originalHref: http://190.15.136.5:8080/sacps/paginas
/administracion/index.jsf
```

Lista de casos en análisis, cada caso puede ser con una misma URL o distinta

Inicia y finaliza la grabación

Inicia la reproducción de la grabación

Tabla en donde se muestra la interacción con el navegador

Propiedades para captura de la interacción, se puede editar

Tipos de salidas o resultados después de la reproducción

Log de la reproducción.

Elaborado por: Víctor Quimbiamba

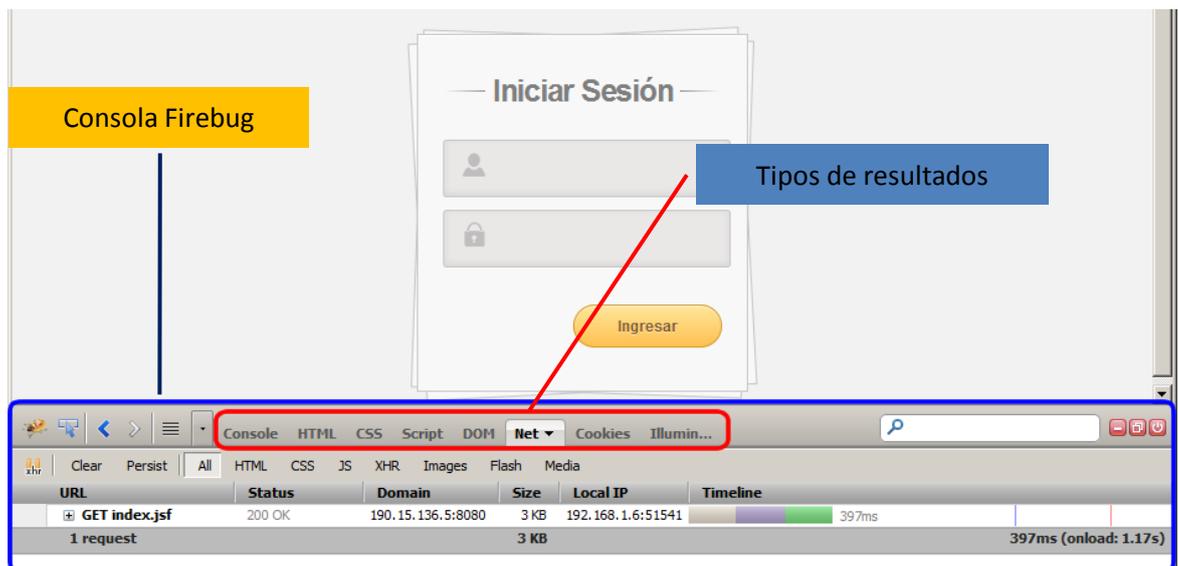
Anexo 4 – FIREBUG

Firebug es una extensión de Firefox creada y diseñada especialmente para desarrolladores y programadores web. Es un paquete de utilidades con el que se puede analizar (revisar velocidad de carga, estructura DOM), editar, monitorizar y depurar el código fuente, CSS, HTML y JavaScript de una página web de manera instantánea. (<http://es.wikipedia.org/wiki/Firebug>)

La instalación de del complemento Firebug es igual a la instalación de Selenium y se puede ver en el Anexo 3, además ya está disponible para exploradores como Google Chrome.

El uso es muy simple en cualquier explorador que se instale basta con presionar la tecla F12 y la ventana del explorador reservará un espacio en la parte inferior para la consola del Firebug como se puede ver en la figura 111

Figura 111. Barra de herramientas Firebug en explorador



Elaborado por: Víctor Quimbiamba

Una vez que se inicia el complemento Firebug captura las peticiones que se ejecuten en el navegador y separa los resultados.