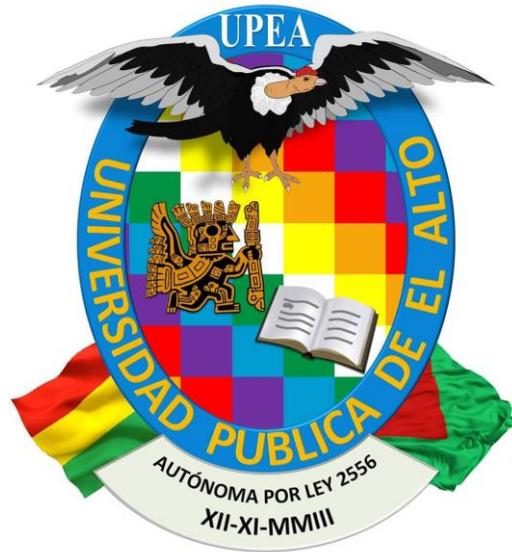


UNIVERSIDAD PÚBLICA DE EL ALTO

CARRERA INGENIERÍA DE SISTEMAS



PROYECTO DE GRADO

SISTEMA DE EMISIÓN DE RECIBOS, CONTROL EN DISTRIBUCIÓN DE INSUMOS MEDICOS EN LABORATORIOS CLÍNICOS

CASO: CARRERA DE ODONTOLOGÍA UNIVERSIDAD PÚBLICA DE EL ALTO

Para Optar al Título de Licenciatura en Ingeniería de Sistemas
MENCIÓN: GESTIÓN Y PRODUCCIÓN

Postulante: Carlos Daniel Apaza Parisaca

Tutor Metodológico: Ing. Marisol Arguedas Balladares

Tutor Especialista: Ing. Eduardo Rolando Flores Rodriguez

Tutor Revisor: Lic. Fredy Alanoca Coareti

EL ALTO – BOLIVIA

2020

DEDICATORIA

A las dos mujeres más especiales que están presentes en mi vida mi esposa Sandra Pamela por darme el valor de seguir luchando en la vida, con su constante apoyo y amor, a mi querida hija Gabriela Adalia por estar presente siempre en cada uno de los pasos que di, a mis queridos padres Cirilo y Virginia (con mucho cariño mamita a ti que estas al lado de nuestro creador), que me brindaron su apoyo incondicional. Al ser supremo Dios que guía mis pasos, a toda mi familia, gracias por estar presentes durante todo este proceso.

AGRADECIMIENTOS

A Dios por haberme permitido cumplir con esta meta de mi vida, agradecer a mis tutores que estuvieron presentes en el desarrollo y conclusión de este proyecto de grado. A mi tutor metodológico Ing. Marisol Arguedas Balladares por su conocimiento, apoyo, confianza, tiempo, paciencia y motivación constante para la culminación del presente proyecto. A mi tutor especialista Ing. Eduardo Rolando Flores Rodríguez por su conocimiento y orientación durante el desarrollo del presente proyecto. A mi tutor revisor Lic. Fredy Alanoca Coareti por su disponibilidad, su acertada orientación y sugerencias durante el desarrollo del presente proyecto. A la Clínica Odontológica de la carrera de Odontología de la UPEA por otorgarme la oportunidad y confianza de desarrollar el sistema. A la Universidad Pública de El Alto, por acogerme en sus aulas durante todos los años de estudio, a mi queridísima carrera Ingeniería de Sistemas.

¡¡¡Muchas Gracias a todos!!!

RESUMEN

En laboratorios clínicos de la Carrera de Odontología de la Universidad Pública de El Alto se ofrecen servicios a la población de la urbe y principalmente permite a los estudiantes de últimos años realizar sus prácticas profesionales. En esta institución actualmente se tienen problemas en la emisión de recibos y el control de distribución de insumos médicos, donde realizan varios procedimientos de forma manual, y además se ha identificado inconvenientes con la seguridad de los datos y la información del negocio.

El presente proyecto de grado propone el desarrollo de un sistema de emisión de recibos y control de distribución de insumos médicos para optimizar los recursos, procesos y procedimientos identificados. El sistema permite el registro de pacientes, estudiantes y docentes; permite el control de distribución de los insumos médicos y formularios de historial clínico que son dispensados a los estudiantes que hacen su práctica profesional y gestiona la emisión de recibos para un mejor control de ingresos económicos a la institución. El software se desarrolla con base al Método de Ingeniería Web basado en el Lenguaje de Modelado Unificado (UWE), en un entorno Web el cual puede ser accedido desde Internet o red privada. Asimismo, hace uso de herramientas de desarrollo, como ser: el lenguaje PHP con el framework Codeigniter; el gestor de base de datos MySQL, el servidor Web Apache y para el diseño del cliente Web se emplea Javascript y HTML5.

INDICE GENERAL

CAPITULO 1 MARCO PRELIMINAR	1
1.1. Introducción	1
1.2. Antecedentes	2
1.2.1. Antecedentes institucionales	2
1.2.2. Antecedentes internacionales.....	2
1.2.3. Antecedentes nacionales.....	3
1.3. Planteamiento del problema	3
1.3.1. Problema Principal.....	5
1.3.2. Problemas Secundarios.....	5
1.4. Objetivos	5
1.4.1. Objetivo General.....	5
1.4.2. Objetivos Específicos	5
1.5. Justificación	6
1.5.1. Justificación Técnica.....	6
1.5.2. Justificación Económica	6
1.5.3. Justificación Social	7
1.6. Metodología	7
1.6.1. Método de Ingeniería Web basado en el Lenguaje de Modelado Unificado	7
1.6.2. Kanban	7
1.7. Herramientas	8
1.7.1. Lenguaje de programación PHP.....	8
1.7.2. Sistema Gestor de Bases de Datos MySQL.....	8

1.7.3. Servidor Web Apache.....	8
1.8. Alcances y límites	8
1.8.1. Alcances	8
1.8.2. Límites	9
1.9. Aportes	9
CAPITULO 2 MARCO TEORICO.....	10
2.1. Odontología	10
2.1.1. Protocolo general de atención en odontología.....	10
2.2. Comprobante de pago	12
2.2.1. Tipos de comprobante de pago	12
2.3. Distribución de insumos.....	13
2.3.1. Distribución física.....	13
2.3.2. Control de inventario.....	13
2.3.3. Métodos de control de inventario	14
2.3.4. Insumos médicos	14
2.4. Ingeniería de software	14
2.4.1. Sistema informático	15
2.4.2. Modelos de proceso de software	16
2.4.4. Desarrollo ágil.....	18
2.4.5. Método Kanban.....	19
2.4.6. Metodología de ingeniería Web basado en el Lenguaje de Modelado Unificado.....	22
2.4.7. Modelos de arquitectura de software	24
2.4.8. Herramientas para desarrollo Web	25
2.4.8.1. Lenguaje de programación Web PHP.....	25

2.4.8.2. Sistema de gestión de base de datos MySQL	26
2.4.8.3. Servidor Web Apache	26
2.4.9. Pruebas técnicas para aplicaciones Web	27
2.4.9.1. Estrategia de pruebas para aplicaciones Web.....	27
2.4.10. Métricas de producto software	28
2.4.11. Método de estimación de costos basado en puntos de casos de uso ...	29
CAPITULO 3 MARCO APLICATIVO	32
3.1. Análisis de la situación actual	32
3.2. Análisis de requerimientos	36
3.2.1. Requerimientos funcionales.....	48
3.2.2. Requerimientos no funcionales.....	50
3.3. Diseño.....	52
3.3.1. Modelo de contenido.....	52
3.3.2. Modelo de navegación	53
3.3.3. Modelo de proceso	56
3.4. Construcción.....	60
3.4.1. Arquitectura del sistema	60
3.4.2. Base de datos	61
3.4.3. Interfaces gráficas de usuario	62
3.5. Pruebas	77
CAPITULO 4 CALIDAD Y SEGURIDAD	83
4.1. Evaluación de la calidad	83
4.2. Medidas de seguridad del sistema.....	84
4.2.1. Autenticación	84
4.2.2. Control de acceso basado en roles.....	84

4.2.3. Políticas de seguridad de la base de datos	84
4.3. Evaluación de seguridad Web	85
CAPITULO 5 COSTOS Y BENEFICIOS	87
5.1. Estimación de costos con base al método de Puntos de Casos de Uso	87
CAPITULO 6 CONCLUSIONES Y RECOMENDACIONES	91
6.1. Conclusiones	91
6.1. Recomendaciones	92
6.1.1. Recomendaciones para la institución	92
6.1.2. Recomendaciones para trabajos futuros	93
REFERENCIAS BIBLIOGRAFICAS	94
ANEXOS	98

INDICE DE FIGURAS

Figura 1. Protocolo General de Atención en Clínica Odontológica	11
Figura 2. Capas de la ingeniería de software.....	15
Figura 3. Proceso de transformación de datos en información	16
Figura 4. Representación del modelo de proceso de la cascada.....	17
Figura 5. Tablero Kanban.	21
Figura 6. Modelos de UWE.	23
Figura 7. Elementos de la notación en UWE.....	24
Figura 8. Representación del proceso de pruebas de aplicaciones Web.....	28
Figura 9. Procedimiento para el registro de tratamientos clínicos	33
Figura 10. Procedimiento para el registro de formularios de historial clínico	33
Figura 11. Procedimiento para el registro de estudiantes	34
Figura 12. Procedimiento para el registro de venta y emisión de recibos	34
Figura 13. Procedimiento para la anulación de recibos de venta.....	35
Figura 14. Procedimiento para emisión de reporte diario, semanal o mensual.....	35
Figura 15. Jerarquía de actores que interactúan con el sistema.....	37
Figura 16. Modelo general de casos de uso agrupado por paquetes	37
Figura 17. Casos de uso para gestionar usuarios.....	38
Figura 18. Casos de uso para gestionar unidades organizacionales	39
Figura 19. Casos de uso para gestionar cargos.....	39
Figura 20. Casos de uso para gestionar roles de usuario.....	39
Figura 21. Casos de uso para gestionar países.....	40
Figura 22. Casos de uso para gestionar proveedores.	41
Figura 23. Casos de uso para gestionar cargos.....	41
Figura 24. Casos de uso para gestionar cargos.....	41
Figura 25. Casos de uso para gestionar servicios.	42
Figura 26. Casos de uso para gestionar cargos.....	43
Figura 27. Casos de uso para gestionar cargos.....	43
Figura 28. Casos de uso para gestionar cargos.....	43

Figura 29. Casos de uso para gestionar notificaciones.....	44
Figura 30. Casos de uso para gestionar insumos.	44
Figura 31. Casos de uso para gestionar ingreso de insumos.	45
Figura 32. Casos de uso para gestionar salida de insumo.	45
Figura 33. Casos de uso para gestionar recibos.....	46
Figura 34. Casos de uso para gestionar pacientes.....	47
Figura 35. Casos de uso para gestionar docentes.....	47
Figura 36. Casos de uso para gestionar estudiantes.....	47
Figura 37. Casos de uso para gestionar reportes.....	48
Figura 38. Modelo de contenido, representa los datos que estarán presentes en la base de datos.....	54
Figura 39. Diagrama de navegación para el usuario con rol de Administrador.....	55
Figura 40. Diagrama de navegación para el usuario con rol de Encargado de insumos.....	55
Figura 41 Diagrama de navegación para el usuario con rol de Encargado de caja.....	56
Figura 42 Diagrama de navegación para el usuario con rol Personal de servicio.	56
Figura 43 Diagrama de actividad para especificar el procedimiento de acceso al sistema.....	57
Figura 44 Diagrama de actividad para especificar el procedimiento de listar ítems.....	58
Figura 45 Diagrama de actividad para especificar el procedimiento de registro de un ítem.....	58
Figura 46 Diagrama de actividad para especificar el procedimiento de actualizar un ítem.....	59
Figura 47 Diagrama de actividad para especificar el procedimiento de dar baja un ítem.....	59
Figura 48. Arquitectura del sistema.....	61
Figura 49. Formulario de acceso al sistema.....	63
Figura 50. Página de bienvenida al usuario.....	63
Figura 51. Interfaces gráficas para la gestión de unidades organizacionales.....	64

Figura 52. Interfaces gráficas para gestionar cargos	65
Figura 53. Interfaces gráficas para gestionar usuarios	66
Figura 54. Interfaces gráficas para gestionar estudiantes.....	67
Figura 55. Interfaces gráficas para gestionar docentes.	68
Figura 56. Interfaces gráficas para gestionar materias o especialidades.....	69
Figura 57. Interfaces gráficas para gestionar tratamientos	70
Figura 58. Interfaces gráficas para gestionar formularios de historial clínico	71
Figura 59. Interfaces gráficas para gestionar marcas de insumos.....	72
Figura 60. Interfaces gráficas para gestionar proveedores de insumos.....	73
Figura 61. Interfaces gráficas para gestionar unidades de medida.....	74
Figura 61. Interfaces gráficas para gestionar los insumos, sus ingresos y salidas	75
Figura 63. Interfaces gráficas para gestionar recibos	76
Figura 64. Interfaces gráficas para gestionar unidades de medida.....	77
Figura 65 Resultados obtenidos tras analizar la seguridad del sistema con Zed Attack Proxy	86

INDICE DE TABLAS

Tabla 1. Cantidad de horas – persona según el valor calculado en los factores ambientales.....	31
Tabla 2. Identificación de actores del sistema.	36
Tabla 3. Lista de requerimientos funcionales.	49
Tabla 4. Lista de requerimientos no funcionales.	50
Tabla 5 Requerimientos mínimos para el servidor.	52
Tabla 6. Requerimientos mínimos para las terminales de usuario.	52
Tabla 7 Atributos necesarios para la trazabilidad de registro de la base de datos	62
Tabla 8 Definición de casos de prueba de caja negra	78
Tabla 9 Resultados tras haber efectuado los casos de prueba de caja negra después de la tercera iteración o revisión del sistema	80
Tabla 10 Resultados de las pruebas de desempeño del sistema muestran un tiempo de respuesta menor a 5 segundos como esta especificado en los requerimientos	82
Tabla 11 Resultados de la evaluación de calidad del sistema realizado con base al estándar ISO/IEC 9126.	83
Tabla 12 Cálculo del Factor de peso de los actores sin ajustar	87
Tabla 13 Cálculo del Factor de peso de los casos de uso sin ajustar	88
Tabla 14 Cálculo de los factores técnicos	88
Tabla 15 Cálculo de los factores ambientales.....	89
Tabla 16 Estimación del costo de software	90

CAPITULO 1 MARCO PRELIMINAR

Resumen

En este capítulo se describe la problemática que aborda el presente proyecto de grado, realizando una revisión de antecedentes, la respectiva identificación de los problemas relacionados a los procesos y procedimientos que se efectúan en Laboratorios Clínicos de la Carrera de Odontología - Universidad Pública de El Alto, y finalmente se realiza el planteamiento de la solución con un enfoque de ingeniería de software.

1.1. Introducción

En el mundo, en las últimas dos décadas la implementación de sistemas de información para optimizar los recursos y los procesos de negocio ha sido más recurrente que en el siglo pasado. En Bolivia también se ha seguido esa tendencia para mejorar las empresas o instituciones con la finalidad de ser más eficientes.

En la Carrera de Odontología de la Universidad Pública de El Alto (UPEA) se tienen los laboratorios clínicos, también conocido como Clínica Odontológica, donde se ofrecen servicios a la población de la urbe y principalmente permite a los estudiantes de últimos años realizar sus prácticas profesionales. En esta institución actualmente se tienen problemas en la emisión de recibos y el control de distribución de insumos médicos, donde realizan varios procedimientos de forma manual, como ser: registro de venta y elaboración de recibo, contabilización de ingresos, consolidación de datos de caja para reportes, control de salidas de insumos, inventariado de insumos, entre otros. Además, se ha identificado inconvenientes con la seguridad de los datos y la información del negocio.

El presente proyecto de grado propone el desarrollo de un sistema de emisión de recibos y control de distribución de insumos médicos para optimizar los recursos, procesos y procedimientos identificados. El sistema permite el registro de pacientes, estudiantes y docentes; permite el control de distribución de los insumos médicos y formularios de historial clínico que son dispensados a los estudiantes que hacen su

práctica profesional y gestiona la emisión de recibos para un mejor control de ingresos económicos a la institución.

El sistema está diseñado con base al Método de Ingeniería Web basado en el Lenguaje de Modelado Unificado (UWE), y está desarrollado para un entorno Web el cual puede ser accedido desde Internet. Asimismo, hace uso de herramientas de desarrollo, como ser: el lenguaje PHP con el framework Codeigniter versión 3; el gestor de base de datos MySQL versión 8, el servidor Web Apache versión 2.4 y para el diseño del cliente Web se emplea Javascript y HTML5.

1.2. Antecedentes

1.2.1. Antecedentes institucionales

La Carrera de Odontología de la Universidad Pública de El Alto (UPEA) tiene por objeto la formación de profesionales en medicina bucal y estética. En este sentido, con el fin de formar profesionales idóneos la UPEA creó los laboratorios clínicos, públicamente conocido como Clínica Universitaria Odontológica en el año 2007 ubicado en la zona Villa Esperanza con el objetivo de posibilitar que los estudiantes de la carrera de odontología realicen prácticas completas para su titulación de pre-grado y post-grado; además se creó con la visión de ser el centro asistencial de referencia en odontología de la ciudad del El Alto (UPEA, sf). Esta institución lleva funcionando varios años hasta la fecha, atendiendo a más de 120.000 pacientes en las diversas especialidades odontológicas y permitiendo a estudiantes adquirir las competencias necesarias para su inserción al mercado laboral.

1.2.2. Antecedentes internacionales

En la tesis titulada “Sistema de emisión de comprobantes de pago electrónicos en el proceso de facturación de CONTASIS SAC” presentada en la Universidad Nacional del Centro del Perú, se aborda el impacto que tiene el uso del sistema de facturación electrónica sobre los costos del proceso de facturación en la empresa Contasis (Amaro, 2017). Para el desarrollo del sistema se utilizó el Proceso Unificado Ágil (AUP) como metodología.

En el trabajo titulado “Diseño e Implementación de la Factura Electrónica como mecanismo para hacer más eficiente el proceso de facturación en Certicámara S.A” presentado en la Universidad de la Salle, Colombia, se determina las condiciones de diseño y de implementación de la facturación electrónica en Certicámara S.A. (Herrera, 2011).

1.2.3. Antecedentes nacionales

En el proyecto de grado titulado “Sistema web integrado de control, pago e historial de socios basado en biométrico dactilar caso: empresa CIMCO LTDA” presentado en la Universidad Mayor de San Andres, se diseña e implementa un sistema web integrado de control, pago e historial de socios basado en biométrico dactilar para la cooperativa CIMCO LTDA con el fin de brindar información consistente, oportuna y segura (Tito, 2015). Para el desarrollo de este proyecto se utilizó la metodología de Proceso Unificado Ágil (AUP).

En el proyecto de grado titulado “Sistema web para el control y administración de recursos humanos CASO: Empresa de Limpieza Industrial TOTES LTDA”, presentado en la Universidad Mayor de San Andres, se ha desarrollado un sistema de información para la unidad de Recursos Humanos, que realiza tareas de reclutamiento de personal, asignación de personal a los contratos a las cuales brinda servicios de limpieza, control de asistencias y elaboración de planillas de pago (Monrroy, 2014). En este trabajo se hace uso de la Metodología de Proceso Unificado (RUP).

1.3. Planteamiento del problema

Luego de analizar la situación actual de laboratorios clínicos de la Carrera de Odontología se pudo evidenciar un conjunto de problemas en torno a dos escenarios, como ser: la emisión de recibos en cajas y el control en la distribución de insumos médicos.

En primera instancia, se ha identificado deficiencias en el registro y emisión de recibos debido a que el registro de ventas y elaboración de recibos se efectúa de forma manual, además que la contabilización de ingresos por venta de tratamientos y formularios se hace mediante la revisión de recibos en físico. Cabe hacer notar que los recibos sirven como documento de constancia para que un estudiante haga uso de otros servicios al interior de la institución. En consecuencia, hay una tendencia a fallos en la coordinación de trabajos con otras unidades que dependen del recibo como comprobante. Asimismo, suelen existir fallas en la consolidación de datos financieros, lo cual imposibilita justificar de manera precisa el gasto del presupuesto de la institución.

Por otra parte, se evidenció que el acceso a todos los datos de caja está centrado en Excel, ya que el registro de datos de estudiantes, materias o especialidades, tratamientos, pacientes y ventas se realiza en un solo documento electrónico. Como resultado existe riesgo de pérdida o alteración de la información del negocio sin posibilidad de rastreo de responsables, ya que cualquier persona con acceso a la computadora central de caja puede modificar los datos del documento Excel.

También, debido a que la consolidación y cálculos de datos se realizan de forma manual, se generan imprecisiones al momento de generar los reportes diarios, semanales y mensuales, lo que podría acarrear riesgos y pérdida de oportunidad en la toma de decisiones.

Finalmente, se ha identificado un déficit de control en la distribución de insumos médicos a los estudiantes, esto a causa de un inventario manual en dependencias de la Unidad de Insumos, la inexistencia de respaldos de información sobre el inventario de insumos para distribuir y el control manual de salidas de los insumos médicos. Como efecto se evidencia el riesgo de pérdida de insumos médicos no registrados correctamente, lo cual también imposibilita una justificación precisa de lo utilizado en un tiempo determinado ante las autoridades competentes.

1.3.1. Problema Principal

Con base a los problemas identificados en laboratorios clínicos de la Carrera de Odontología, y haber efectuado el análisis de problemas para el presente proyecto de grado (Anexo A), se formula el problema principal de la siguiente forma:

¿Cómo mejorar la eficiencia en los procesos de emisión de recibos y distribución de insumos médicos en laboratorios clínicos de la Carrera de Odontología de la Universidad Pública de El Alto?

1.3.2. Problemas Secundarios

- Deficiencias en el proceso de emisión de recibos de caja, lo cual genera tendencia a fallos en la coordinación de trabajos con otras unidades, fallas en la consolidación de datos financieros, imprecisiones en los reportes y riesgo de pérdida o alteración de información del negocio.
- Déficit de control en la distribución de insumos médicos a los estudiantes lo que genera riesgo de pérdida de insumos y la imposibilidad de justificar las salidas de inventario en un determinado tiempo.

1.4. Objetivos

1.4.1. Objetivo General

Desarrollar un sistema informático para mejorar la eficiencia en los procesos de emisión de recibos y distribución de insumos médicos en laboratorios clínicos de la Carrera de Odontología de la Universidad Pública de El Alto.

1.4.2. Objetivos Específicos

- Describir el fundamento teórico relacionado a los procesos de emisión de recibos, distribución de insumos médicos, y las metodologías o herramientas que permitan plantear una solución adecuada a los escenarios del negocio.
- Diseñar el sistema informático con base al Método de Ingeniería Web basado en el Lenguaje de Modelado Unificado que permita optimizar los procesos de

emisión de recibos y distribución de insumos médicos en laboratorios clínicos de la Carrera de Odontología de la UPEA.

- Desarrollar el sistema informático con base a métodos y herramientas de ingeniería del software para regular los procesos de negocio identificados.
- Realizar las pruebas técnicas al sistema informático para garantizar su correcto funcionamiento y calidad.

1.5. Justificación

1.5.1. Justificación Técnica

El sistema informático para emisión de recibos y distribución de insumos médicos de laboratorios clínicos en Carrera de Odontología representa un avance tecnológico para las personas que trabajan en esta institución, pues en el estado actual de su gestión tienen que hacer bastante labor manual para realizar reportes y también existe el riesgo de cometer errores que prácticamente pueden ser subsanados por un sistema de información moderno.

Cabe mencionar, que para la implementación del sistema propuesto se cuenta con espacio en un Centro de Datos, dado que la institución pertenece a la UPEA, lo cual facilita su instalación y despliegue en instalaciones de la Carrera de Odontología.

1.5.2. Justificación Económica

La gestión ineficiente en el proceso emisión de recibos y control de distribución de insumos médicos en laboratorios clínicos de la Carrera de Odontología representa un riesgo para los recursos económicos de esta entidad perteneciente a la UPEA. El sistema informático propuesto permite controlar y regular los procedimientos relacionados a la emisión de recibos de venta y control de distribución de insumos médicos, así como la automatización de reportes para una gestión eficiente de recursos.

1.5.3. Justificación Social

El sistema informático propuesto para la emisión de recibos y control de distribución de insumos médicos permite al personal perteneciente a la clínica odontológica gestionar de manera eficiente los procedimientos y los recursos. Asimismo, permite beneficiar de manera indirecta a los estudiantes que realizan sus prácticas en la institución, como también a los pacientes que son los clientes del negocio.

1.6. Metodología

1.6.1. Método de Ingeniería Web basado en el Lenguaje de Modelado Unificado

El método de ingeniería utilizado para el modelado del sistema informático propuesto es el Método de Ingeniería Web basado en el Lenguaje de Modelado Unificado (UWE), el cual fue propuesto por Nora Koch, del Instituto de Informática de la Universidad de Múnich de Alemania, y consiste de un método que permite especificar de mejor manera una aplicación Web en base a un proceso que utiliza el Proceso Unificado de Desarrollo de Software, por lo cual el proceso es iterativo e incremental, y además mantiene una notación estándar basada en el uso del Lenguaje de Modelado Unificado (Nieves, Ucán & Menéndez, 2014; Rodríguez, 2009).

1.6.2. Kanban

El método utilizado para gestionar el presente proyecto es Kanban, el cual es un método empleado para la gestión del proceso de software que originalmente fue aplicado a los procesos de fabricación de la industria automotriz y en los últimos años ha ido ganando popularidad debido a que es muy fácil de utilizar, actualizar y asumir por parte del equipo de desarrollo, además destaca por tener una técnica de gestión de tareas muy visual que permite ver a golpe de vista el estado de los proyectos (IEBS, 2019; Kanbanize, 2020).

1.7. Herramientas

1.7.1. Lenguaje de programación PHP

PHP, acrónimo recursivo de PHP HyperText Preprocessor es un lenguaje interpretado de código abierto muy popular, el cual es especialmente adecuado para el desarrollo Web, y se distingue de otros lenguajes porque es ejecutado en el servidor para generar lenguaje HTML el cual es enviado e interpretado por el cliente, que por lo general es un navegador Web (PHP Group, 2020).

1.7.2. Sistema Gestor de Bases de Datos MySQL

MySQL es un sistema de gestión de base de datos relacional desarrollado bajo la filosofía de código abierto con base a un modelo cliente - servidor, considerado como el más popular del mundo ya que es utilizado en aplicaciones Web por múltiples empresas como ser: Facebook, Google, Youtube, entre otros (Oracle, 2020).

1.7.3. Servidor Web Apache

Apache es un servidor Web de código abierto que implementa el protocolo HTTP que permite procesar las peticiones realizadas por un cliente Web utilizando el puerto TCP 80, aunque también permite conexiones seguras a través del puerto TCP 443 (Apache, 2020).

1.8. Alcances y límites

1.8.1. Alcances

El sistema presenta características funcionales generales las cuales se describen a continuación:

- Gestionar el registro de formularios de historial por especialidades o asignaturas.
- Gestionar el registro de tratamientos clínicos.
- Gestionar el registro de estudiantes, docentes y pacientes.

- Gestionar la emisión de recibos de venta.
- Gestionar el control de inventario de insumos para atención médica.
- Gestionar la notificación de recibos para mejorar la coordinación de trabajo entre unidades de la institución.
- Generar los reportes diarios, semanales y mensuales para el área de caja e insumos.

1.8.2. Limites

El sistema presenta las siguientes limitantes, que son:

- No realiza facturación, solo efectúa la emisión de recibos.
- El sistema realiza el control de distribución de los insumos en ambientes de la Unidad de Insumos Médicos ubicada en cada piso de laboratorio clínico, no contempla la gestión de almacén central de insumos.
- Las terminales de acceso deben tener Internet para poder acceder al sistema.
- El sistema no debería ser instalado en una maquina personal, sino debe estar en un centro de datos donde el riesgo de pérdida de información sea menor.

1.9. Aportes

En laboratorio clínico de la Carrera de Odontología se utiliza los archivos Excel para gestionar los procesos de emisión de recibos y control de distribución de insumos médicos a los estudiantes practicantes de dicha carrera. Los riesgos de alterar o perder información están latentes mientras sigan usando estas herramientas de trabajo.

El sistema propuesto pretende reducir esa deficiencia a la hora de emitir reportes para la toma de decisiones, además de reducir los riesgos de seguridad de la información. Asimismo, el sistema permite la consolidación de los datos financieros para una gestión eficiente de recursos económicos.

Por otra parte, el sistema facilita el trabajo de las personas que laboran en la clínica odontológica, lo cual alivia su carga laboral.

CAPITULO 2 MARCO TEORICO

Resumen

En este capítulo se desarrolla el fundamento teórico asociado a la atención en odontología, procesos de emisión de recibos y control de distribución de insumos médicos. A su vez se describen los fundamentos metodológicos, técnicas y herramientas que permiten plantear y desarrollar un software adecuado y de calidad.

2.1. Odontología

Uno de los propósitos principales de la odontología es ayudar a las personas a alcanzar y conservar al máximo la salud oral durante toda la vida. Según Lagunas (s.f.) la odontología se puede conceptualizar como el estudio o la ciencia que se encarga del diagnóstico, tratamiento y prevención de enfermedades para evitar enfermedades del aparato estomatognático. Esta disciplina en la actualidad es una de las más requeridas, lo cual hace que el profesional de la salud bucal tenga la necesidad de conocer una multitud de capacidades durante su formación, y ofrecer altos estándares en sus servicios (Valero, 2017). Esta última afirmación precisamente es uno de los fines de la formación de profesionales en Clínica Odontológica de la UPEA.

2.1.1. Protocolo general de atención en odontología

Según entrevista realizada a la Dra. Ponce, Jefa del área clínica de laboratorios clínicos de la carrera de odontología de la UPEA, el protocolo de atención presencial de pacientes inicia con la recepción del paciente por parte del estudiante para su posterior valoración con el consentimiento del docente. Una vez realizada la valoración el paciente debe pagar por su tratamiento y el estudiante debe facilitar el pago del mismo ante cajas, y a la vez adquirir el historial clínico que corresponde a la especialidad correspondiente. En este punto, una vez adquiridos los comprobantes de caja e historial, el estudiante debe aproximarse al área de insumos, radiología o quirófano previa autorización del docente para recabar los

insumos médicos, o hacer uso del servicio respectivamente y así tratar al paciente de acuerdo a su diagnóstico. Este protocolo general se ilustra en la Figura 1.

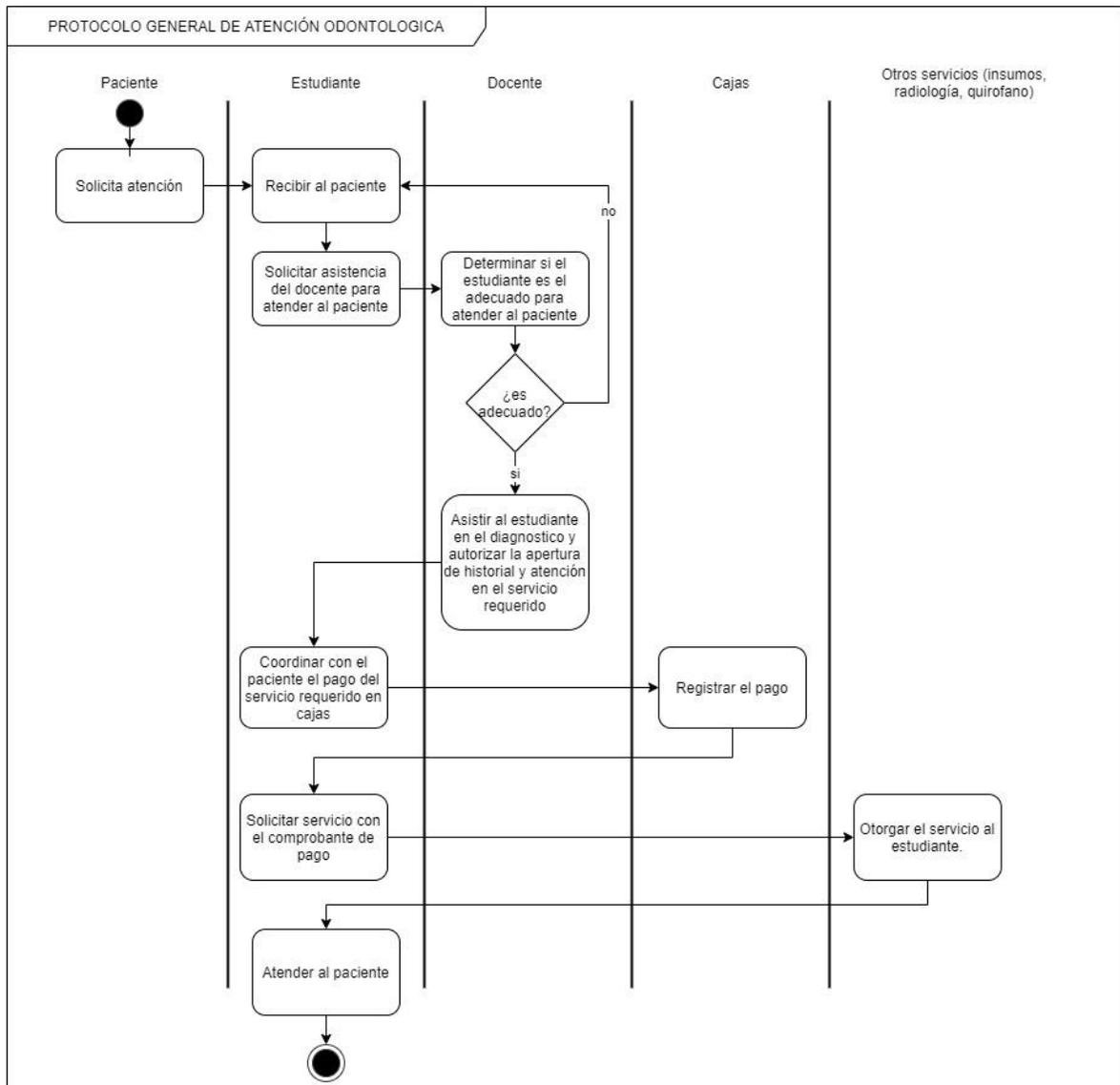


Figura 1. Protocolo General de Atención en Clínica Odontológica

A pesar de estar limitados los servicios de salud por la pandemia del COVID-19, la atención de los servicios odontología requieren ser reanudados con las precauciones que ameritan. Según el Colegio de Odontólogos de La Paz (2020), estas acciones de mitigación de propagación del COVID-19 empiezan desde la identificación de la patología mediante la telemedicina, etapa que determina si un paciente requiere la atención presencial o no.

En sí, el protocolo en Latinoamérica contempla la telemedicina (tele-odontología) como medio para iniciar la identificación de las urgencias o emergencias buco-dentales, y si el caso amerita la atención se extiende a la atención presencial con todas las medidas de bioseguridad establecidas para reducir el riesgo de contagio (Asociación Latinoamericana de Odontopediatría, 2020).

2.2. Comprobante de pago

El comprobante de pago, comprobante financiero o documento comercial son todos los comprobantes extendidos de manera digital o por escrito en los que se deja constancia de las operaciones que se realizan en la actividad mercantil, de acuerdo con los usos y costumbres generalizadas y las disposiciones de la ley; éste acredita la compra y venta de un producto o la prestación de un servicio (Caurin, 2016; Escolme, 2012).

Asimismo, Caurin (2016) afirma que el comprobante de pago posee un mínimo de información, como ser: tipo de comprobante, las personas que intervienen, la fecha, el lugar, el concepto y el monto.

2.2.1. Tipos de comprobante de pago

Los comprobantes de pago se clasifican con base a distintos criterios. Desde el punto de vista de quien recibe o emite la documentación los comprobantes se dividen en: externos, es decir emitidos fuera de la empresa o entidad que son recibidos para ser conservados; y los internos, que son los documentos emitidos por la empresa o entidad (Escolme, 2012).

Desde el punto de vista de la documentación que se registra y archiva, los comprobantes de pago se clasifican en: factura, nota de débito, nota de crédito, ticket, recibo, pagaré, cheque, nota de crédito bancario, orden de compra, nota de venta, resumen de cuenta, entre otros (Escolme, 2012). En particular, el recibo es una constancia de pago o de haber recibido dinero, su firma puesta al pie es la prueba de una transacción efectiva, con importancia legal y contable. La información mínima que posee el recibo se resume en: lugar y fecha en que se otorga, nombre

de la persona que recibe, la cantidad de dinero, el concepto y la firma de quien recibe.

2.3. Distribución de insumos

2.3.1. Distribución física

La distribución física es el término utilizado para describir las actividades relativas al movimiento de la cantidad correcta de los productos adecuados al lugar preciso en el momento exacto (Actualidad Empresa, 2016). Un sistema de distribución eficaz será aquel que tenga a disposición del mercado los productos que éste demande, en la cantidad precisa y en momento oportuno.

2.3.2. Control de inventario

Los inventarios o stocks son la cantidad de bienes que una empresa o entidad mantiene en existencia en un momento dado, los cuales están destinados para la venta ordinaria o para ser consumidos en la producción de bienes o servicios (Durán, 2012). Los inventarios existen por múltiples razones, como por ejemplo: prevén la escasez, permiten obtener ganancias adicionales cuando hay alzas de precios, entre otros.

La gestión de inventarios se refiere al manejo y control adecuado de las existencias de los diferentes productos y tiene por objeto determinar la cantidad o nivel adecuado de inventario que debería mantenerse de tal forma que se pueda lograr satisfacer la demanda (Arias, 2015). Para este fin, se debe tomar en cuenta el cálculo de ciertos valores, como ser: nivel mínimo de existencia, nivel máximo de existencia, nivel teórico de consumo, nivel de reorden, pedido óptimo e intervalo de revisión.

Por otra parte, el control de inventario se refiere a los procesos que coadyuvan al suministro, accesibilidad y almacenamiento de productos en alguna compañía para minimizar los tiempos y costos relacionados con el manejo del mismo (HIPODEC,

2018). En otras palabras, es un mecanismo a través del cual se administra de manera eficiente el movimiento y almacenamiento.

2.3.3. Métodos de control de inventario

Para el control del inventario se tiene diversos métodos de control, entre los cuales se puede describir:

a) Método ABC, conocido como método 80/20, que consiste en dividir los productos en tres categorías según su importancia, cantidad y valor, así es más fácil identificar los productos más valiosos (ESERP, 2020).

b) Método PEPS, conocido como el método que sigue el criterio de primero en entrar primero en salir, consiste en identificar los primeros artículos en entrar a almacén para que sean los primeros en salir a la venta o ser distribuidos y utilizados en la producción o provisión de un servicio (ESERP, 2020).

c) Método EOQ o de cantidad económica de pedido, se utiliza cuando la empresa tiene una demanda y frecuencia de uso de inventario constantes en el tiempo (ESERP, 2020).

2.3.4. Insumos médicos

Los insumos médicos se constituyen en diferentes sustancias, artículos, material e instrumentos que son utilizados por los especialistas de la salud, tanto en consultorios como en cirugías, para el diagnóstico, tratamiento o prevención de enfermedades que para su uso no requieren uso de fuentes de energía; estos otorgan soluciones a los problemas que se le presentan a sus pacientes (DIGEMID, 2009; Strenamedical, 2019).

2.4. Ingeniería de software

El software de computadora es el producto que construyen los programadores profesionales, éste incluye programas que se ejecutan en una computadora de

cualquier tamaño y arquitectura (Pressman, 2010). Los productos software se desarrollan de acuerdo a las necesidades o requerimientos de un cliente.

La ingeniería del software es la aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento de software, es decir, comprende todos los aspectos de la producción de software desde las etapas iniciales de la especificación del sistema, hasta el mantenimiento de éste después de que se utiliza (Pressman, 2010; Sommerville, 2005). La ingeniería de software es una tecnología con varias capas (Figura 2), como ser: el compromiso organizacional con la calidad el cual la base; el proceso de ingeniería que es el aglutinante que une las capas de tecnología y permite el desarrollo racional y oportuno del software dado que define una estructura que debe establecerse para una obtención eficaz del producto; los métodos de ingeniería que proporcionan la experiencia técnica para elaborar software, los cuales incluyen un conjunto amplio de tareas como análisis de requerimientos, modelado de diseño, construcción del programa y pruebas; y las herramientas de la ingeniería que proporcionan un apoyo para el proceso.



Figura 2. Capas de la ingeniería de software.
Fuente: (Pressman, 2010)

2.4.1. Sistema informático

La información representa un conjunto de datos transformados de forma que contribuye a reducir la incertidumbre del futuro y, por tanto, ayuda la toma de decisiones, a diferencia de los datos, que son símbolos que representan valores de atributos o sucesos (Lapiedra, Devece & Guiral, 2011). En este sentido, la relación entre los datos y la información es equivalente a la que existe entre la materia prima

y el producto acabado. Una representación de este concepto se presenta en la Figura 3.



Figura 3. Proceso de transformación de datos en información
Fuente: (Lapiedra et al., 2011)

Asimismo, la información posee cualidades las cuales determinan su valor. Lapiedra et al. (2011) afirman que la información debe: ser relevante, ser exacta, ser completa, ser de confianza, comunicar con la persona correcta, ser puntual, ser detallada, ser comprensible. Por otra parte, un sistema de información es el conjunto formal de procesos que operando sobre una colección de datos estructurada de acuerdo con las necesidades de una organización, ésta recopila, elabora y distribuye información necesaria para la operación de dicha organización y para las actividades de dirección y control (Lapiedra et al., 2011). Un sistema de información no está precisamente ligado a la informática, ya que puede hacer uso de otras herramientas para su fin.

En este punto, resulta oportuno aclarar la diferencia entre sistema de información y sistema informático. Un sistema informático consiste en la compleja interconexión de numerosos componentes de hardware y software los cuales son básicamente sistemas deterministas y formales, de tal forma que dado un conjunto de entradas determinado siempre se obtiene un conjunto de salidas las cuales representan información necesaria para la continuidad de las actividades de una organización (Doña, s.f.; Lapiedra et al., 2011). Por tanto, un sistema informático es una herramienta utilizada para facilitar las tareas en un sistema de información.

2.4.2. Modelos de proceso de software

Un proceso de software es un conjunto de actividades, acciones, tareas y resultados asociados que producen un producto software, cuya estructura establece el fundamento para el proceso completo de la ingeniería de software (Pressman, 2010; Sommerville, 2005). Las actividades estructurales se dividen en: comunicación,

planeación, modelado, construcción y despliegue, las cuales también se conocen como ciclo de vida del software, y también están definidas por un flujo de proceso, como ser: lineal, iterativo, evolutivo y en paralelo, cada uno de acuerdo a un modelo de proceso.

Los modelos de proceso tienen la finalidad de poner orden en el caos del desarrollo del software y se basan en algún enfoque o paradigma, como ser: enfoque en cascada, desarrollo iterativo, y enfoque basado en componentes (Pressman, 2010; Sommerville, 2005). Todos los modelos de proceso del software pueden incluir las actividades estructurales, éstos se dividen en: modelo de proceso en cascada, modelos de proceso incremental, modelos de proceso evolutivo, modelos concurrentes, entre otros.

a) Modelo de la cascada, se aplica por lo general cuando los requerimientos se comprenden bien desde la comunicación hasta el despliegue; y sigue un enfoque sistemático y secuencial para el desarrollo del software (Pressman, 2010). Una ilustración del modelo se presenta en la Figura 4.

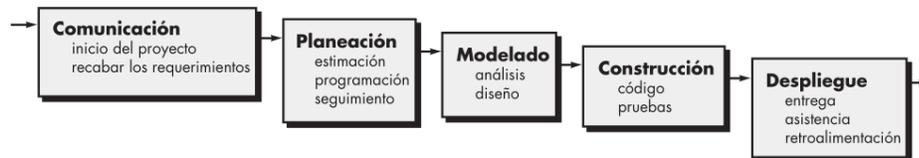


Figura 4. Representación del modelo de proceso de la cascada.
Fuente: (Pressman, 2011).

b) Modelo de proceso incremental, se aplica por lo general cuando los requerimientos iniciales del software están razonablemente bien definidos, pero el alcance general del esfuerzo de desarrollo imposibilita un proceso lineal, donde se dé la necesidad imperiosa de dar rápidamente cierta funcionalidad limitada de software a los usuarios y aumentarla en entregas posteriores (Pressman, 2010). Este modelo combina el flujo de proceso lineal para cada incremento, y el flujo paralelo entre cada incremento de ser necesario.

c) Modelos de proceso evolutivo, se aplican por lo general cuando los requerimientos del negocio y del producto cambian conforme avanza el desarrollo, por tanto se requieren iteraciones que permitan desarrollar versiones cada vez mas complejas del software (Pressman, 2010). Entre los modelos que siguen un proceso evolutivo se puede mencionar al modelo del prototipado y el modelo en espiral.

d) Modelos concurrentes, también llamada ingeniería concurrente, permite que un equipo de software represente elementos iterativos y concurrentes de cualquiera de los modelos de proceso anteriormente descritos (Pressman, 2010).

2.4.4. Desarrollo ágil

Según Pressman (2010), cualquier proceso del software se considera ágil si aborda ciertas suposiciones clave acerca de la mayoría del software, como ser:

- a) Es difícil determinar los requerimientos que persistirán y cuales cambiarán.
- b) Para muchos tipos de software, el diseño y la construcción deben ejecutarse de manera paralela, de modo que los modelos de diseño se prueban a medida que se crean.
- c) El análisis, el diseño, la construcción y las pruebas no son tan predecibles como nos gustaría.

Dado que se presentan escenarios impredecibles, la adaptabilidad es una característica clave para que un proceso de considere ágil. Según Pressman (2010), se deben considerar doce principios propuestos por la Alianza Ágil para alcanzar la agilidad en un proceso, estos son:

1. La prioridad más alta es satisfacer al cliente a través de la entrega pronta y continua de software valioso.
2. Son bienvenidos los requerimientos cambiantes, aún en una etapa avanzada del desarrollo.
3. Entregar con frecuencia software que funcione, de dos semanas a un par de meses.

4. Las personas de negocios y los desarrolladores deben trabajar juntos, a diario y durante todo el proyecto.
5. Hay que desarrollar los proyectos con individuos motivados.
6. El método más eficiente y eficaz para transmitir información a los integrantes de un equipo de desarrollo es la conversación cara a cara.
7. La medida principal de avance es el software que funciona.
8. Los procesos ágiles promueven el desarrollo sostenible.
9. La atención continua a la excelencia técnica y el buen diseño mejora la agilidad.
10. Es esencial la simplicidad.
11. Las mejores arquitecturas, requerimientos y diseños surgen de los equipos con organización propia.
12. El equipo reflexiona a intervalos regulares sobre cómo ser eficaz para después afinar y ajustar su comportamiento.

Asimismo, este autor afirma que no todo método de desarrollo ágil aplica estos 12 principios con igual intensidad.

Entre los métodos de desarrollo ágil que aplican los principios descritos anteriormente, se tienen, por ejemplo: la programación extrema (XP), SCRUM, el desarrollo adaptativo de software (DAS), KANBAN, método de desarrollo de sistemas dinámicos (MDSD), Cristal, desarrollo impulsado por las características (DIC), desarrollo esbelto de software (DES), proceso unificado ágil (AUP), entre otros (Pressman, 2010).

2.4.5. Método Kanban

Kanban es un método empleado para la gestión del proceso de software que originalmente fue aplicado a los procesos de fabricación de la industria automotriz y en los últimos años ha ido ganando popularidad debido a que es muy fácil de utilizar, actualizar y asumir por parte del equipo de desarrollo, además destaca por tener una técnica de gestión de tareas muy visual que permite ver a golpe de vista el estado de los proyectos (IEBS, 2019; Kanbanize, 2020).

David J. Anderson, reconocido líder de la adopción del agilismo formuló el método Kanban como una aproximación al proceso evolutivo e incremental, enfocado en llevar a cabo las tareas pendientes con base a cuatro principios fundamentales y seis prácticas (Kanbanize, 2020). Estos cuatro principios se describen brevemente a continuación.

- Principio 1: empezar con que hace ahora, se refiere a que Kanban no requiere configuración y puede ser aplicado a flujos reales de trabajo o procesos activos para identificar los problemas.
- Principio 2: comprometerse a buscar e implementar cambios incrementales y evolutivos, se refiere a que Kanban está diseñado para implementarse con una mínima resistencia por lo que trata de pequeños y continuos cambios incrementales y evolutivos del proceso actual.
- Principio 3: respetar los procesos, las responsabilidades y los cargos actuales, se refiere a que Kanban reconoce que los procesos en curso, los roles, las responsabilidades y los cargos existentes pueden tener valor y vale la pena conservarlos.
- Principio 4: animar el liderazgo en todos los niveles, se refiere a que Kanban promueve los liderazgos que surgen de actos día a día de las personas que están en el equipo.

Por otra parte, las prácticas de Kanban se dividen en seis, las cuales son: visualizar el flujo de trabajo, eliminar las interrupciones, gestionar el flujo, hacer las políticas explícitas, circuitos de retroalimentación y mejora con trabajo colaborativo (Kanbanize, 2020).

- Visualizar el flujo de trabajo, permite entender qué se necesita para el transcurso de un producto desde su pedido hasta su entrega, solo después de entender cómo funciona el actual flujo se puede aspirar a mejorarlo. Para visualizar el proceso en Kanban se hace uso de un tablero de tarjetas y columnas, donde cada columna representa un paso en el flujo, y una tarjeta Kanban representa un elemento de trabajo (Kanbanize, 2020). Cuando se comienza a trabajar en un elemento determinado se lo arrastra hasta la columna Pendiente, y cuando

está concluido se lo mueve hasta la columna Hecho. Una ilustración del tablero Kanban se muestra en la Figura 5.

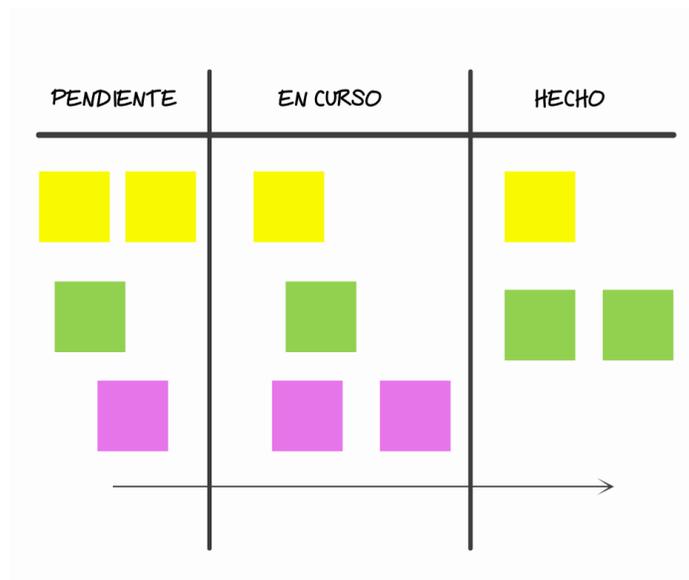


Figura 5. Tablero Kanban.
Fuente: Elaborado con base a (Kanbanize, 2020).

- Eliminar las interrupciones, hace referencia a que un cambio de enfoque puede dañar seriamente el proceso y las tareas (Kanbanize, 2020). Por tanto se deben establecer los límites del trabajo en el proceso.
- Gestionar el flujo, hace referencia a que se debe crear un flujo continuo e ininterrumpido de elementos de trabajo o tareas a través del proceso de producción (Kanbanize, 2020).
- Hacer las políticas explícitas, se refiere a que no se puede mejorar algo que no se entiende, por lo que el proceso debe estar bien definido, publicado y promovido para que todos estén familiarizados con los objetivos (Kanbanize, 2020).
- Circuitos de retroalimentación, hace referencia a que se deben efectuar reuniones regulares para sincronizar el trabajo del equipo y reuniones de revisión de entregas con los interesados finales (Kanbanize, 2020). La duración promedio de reunión debe oscilar los 10 a 15 minutos.

- Mejorar colaborando, hace referencia a una mejora continua y sostenible del conocimiento sobre el trabajo, flujo de trabajo, el proceso y riesgos, mediante una comunicación constante del equipo (Kanbanize, 2020).

El método Kanban está especialmente indicado para aquellas organizaciones que requieran de flexibilidad especialmente en la entrada de tareas, así como el seguimiento, la priorización y la supervisión del equipo de trabajo (IEBS, 2019). Éste método es fácil de adoptar en una organización, los tableros Kanban ayudan a visualizar el trabajo, y los límites del trabajo permiten ser más eficiente (Kanbanize, 2020).

2.4.6. Metodología de ingeniería Web basado en el Lenguaje de Modelado Unificado

La ingeniería Web es una versión adaptada del enfoque de ingeniería de software que propone una estructura ágil, pero disciplinada para construir sistemas y aplicaciones basados en Web con calidad Industrial (Pressman, 2010).

El Método de Ingeniería Web basado en el Lenguaje de Modelado Unificado (UWE), fue propuesto por Nora Koch, del Instituto de Informática de la Universidad de Múnich de Alemania, y consiste de un método que permite especificar de mejor manera una aplicación Web en base a un proceso que utiliza el Proceso Unificado de Desarrollo de Software, por lo cual el proceso es iterativo e incremental, y además mantiene una notación estándar basada en el uso del Lenguaje de Modelado Unificado (Rodríguez, 2009) (Nieves, Ucán & Menéndez, 2014). Este método a su vez tiene como fundamento el Desarrollo Dirigido por Modelos (MDD 16), y consta de cinco fases o modelos, los cuales se ilustran en la Figura 6, y son: a) el análisis de requerimientos, b) modelo de contenido, c) modelo de navegación, d) el modelo de presentación, y el modelo de proceso.

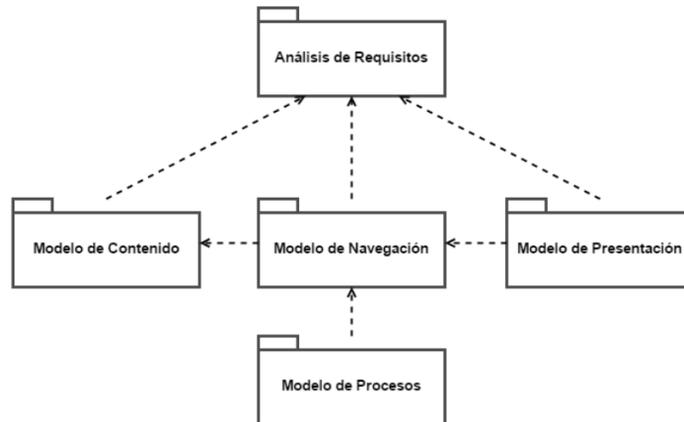


Figura 6. Modelos de UWE.
Fuente: (Nieves et al., 2014).

a) Análisis de requisitos, consiste en la especificación de los casos de uso del sistema, y plasma los requisitos funcionales de una aplicación Web (Rodríguez, 2009) (Nieves et al., 2014).

b) Modelo de contenido, o diseño conceptual, es la etapa donde se representa el dominio del problema con un diagrama de clases de UML, definiendo a detalle los conceptos involucrados en la aplicación (Rodríguez, 2009) (Nieves et al., 2014).

c) Modelo de navegación, representa la navegación de los objetos dentro de la aplicación y conjunto de estructuras como son: índices, menús y consultas, cuya notación se ilustra en la Figura 7. Además comprende dos etapas, 1) la definición del espacio de navegación, y 2) el diseño de las estructuras de navegación (Rodríguez, 2009) (Nieves et al., 2014).

- La definición del espacio de navegación se trata de una vista del diagrama conceptual, y se define mediante un diagrama de clases en UML, y se distinguen con el estereotipo NC.
- El diseño de las estructuras de navegación, establece las estructuras de acceso que permiten visitar los objetos del espacio navegacional, y están constituidas por menús, índices, visitas guiadas y formularios.

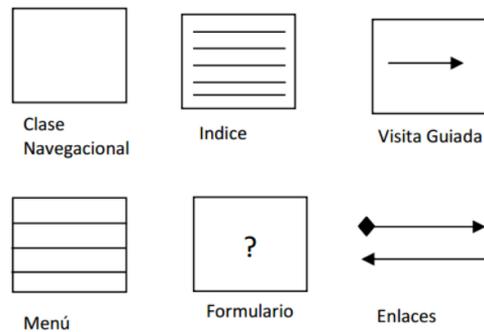


Figura 7. Elementos de la notación en UWE.
Fuente: (Rodríguez, 2009).

d) Modelo de Presentación, representa las interfaces de usuario por medio de vistas abstractas o clases de presentación, y está relacionado con los elementos de las interfaces definidas en HTML, además están definidos como estereotipos UML y cada clase del modelo navegacional tiene asociados un conjunto de clase del modelo de presentación. Los elementos del modelo de presentación son: anclas, entradas de texto, imágenes, audio y botones (Rodríguez, 2009) (Nieves et al., 2014).

e) Modelo de Proceso, representa el aspecto que tienen las actividades que se conectan con cada clase de proceso, y contempla dos tipos de modelos, que son: modelo de estructura de proceso que describe las relaciones entre las distintas clases de proceso, y el modelo del flujo de proceso que especifica las actividades conectadas con cada clase de proceso (Rodríguez, 2009).

Por tanto, con base a la etapa de análisis de requerimientos y el modelado de contenido se obtiene el modelo de navegación; luego con base al modelo de navegación se obtiene el modelo de presentación; y por ultimo con base a todos los modelos anteriores es posible extender el modelo del proceso, que todos en su conjunto representan la funcionalidad completa del software.

2.4.7. Modelos de arquitectura de software

Un modelo de arquitectura representa la estrategia base para estructurar un sistema informático. El diseño arquitectónico representa la estructura de los datos y los

componentes del software que se requieren para construir un sistema basado en computadora (Pressman, 2010).

Sommerville (2005) presenta tres de los modelos arquitectónicos más usados en los proyectos de software, que son: el modelo de repositorio, el modelo cliente servidor y el modelo de capas.

a) El modelo de repositorio, donde los subsistemas que forman un sistema deben intercambiar datos para que puedan trabajar conjuntamente de manera efectiva (Sommerville, 2005). Es decir que los datos se almacenan en una base de datos central a la que pueden acceder todos los subsistemas. Este modelo es adecuado para aplicaciones en las que los datos son generados por un subsistema y son usados por otro.

b) El modelo cliente – servidor permite organizar un sistema en conjuntos de servicios y servidores asociados más unos clientes que acceden y usan los servicios (Sommerville, 2005). Los principales elementos de este modelo son: los servidores que ofrecen servicios a otros subsistemas o a los clientes; un conjunto de clientes que usan los servicios ofrecidos por los servidores; y una red que permite que los clientes accedan a los servicios. Esto último no es estrictamente necesario ya que los clientes y los servidores podrían ejecutarse en una única máquina.

c) El modelo de capas permite organizar el sistema en capas, donde cada una proporciona un conjunto de servicios (Sommerville, 2005). Cada capa representaría una máquina abstracta o subsistema que ofrece servicios a otras capas.

2.4.8. Herramientas para desarrollo Web

2.4.8.1. Lenguaje de programación Web PHP

Un lenguaje de programación es una representación formal que permite a un programador darle instrucciones a una computadora mediante una sintaxis y un conjunto de acciones que son ejecutadas en una computadora (Pressman, 2010). Existen lenguajes de programación para construir todo tipo de software, y

básicamente se clasifican en dos tipos: lenguajes de programación de bajo nivel, cuya sintaxis es lejana al lenguaje humano, como ser el lenguaje máquina y lenguaje ensamblador; y los lenguajes de programación de alto nivel, cuya sintaxis es cercana al lenguaje humano, dado que las instrucciones están definidas en algún idioma como el inglés (Rock, 2019). Los lenguajes de programación de alto nivel requieren ser traducidos a lenguaje máquina mediante intérpretes o compiladores.

PHP, acrónimo recursivo de PHP HyperText Preprocessor es un lenguaje interpretado de código abierto muy popular, el cual es especialmente adecuado para el desarrollo Web, y se distingue de otros lenguajes porque es ejecutado en el servidor para generar lenguaje HTML el cual es enviado e interpretado por el cliente, que por lo general es un navegador Web (PHP Group, 2020). Es un lenguaje procedimental, pero en su versión actual PHP 7 permite también la programación orientada a objetos.

2.4.8.2. Sistema de gestión de base de datos MySQL

MySQL es un sistema de gestión de base de datos relacional desarrollado bajo la filosofía de código abierto con base a un modelo cliente - servidor, considerado como el más popular del mundo ya que es utilizado en aplicaciones Web por múltiples empresas como ser: Facebook, Google, Youtube, entre otros (Oracle, 2020). Esta herramienta permite crear y administrar bases de datos de manera flexible, segura y con un alto rendimiento. Al estar basado en un modelo cliente servidor, el servidor MySQL es el encargado de procesar las consultas y peticiones de un cliente, este último puede ser una aplicación Web u otro software con una interfaz que permita comunicarse con el servidor.

2.4.8.3. Servidor Web Apache

Apache es un servidor Web de código abierto que implementa el protocolo HTTP que permite procesar las peticiones realizadas por un cliente Web utilizando el puerto TCP 80, aunque también permite conexiones seguras a través del puerto TCP 443 (Apache, 2020). Es un servidor fácil de configurar a comparación de otros

programas de similar propósito, funciona por lo general en plataformas Microsoft y GNU/Linux.

2.4.9. Pruebas técnicas para aplicaciones Web

Una prueba técnica para aplicaciones Web es una colección de actividades relacionadas con una sola meta: descubrir errores de contenido, función, utilidad, navegabilidad, rendimiento, capacidad y seguridad, donde participan los ingenieros Web, gestores, clientes y usuarios finales con un único propósito, asegurar la calidad del software (Pressman, 2010).

El proceso de prueba de una aplicación Web inicia enfocándose en los aspectos visibles para el usuario y avanza hacia pruebas que ejercitan la tecnología y la infraestructura. Según Pressman (2010), se realizan siete pasos, que son: prueba de contenido, prueba de interfaz, prueba de navegación, prueba de componente, prueba de configuración, prueba de rendimiento y prueba de seguridad.

Las pruebas técnicas de aplicaciones Web permiten valorar el software de acuerdo a 9 dimensiones, como ser: el contenido, la función, la estructura, la usabilidad, la navegabilidad, el rendimiento, la compatibilidad, la interoperabilidad y la seguridad (Pressman, 2010).

2.4.9.1. Estrategia de pruebas para aplicaciones Web

Según Pressman (2010) la estrategia general para probar aplicaciones Web aplica estrategias y tácticas de sistemas orientados a objetos, que se resumen en:

- El modelo de contenido se revisa a fin de descubrir errores.
- El modelo de interfaz se examina para garantizar que todos los casos de uso se cumplen.
- El modelo de diseño se revisa para descubrir errores de navegación.
- La interfaz de usuario se prueba para descubrir errores en la mecánica de presentación y navegación.
- Los componentes funcionales se someten a pruebas de unidad.

- Se prueba la navegación a lo largo de toda la arquitectura.
- Se prueban distintas instalaciones del software en diferentes entornos para asegurar la compatibilidad con cada configuración.
- Las pruebas de seguridad se realizan con la intención de explotar las vulnerabilidades.
- Se realizan pruebas de rendimiento.
- Se prueba la aplicación Web con una población controlada y monitoreada de usuarios finales para una evaluación del contenido, navegación, usabilidad, compatibilidad, seguridad, confiabilidad y rendimiento.

Una ilustración del proceso de prueba se muestra en la Figura 8.

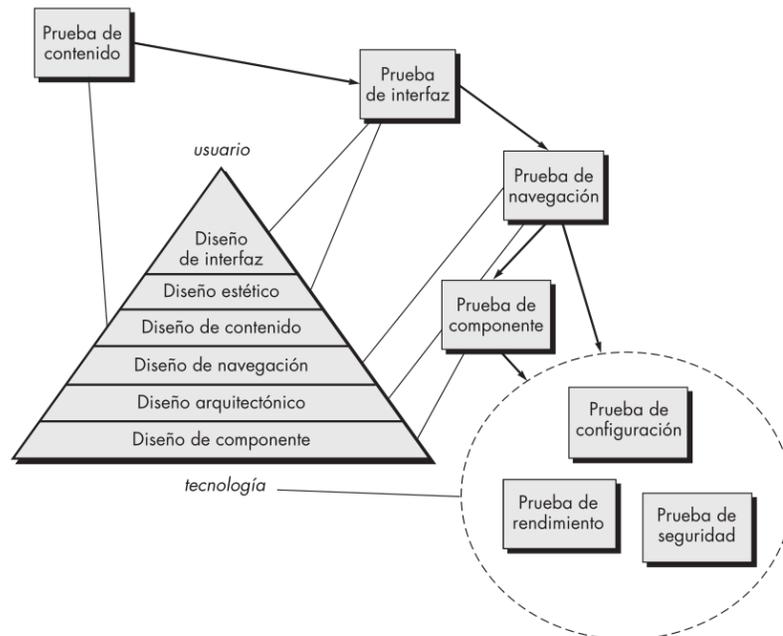


Figura 8. Representación del proceso de pruebas de aplicaciones Web.
Fuente: (Pressman, 2010).

2.4.10. Métricas de producto software

Una medida proporciona un indicio cuantitativo de algún atributo de un producto o proceso, la medición es el acto de tomar una medida, y la métrica es una medida cuantitativa del grado en el que un sistema, componente o proceso posee un atributo determinado (Pressman, 2010). A su vez, un indicador es una métrica o combinación de métricas que proporcionan comprensión acerca del producto en sí.

Según Pressman (2010), las métricas de producto ayudan a los ingenieros de software a obtener comprensión acerca del diseño y la construcción del software que elaboran, al enfocarse en atributos medibles específicos de los productos de trabajo de ingeniería del software. Asimismo, este autor indica que el primer paso en el proceso de medición es determinar las mediciones y métricas del software que sean adecuadas para la presentación del software que se está construyendo. En este sentido, existen métricas para distintos escenarios, como ser:

- Métricas para el modelado de requerimientos, como ser: la métrica basada de punto función o la métrica de punto de casos de uso.
- Métricas de diseño de aplicaciones Web, como ser: métricas de interfaz, métricas estéticas, métricas de contenido y métricas de navegación.
- Métricas para código fuente, como el propuesto en la Halstead para determinar el volumen del código del software.

2.4.11. Método de estimación de costos basado en puntos de casos de uso

Los métodos de estimación requieren como información el esfuerzo necesario para desarrollar el producto, que por lo general depende del tamaño del software a entregar, y para esto se utilizan medidas directas como líneas de código fuente en el caso de COCOMO, o medidas indirectas como en Puntos de Función que se determinan a partir de las funcionalidades del software (Lencina, Medina & Dapozo, 2016).

El método de estimación de puntos de casos de uso consta de cuatro etapas, en las que se desarrollan cálculos, como ser: factor de peso de los actores sin ajustar, factor de peso de los casos de uso sin ajustar, puntos de casos de uso ajustados y esfuerzo de horas hombre (Diaz & Robiolo, 2014; Lencina et al., 2016).

a) Para determinar el factor de peso de los actores sin ajustar (UAW), se debe identificar la complejidad de interacción de los actores de acuerdo a tres categorías, que son: simple, que representa a otro software con una interfaz definida cuya

ponderación es igual a uno; medio, que representa la interacción con otro sistema que interactúa a través de un protocolo de comunicaciones al cual se le asigna una ponderación de dos; y complejo, que representa la interacción de un humano realizada a través de interfaz gráfica cuya ponderación es de tres (Diaz & Robiolo, 2014; Lencina et al., 2016). Al finalizar se debe realizar una sumatoria del producto de la cantidad de actores por categoría.

b) Para determinar el factor de peso de los casos de uso sin ajustar (UUCW), se debe identificar la complejidad de interacción de los casos de uso de acuerdo a tres categorías las cuales están basadas en el número de transacciones, que son: simple, de tres a menos transacciones por caso de uso; medio, de cuatro a siete transacciones por caso de uso; y complejo, de más de siete transacciones por caso de uso (Diaz & Robiolo, 2014; Lencina et al., 2016). Al finalizar se debe realizar una sumatoria del producto de la cantidad de casos de uso por categoría.

En este punto, el valor de los puntos de casos de uso sin ajustar (UUCP) resulta de la suma de los últimos dos valores obtenidos, es decir $UUCP = UAW + UUCW$.

c) Los puntos de casos de uso ajustados, que se obtiene como resultado de multiplicar los puntos de casos de uso sin ajustar por los factores de complejidad técnica (TCP) y los factores ambientales (EF), es decir $UCP = UUCP \times TFC \times EF$.

Los factores de complejidad técnica evalúan trece características del software, que son: sistema distribuido, objetivos de performance o tiempo de respuesta, eficiencia del usuario final, procesamiento interno complejo, código reutilizable, facilidad de instalación, facilidad de uso, portabilidad, facilidad de cambio, concurrencia, objetivos especiales de seguridad, provision de acceso directo a terceros y facilidades especiales de entrenamiento del usuario (Diaz & Robiolo, 2014). Estas características se valoran en una escala de 0 a 2 como irrelevante, 3 a 4 como medio y 5 en caso de ser esencial.

Los factores ambientales se evalúan sobre ocho características relacionadas con las habilidades y experiencia de los involucrados en el proyecto, que son: familiaridad con el modelo de proyecto utilizado, experiencia en la aplicación,

experiencia en orientación a objetos, capacidad del analista líder, motivación, estabilidad de los requerimientos, personal de tiempo parcial y dificultad del lenguaje de programación (Díaz & Robiolo, 2014). Estas características se deben valorar en una escala de 0 como no influyente a 5 como esencial.

d) Finalmente, el esfuerzo de horas hombre se obtiene inicialmente de contar la cantidad de los primeros seis factores ambientales que tienen puntuación menor a tres, y también contar la cantidad de los últimos factores que tienen puntuación mayor a tres (Díaz & Robiolo, 2014). Al finalizar la suma de ambos valores debe evaluarse en la Tabla 1 para determinar la cantidad de horas – persona (CF).

Tabla 1. Cantidad de horas – persona según el valor calculado en los factores ambientales.
Fuente: (Díaz & Robiolo, 2014; Karner, 1993).

Horas-Persona (CF)	Descripción
20	Si el valor es ≤ 2
28	Si el valor es ≤ 4
36	Si el valor es ≥ 5

Por último, el esfuerzo estimado de horas – persona (E), resulta de multiplicar el valor obtenido para los puntos de casos de uso ajustado y las horas – persona.

CAPITULO 3 MARCO APLICATIVO

Resumen

En este capítulo se especifican los requerimientos del software y se define el diseño de los componentes del software con base a la metodología de ingeniería Web basado en el lenguaje de modelado unificado. Asimismo, se describe la construcción del software y las pruebas técnicas respectivas aplicadas.

3.1. Análisis de la situación actual

Realizadas las reuniones y entrevistas informativas para conocer la situación actual en torno a los procesos de emisión de recibos y control de distribución de insumos en laboratorios clínicos de la carrera de Odontología de la UPEA, se pudo identificar un conjunto de procedimientos.

En primera instancia, se pudo identificar el protocolo general de atención odontológica, el cual se describe en el primer capítulo del presente documento, y cuya representación gráfica se observa en la Figura 1. El protocolo muestra de manera sucinta la forma en la cual se procede a atender a los pacientes.

Por otra parte, se ha identificado procedimientos, como ser: registro de tratamientos clínicos (Figura 9), registro de formularios de historial (Figura 10), registro de estudiantes (Figura 11), gestión de venta y emisión de recibos por concepto de tratamientos y formularios de historial clínico (Figura 12 y 13), y finalmente el reportaje diario semanal y mensual por parte de caja (Figura 14). La representación de estos procedimientos se presenta en notación UML con diagramas de actividades en carriles.

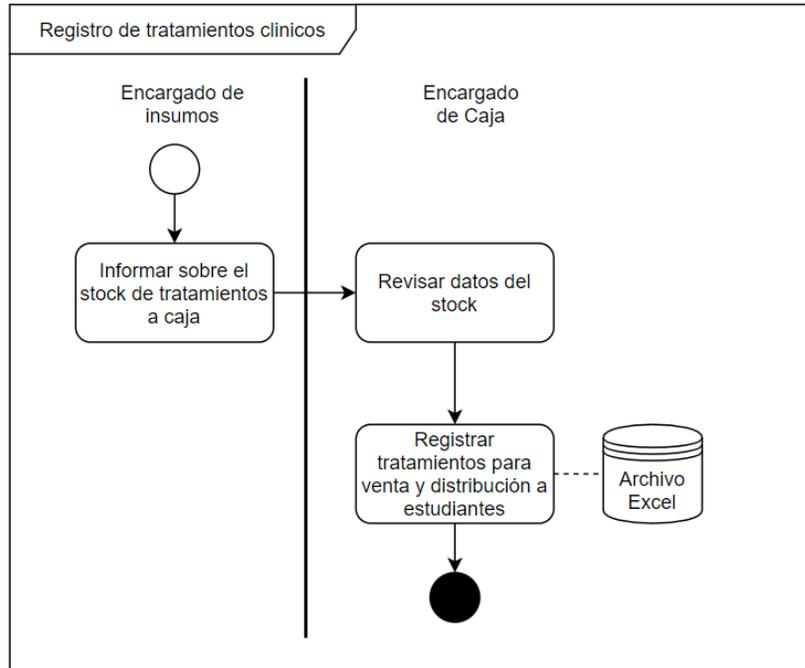


Figura 9. Procedimiento para el registro de tratamientos clínicos

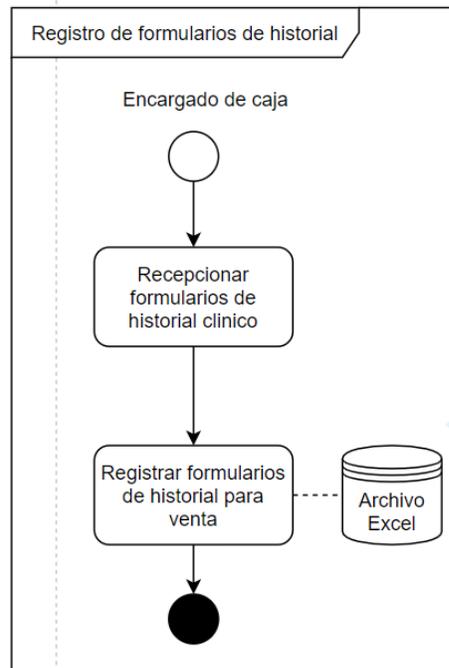


Figura 10. Procedimiento para el registro de formularios de historial clínico

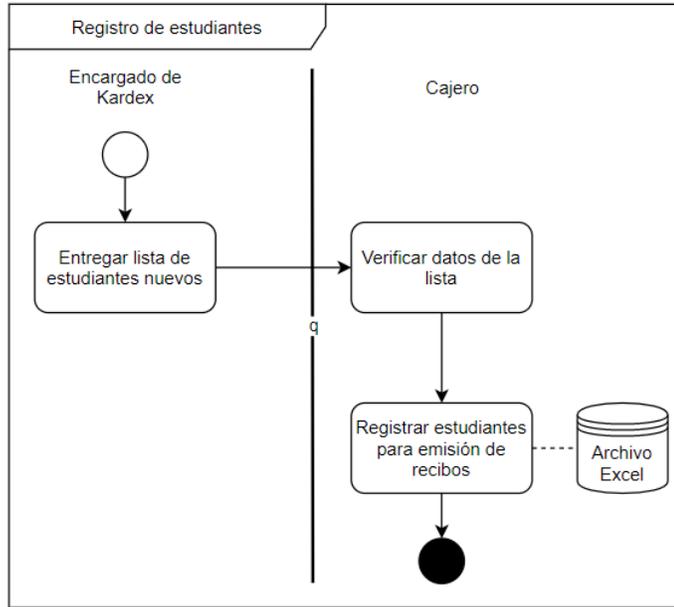


Figura 11. Procedimiento para el registro de estudiantes

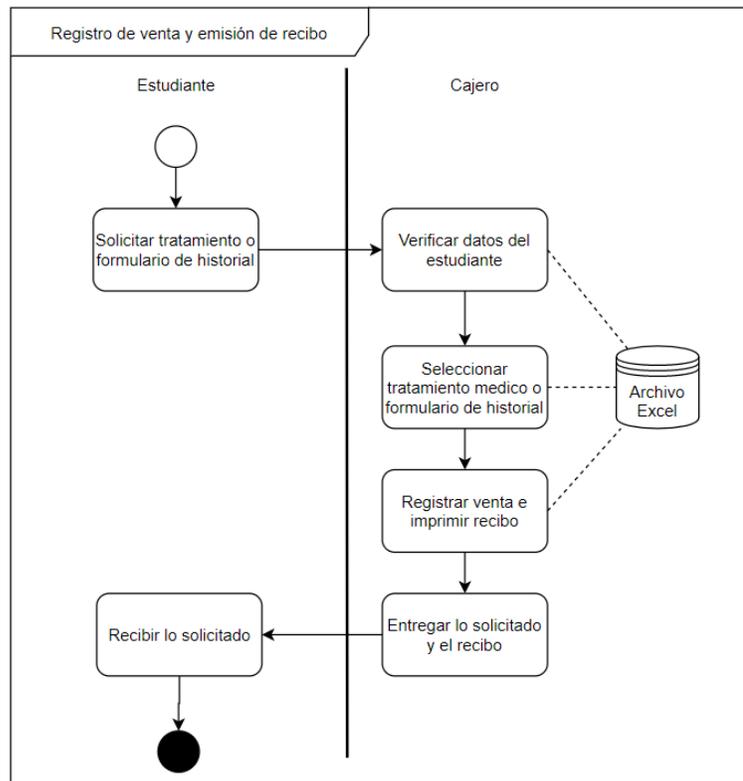


Figura 12. Procedimiento para el registro de venta y emisión de recibos

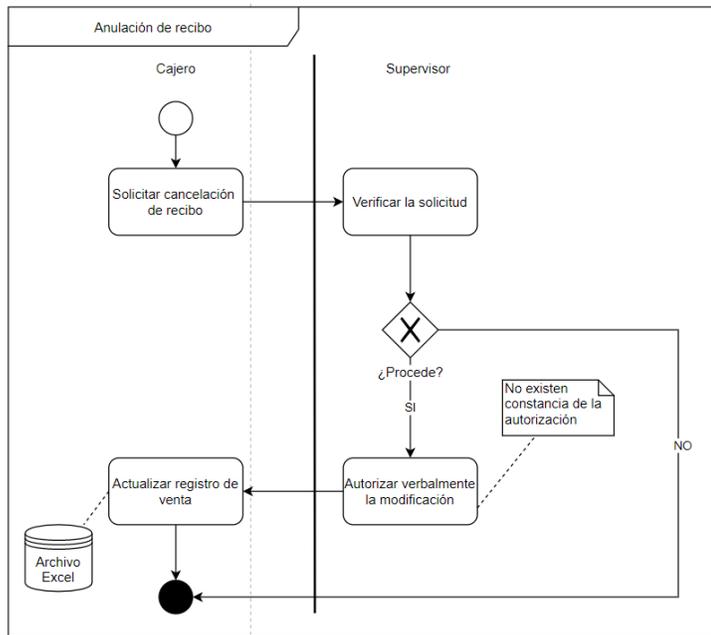


Figura 13. Procedimiento para la anulación de recibos de venta

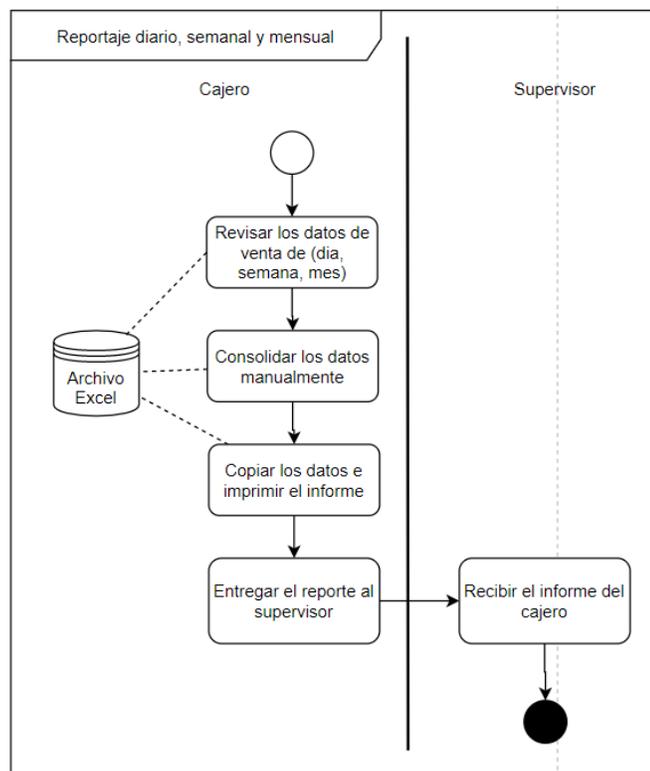


Figura 14. Procedimiento para emisión de reporte diario, semanal o mensual

3.2. Análisis de requerimientos

Una vez descrita la situación actual en cuanto a los procedimientos del negocio, se procede con la especificación de los requerimientos del software mediante la identificación de los casos de uso del sistema (Nieves et al., 2014).

En primera instancia se ha identificado cuatro actores que interactúan con el software, como ser: administrador de sistema, encargado de caja, encargado de insumos y personal de servicio, los cuales cumplen un rol y tienen determinadas funciones que realizar, como se describe en la Tabla 2. Asimismo, estos actores se definen mediante una jerarquía de usuarios para fines de diseño del software, como se muestra en la Figura 15.

Tabla 2. Identificación de actores del sistema.

Actor	Descripción
Administrador de sistema	Es el encargado de las configuraciones generales del software, la configuración del control de las cuentas de usuario y la configuración del control de productos - servicios. El usuario administrador no puede tener acceso a las funciones de otros roles, ésto como política de seguridad.
Encargado de caja	Es el encargado de gestionar los recibos de venta, que implica gestionar los recibos y sus detalles, gestionar los pacientes, estudiantes y docentes cuyos datos están presentes en un recibo emitido por caja.
Encargado de insumos	Es el encargado de gestionar el stock de insumos, las entradas y salidas de los insumos médicos correspondientes a los despachos de tratamientos vendidos en caja. Este actor es dependiente de la unidad del Almacén de Insumos y desempeña sus funciones en ambientes independientes al almacén central.
Personal de servicio	Se refiere al personal que trabaja en algún servicio de Clínica Odontológica, como ser: radiología, quirófano, entre otros. Este usuario debe ser notificado si tiene realizar alguna atención en su servicio de acuerdo al recibo emitido en caja.

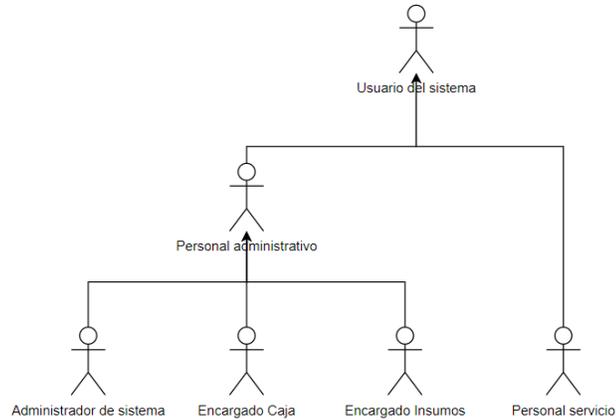


Figura 15. Jerarquía de actores que interactúan con el sistema.

A su vez, los actores identificados hacen uso de determinadas funcionalidades en el sistema las cuales se agrupan en: la gestión de cuentas de usuario, configuración del sistema, el control de insumos, la gestión de recibos y la gestión de reportes. En la Figura 16 se muestra el modelo general de casos de uso agrupado por paquete.

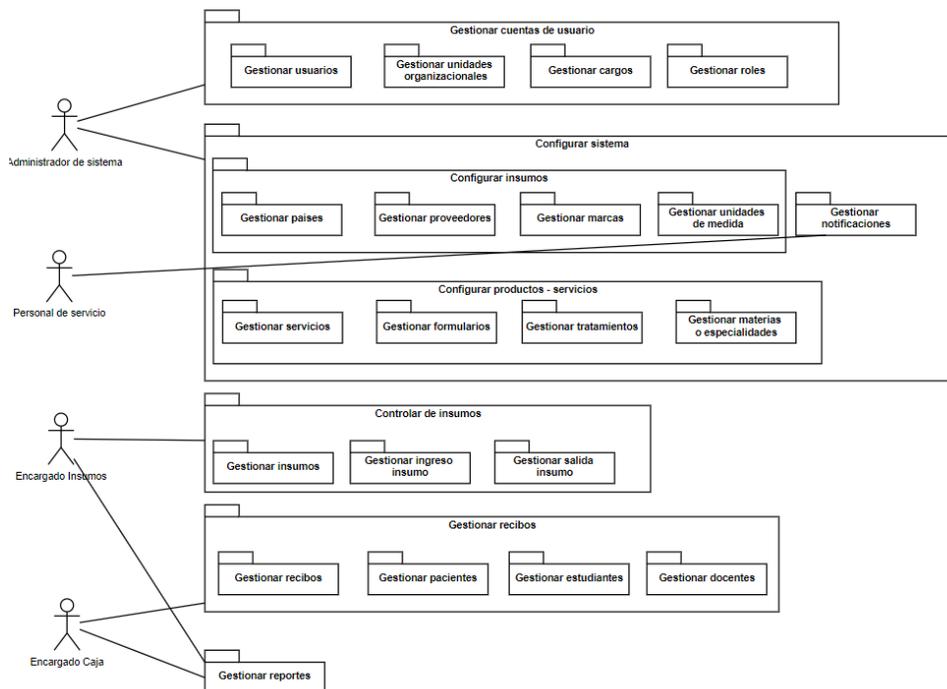


Figura 16. Modelo general de casos de uso agrupado por paquetes

En el paquete Gestionar cuentas de usuario se encuentra cuatro paquetes, que son: a) gestionar usuarios, b) gestionar unidades organizacionales, c) gestionar cargos y d) gestionar roles.

a) Gestionar usuarios, es el módulo que permite realizar operaciones sobre los datos de usuarios del sistema, comprende cinco sub-escenarios, que son: listado, registro, actualización, baja y verificación de credenciales de la cuenta de usuario para el acceso al sistema (Figura 17).

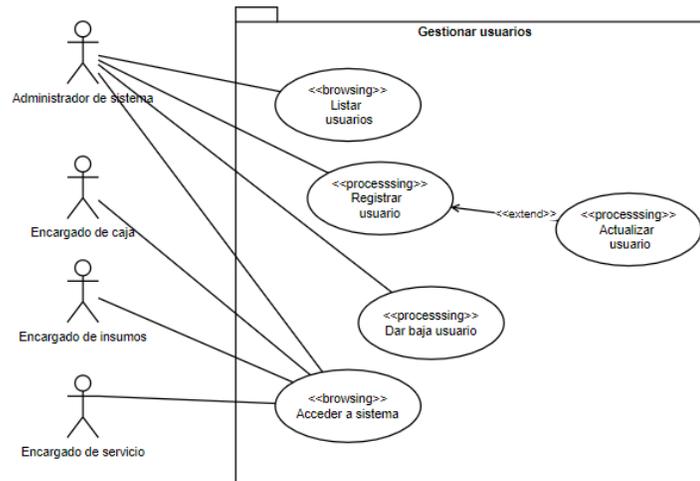


Figura 17. Casos de uso para gestionar usuarios

b) Gestionar unidades organizacionales, es el modulo que permite realizar operaciones sobre los datos de unidades organizacionales, comprende cuatro sub-escenarios, que son: listado, registro, actualización, y baja de unidades organizacionales (Figura 18).

c) Gestionar cargos, es el modulo que permite realizar operaciones sobre los datos de cargos, comprende cuatro sub-escenarios, que son: listado, registro, actualización y baja de cargos (Figura 19).

d) Gestionar roles, es el módulo que permite realizar operaciones sobre los datos de roles del sistema, comprende dos sub-escenarios, que son: listado y actualización de roles (Figura 20).

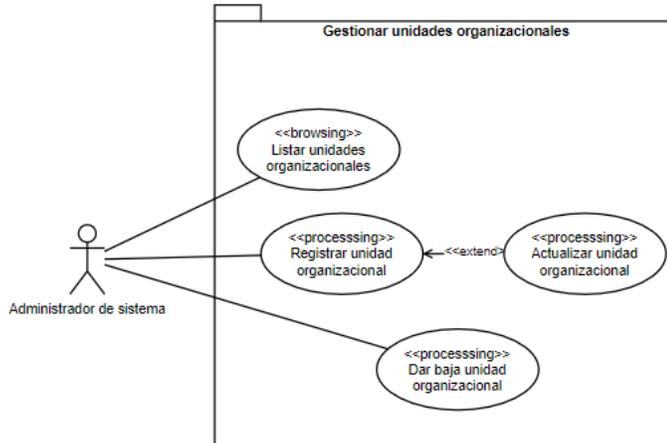


Figura 18. Casos de uso para gestionar unidades organizacionales

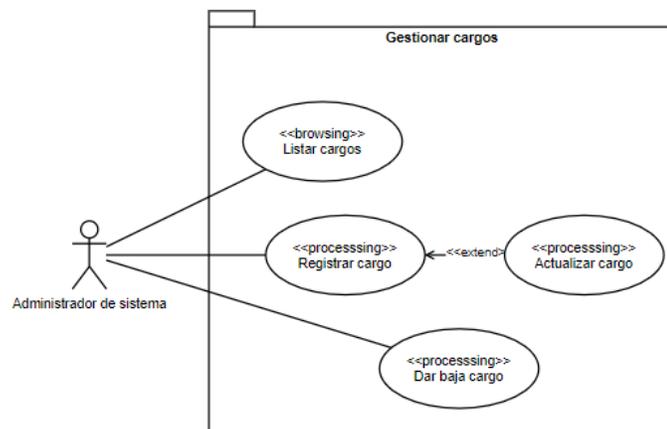


Figura 19. Casos de uso para gestionar cargos.

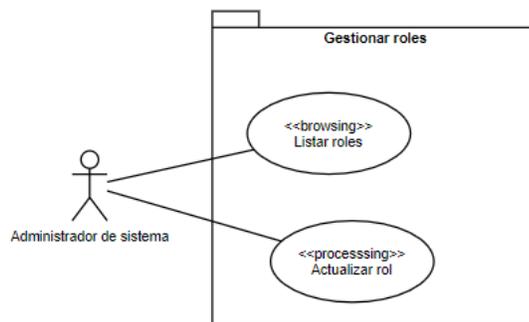


Figura 20. Casos de uso para gestionar roles de usuario.

En el paquete de Configuración de Sistema se encuentra dos sub-paquetes, que son: el paquete de Configuración de Insumos y el paquete de Configuración de productos-servicios. En el paquete de Configuración de Insumos se tiene un grupo

de escenarios, como ser: a) gestionar países, b) gestionar proveedores, c) gestionar marcas, d) gestionar unidades de medida.

a) Gestionar países, es el módulo que permite realizar operaciones sobre los datos de países, comprende cuatro sub-escenarios, que son: listado, registro, actualización y baja de países (Figura 21).

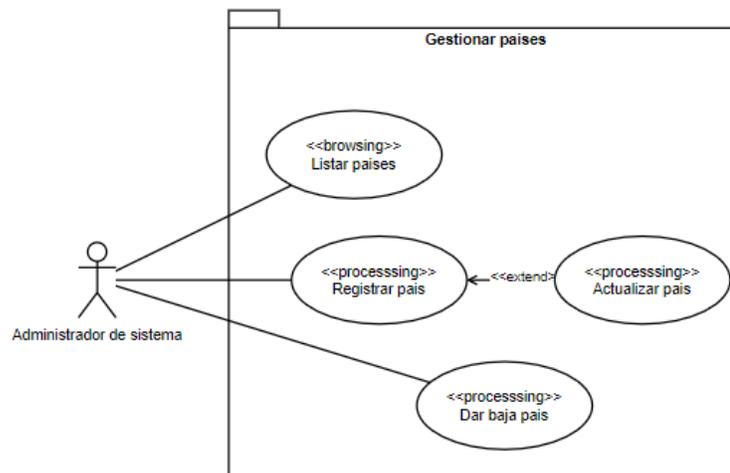


Figura 21. Casos de uso para gestionar países.

b) Gestionar proveedores, es el módulo que permite realizar operaciones sobre los datos de proveedores, comprende cuatro sub-escenarios, que son: listado, registro, actualización y baja de proveedores de insumos médicos (Figura 22).

c) Gestionar marcas, es el módulo que permite realizar operaciones sobre los datos de marcas de insumos, comprende cuatro sub-escenarios, que son: listado, registro, actualización y baja de marcas de insumos médicos (Figura 23).

d) Gestionar unidades de medida, es el módulo que permite realizar operaciones sobre los datos de unidades de medida de insumos, comprende cuatro sub-escenarios, que son: listado, registro, actualización y baja de unidades de medida de insumos médicos (Figura 24).

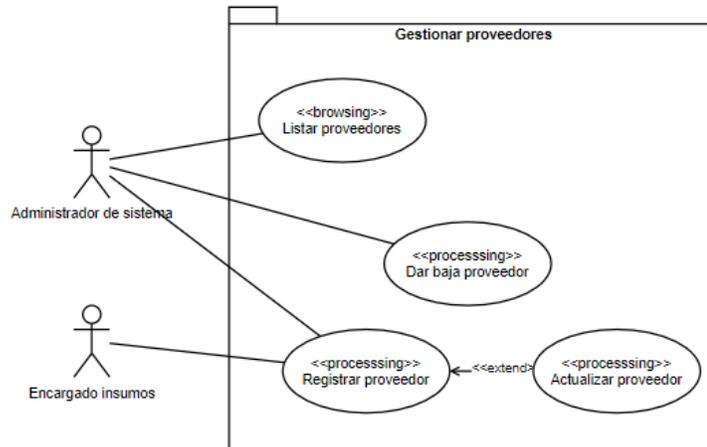


Figura 22. Casos de uso para gestionar proveedores.

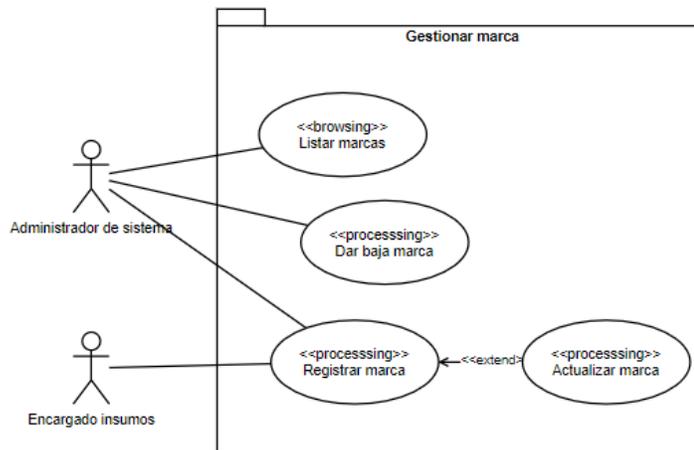


Figura 23. Casos de uso para gestionar cargos.

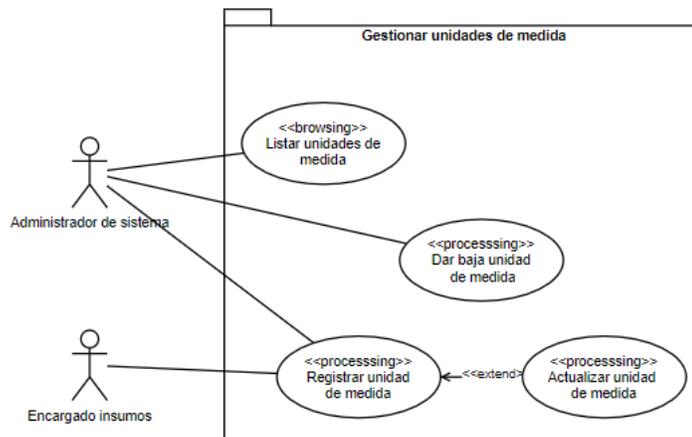


Figura 24. Casos de uso para gestionar cargos.

En el paquete de Configuración de productos-servicios se tiene un grupo de escenarios, como ser: a) gestionar servicios, b) gestionar formularios de historial clínico, c) gestionar tratamientos, d) gestionar materias o especialidades.

a) Gestionar servicios, es el módulo que permite realizar operaciones sobre los datos de servicios, comprende cuatro sub-escenarios, que son: listado, registro, actualización y baja de servicios de clínica odontológica (Figura 25).

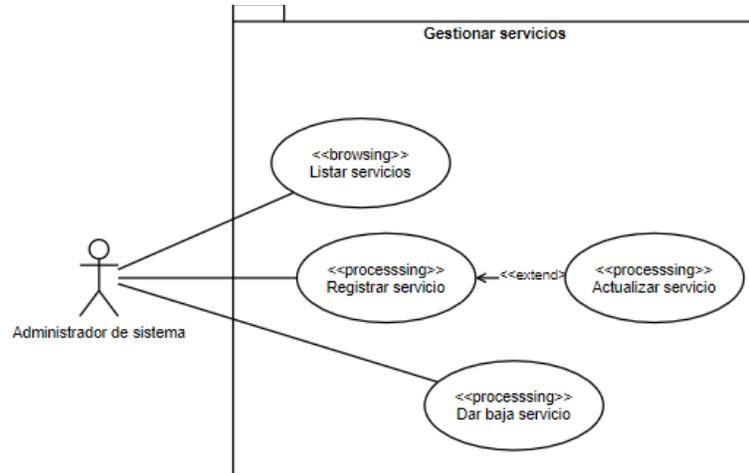


Figura 25. Casos de uso para gestionar servicios.

b) Gestionar formularios de historial, es el módulo que permite realizar operaciones sobre los datos de formularios de historial clínico, comprende cuatro sub-escenarios, que son: listado, registro, actualización y baja de formularios (Figura 26).

c) Gestionar tratamientos, es el módulo que permite realizar operaciones sobre los datos de tratamientos, comprende cuatro sub-escenarios, que son: listado, registro, actualización y baja de tratamientos odontológicos (Figura 27).

d) Gestionar materias-especialidades, es el módulo que permite realizar operaciones sobre los datos de materias o especialidades de clínica odontológica, comprende cuatro sub-escenarios, que son: listado, registro, actualización y baja de materias o especialidades (Figura 28).

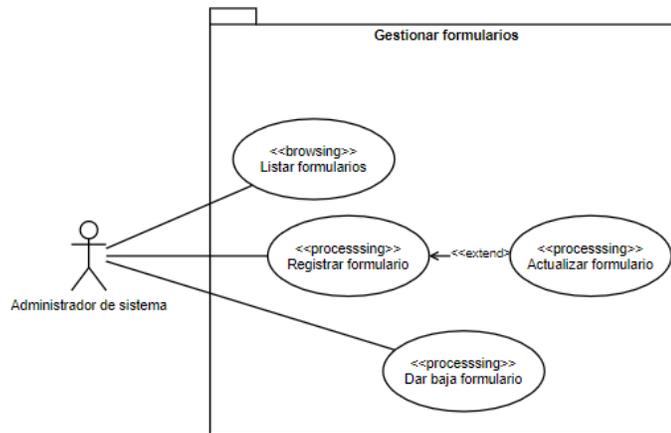


Figura 26. Casos de uso para gestionar cargos.

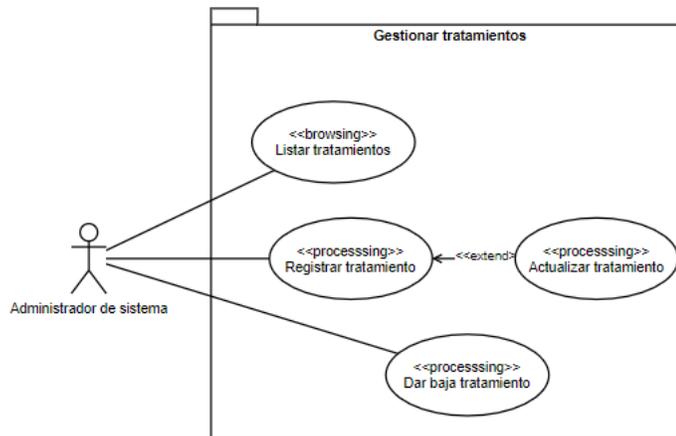


Figura 27. Casos de uso para gestionar cargos.

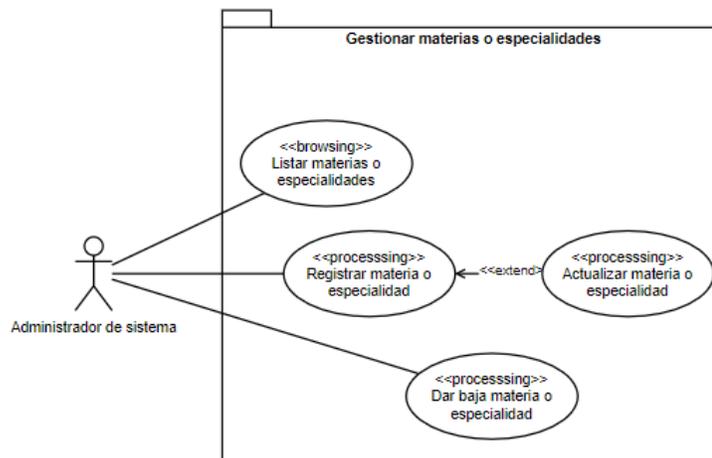


Figura 28. Casos de uso para gestionar cargos.

Por otra parte se tiene el paquete de gestión de notificaciones, el cual permite realizar operaciones sobre los datos de notificaciones, y comprende tres escenarios,

que son: listado de notificaciones general, listado de notificaciones por servicio y actualización de la configuración de notificaciones (Figura 29).

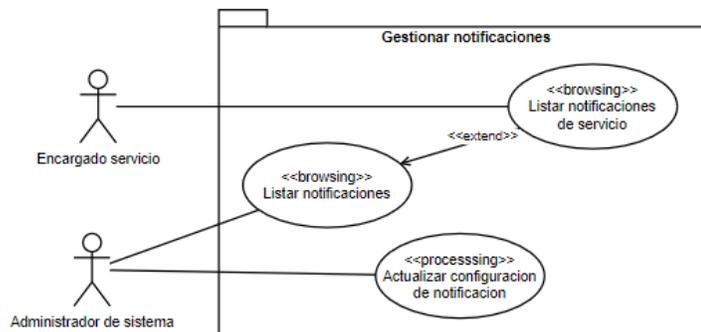


Figura 29. Casos de uso para gestionar notificaciones.

En el paquete de Control de Insumo se encuentran tres paquetes, que son: a) gestionar insumos, b) gestionar ingreso de insumos y c) gestionar salida de insumos.

a) Gestionar insumos, es el módulo que permite realizar operaciones sobre los datos del registro de insumos, comprende cuatro sub-escenarios, que son: listado, registro, actualización y baja de insumos (Figura 30).

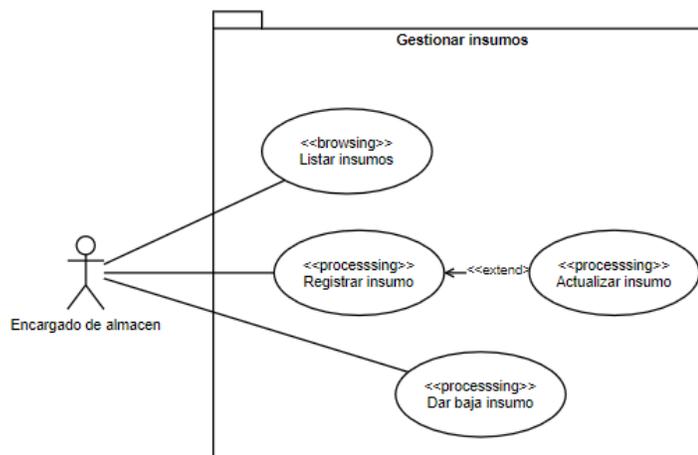


Figura 30. Casos de uso para gestionar insumos.

b) Gestionar ingreso de insumos, es el módulo que permite realizar operaciones sobre los ingresos de insumos, comprende cinco sub-escenarios, que son: listado, registro, actualización, baja del ingreso de insumos, y actualización del stock de insumos (Figura 31).

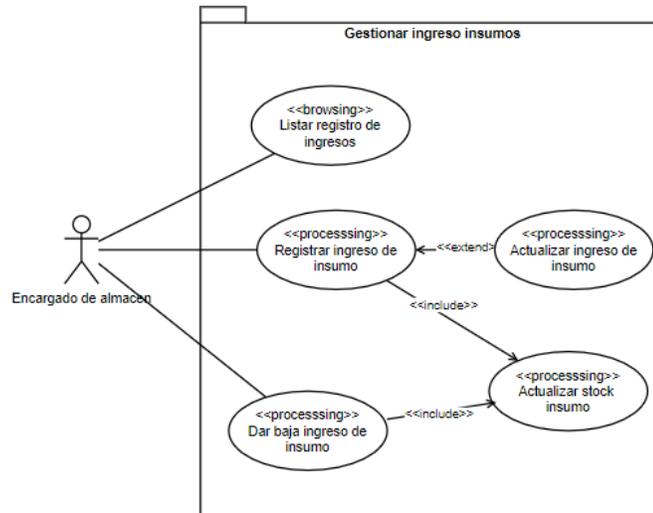


Figura 31. Casos de uso para gestionar ingreso de insumos.

c) Gestionar salidas de insumos, es el módulo que permite realizar operaciones sobre los datos de salidas de insumos, comprende cuatro sub-escenarios, que son: listado, registro, actualización, baja de salidas de insumos, y actualización del stock de insumos (Figura 32).

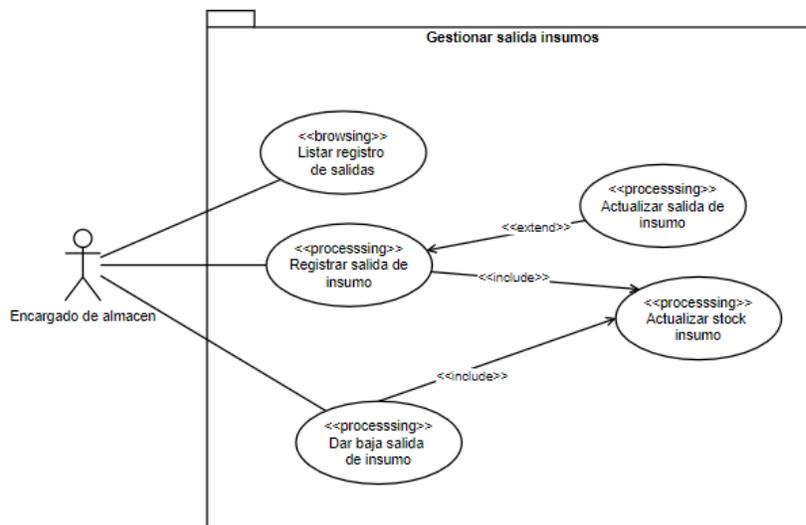


Figura 32. Casos de uso para gestionar salida de insumo.

En el paquete de Gestión de recibos se encuentran cuatro paquetes, que son: a) gestionar recibos, b) gestionar pacientes, c) gestionar estudiantes y d) gestionar docentes.

a) Gestionar recibos, es el módulo que permite realizar operaciones sobre los datos de recibos, comprende ocho sub-escenarios, que son: listado, registro, baja, impresión de recibos. Asimismo, permite registrar paciente, registrar el detalle del pedido, registrar la autorización del docente y realizar una búsqueda de pacientes (Figura 33).

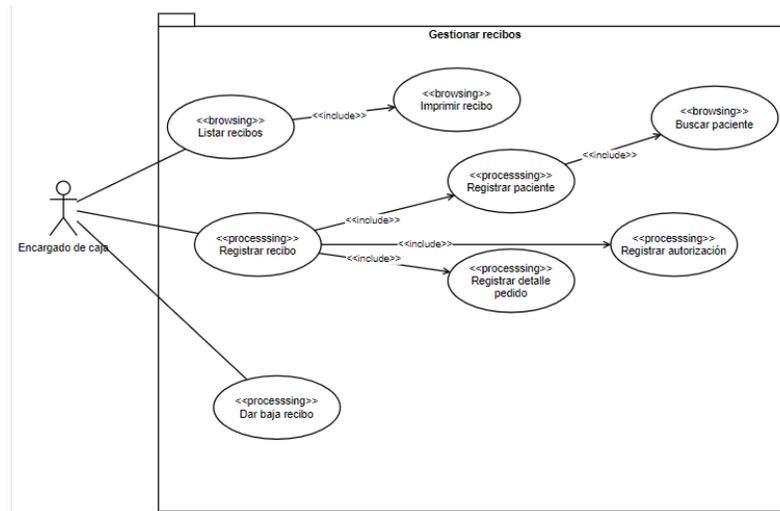


Figura 33. Casos de uso para gestionar recibos

b) Gestionar pacientes, es el módulo que permite realizar operaciones sobre los datos de pacientes, comprende tres sub-escenarios, que son: listado, actualización y baja de ingreso de pacientes (Figura 34).

c) Gestionar docentes, es el módulo que permite realizar operaciones sobre los datos de docentes, comprende cuatro sub-escenarios, que son: listado, registro, actualización y baja de docentes (Figura 35).

d) Gestionar estudiantes, es el módulo que permite realizar operaciones sobre los datos de estudiantes, comprende cuatro sub-escenarios, que son: listado, registro, actualización y baja de estudiantes (Figura 36).

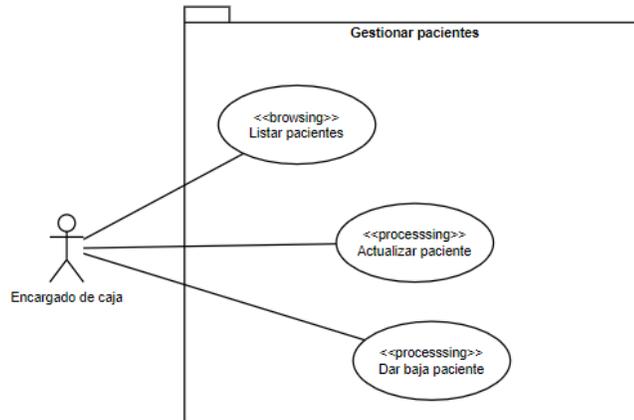


Figura 34. Casos de uso para gestionar pacientes

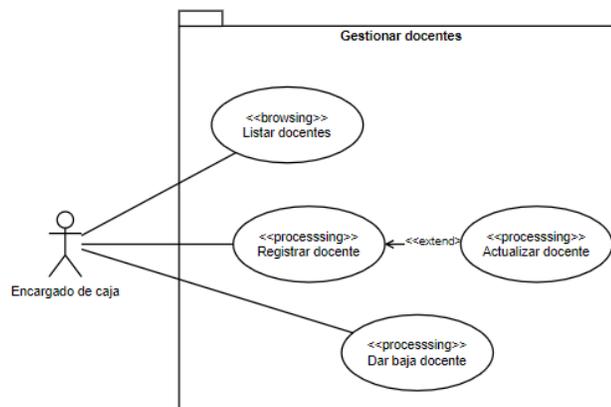


Figura 35. Casos de uso para gestionar docentes

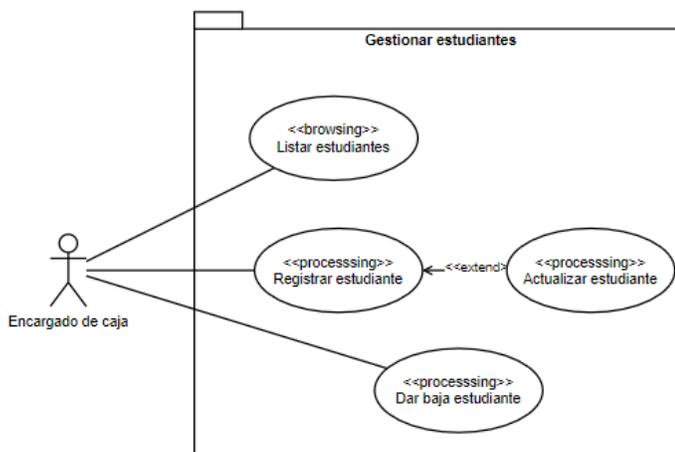


Figura 36. Casos de uso para gestionar estudiantes

Finalmente, en el paquete de Gestión de reportes se encuentran diecisiete paquetes destinados a mostrar reportes como se muestra en la Figura 37. A su vez estos se agrupan en: reportes de ingreso económico diario, reportes de ingresos económico

entre rango de fechas (semanal, mensual, entre otros), reporte de tratamientos más vendidos, reporte de tratamientos realizados por estudiante, reporte de tratamientos con más recaudación, reporte de ingreso y salidas de insumos.

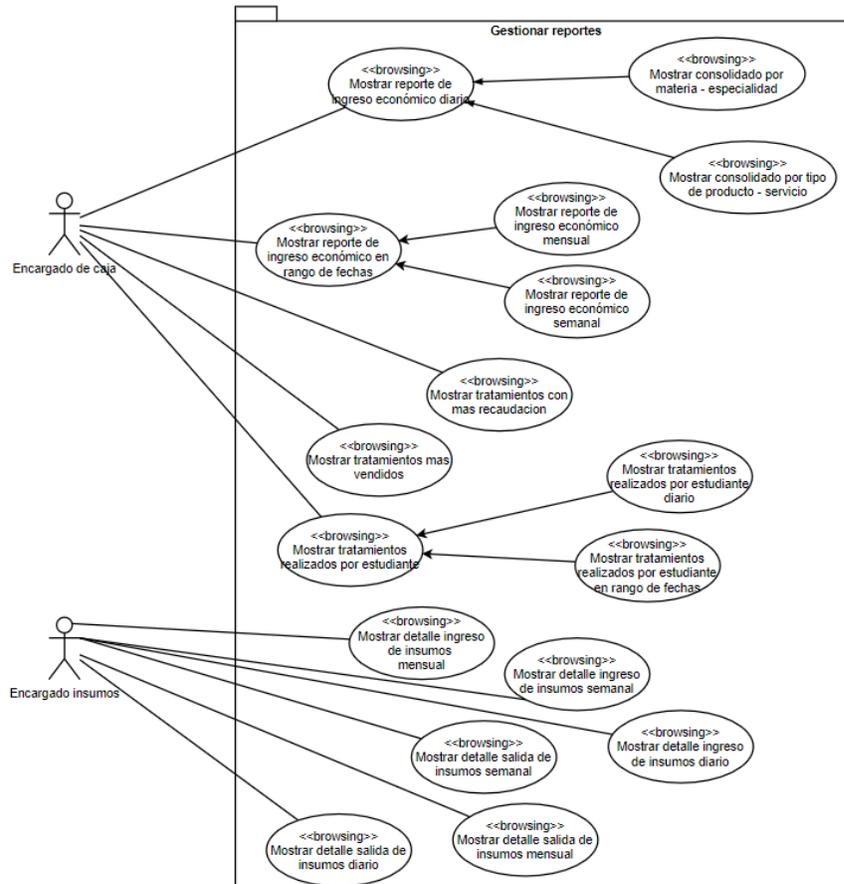


Figura 37. Casos de uso para gestionar reportes

3.2.1. Requerimientos funcionales

Analizados los escenarios correspondientes, se identificó los requerimientos los cuales se dividen en seis grupos que corresponden a la estructura de paquetes UML en el modelo de casos de uso. Estos son: gestión de cuentas de usuario, configuración de insumos, configuración de productos – servicios, control de insumos, gestión de recibos y gestión de reportes. La lista de requerimientos se presenta en la Tabla 3. Los requerimientos funcionales corresponden a la dimensión de Funcionalidad del estándar ISO/IEC 9126.

Tabla 3. Lista de requerimientos funcionales.

#	Descripción del requerimiento
Gestión de cuentas de usuario	
1	Gestionar los usuarios del sistema, permitiendo el listado, registro, edición y baja de usuario. Estas operaciones las debe realizar el administrador del sistema.
2	Gestionar las unidades organizacionales, permitiendo el listado, registro, edición y baja de unidad. Este módulo tiene como finalidad de organizar a los usuarios del sistema. Estas operaciones las debe realizar el administrador del sistema.
3	Gestionar los cargos de la entidad, permitiendo el listado, registro, edición y baja de cargos. Este módulo tiene como finalidad de organizar a los usuarios del sistema. Estas operaciones las debe realizar el administrador del sistema.
4	Gestionar roles de usuario, permitiendo únicamente el listado y edición del nombre del rol. Los roles de usuario se crean una sola vez, y no se permite su baja o eliminación. Estas operaciones las debe realizar solamente el administrador de sistema.
Configuración de insumos	
5	Gestionar países, permitiendo el listado, registro, edición y baja de países. Este módulo tiene como finalidad la organización de proveedores de insumos médicos. Estas operaciones las debe realizar solamente el administrador de sistema.
6	Gestionar proveedores, permitiendo el listado, registro, edición y baja de proveedores de insumos. Este módulo tiene como finalidad la organización de los insumos médicos que se van a registrar. Estas operaciones las puede realizar el administrador del sistema o el encargado de insumos.
7	Gestionar marcas, permitiendo el listado, registro, edición y baja de marcas de insumos. Este módulo tiene como finalidad la organización de los insumos médicos que se van a registrar. Estas operaciones las puede realizar el administrador del sistema o el encargado de insumos.
8	Gestionar unidades de medida, permitiendo el listado, registro, edición y baja de unidades de medida. Este módulo tiene como finalidad la organización de los insumos médicos que se van a registrar. Estas operaciones las puede realizar el administrador del sistema o el encargado de insumos.
9	Gestionar notificaciones, permitiendo al Encargado de servicio visualizar los recibos por concepto de tratamiento que corresponden a un servicio de la clínica. El administrador del sistema puede actualizar la configuración de notificaciones, cambiando el mensaje que va a realizar al sistema al encargado de servicio.
Configuración de productos - servicios	
10	Gestionar servicios, permitiendo el listado, registro, edición y baja de servicios. Este módulo tiene como finalidad la organización de tratamientos ofertados en caja. Estas operaciones las puede realizar el administrador de sistema.
11	Gestionar formularios de historial, permitiendo el listado, registro, edición y baja de formularios de historial clínico. Este módulo tiene como finalidad la organización de historiales clínicos disponibles en el sistema. Estas operaciones las puede realizar el administrador del sistema.
12	Gestionar tratamientos, permitiendo el listado, registro, edición y baja de tratamientos. Este módulo tiene como finalidad la organización de tratamientos clínicos disponibles en el sistema. Estas operaciones las puede realizar el administrador del sistema.
13	Gestionar las materias o especialidades, permitiendo el listado, registro, edición y baja de especialidades. Este módulo tiene como finalidad la organización de tratamientos y

	formularios de historial clínico. Estas operaciones las puede realizar el administrador de sistema.
	Control de insumos
14	Gestionar los insumos, permitiendo el listado, registro, edición y baja de insumos. A su vez, gestionar el ingreso insumos actualizando el stock en inventario, como también la salida de insumos correspondiente a los recibos emitidos en caja y actualizando respectivamente el stock en inventario. Estas operaciones las debe realizar el encargado de insumos.
15	Gestionar los recibos, permitiendo el listado, registro y baja de recibos. El registro de recibos implicará el registro de paciente, el registro de la autorización del docente para realizar la emisión del recibo correspondiente a una venta de tratamiento o formulario de historial clínico, y el registro del detalle del pedido correspondiente al recibo. Estas operaciones las debe realizar el encargado de caja.
16	Gestionar los pacientes, permitiendo el listado, actualización y baja de pacientes. Estas operaciones las puede realizar el encargado de caja.
17	Gestionar los docentes, permitiendo el listado, registro, actualización y baja de docentes. Estas operaciones las puede realizar el encargado de caja.
18	Gestionar los estudiantes, permitiendo el listado, registro, actualización y baja de estudiantes. Estas operaciones las puede realizar el encargado de caja.
	Gestión de reportes
19	Mostrar el reporte de ingreso económico, consolidado por materia o especialidad, o por producto o servicio, de forma diaria o entre un rango de fechas (semanal, mensual, trimestral, entre otros).
20	Mostrar el reporte de los tratamientos con mayor recaudación.
21	Mostrar el reporte de los tratamientos más vendidos.
22	Mostrar el detalle del ingreso de insumos, de forma diaria o entre un rango de fechas (semanal, mensual, entre otros).
23	Mostrar el detalle de las salidas de insumos de forma diaria o entre un rango de fechas (semanal, mensual, entre otros).

3.2.2. Requerimientos no funcionales

Los requerimientos no funcionales corresponden al resto de dimensiones del modelo calidad de producto software del estándar ISO/IEC 9126. Estas dimensiones son: eficiencia del desempeño, compatibilidad, usabilidad, fiabilidad, seguridad, mantenibilidad, portabilidad. Los requerimientos no funcionales identificados se describen en la Tabla 4.

Tabla 4. Lista de requerimientos no funcionales.

#	Requerimientos no funcionales
	Eficiencia del desempeño
1	El tiempo de respuesta promedio para cargar las interfaces Web y realizar operaciones no debe exceder los 5 segundos.

2	Se deben optimizar los recursos del proyecto (imágenes, archivos de código) para minimizar los tiempos de respuesta del servidor Web.
3	El sistema debe prevenir la concurrencia de al menos 10 usuarios a la vez.
	Compatibilidad
4	En caso de que el sistema necesite comunicarse con otros servicios externos, debe realizarlo empleando protocolos y funciones estándar
5	El código del sistema no debe interferir con las operaciones de otros servicios en caso de compartir recursos en un mismo servidor.
	Usabilidad
6	Se debe utilizar tipografías y colores que hagan legible la interpretación de las interfaces gráficas de usuario, además de estar enmarcados a la imagen corporativa de la institución.
7	Los términos utilizados en las interfaces gráficas de usuario deben ser compatibles con el vocabulario de los usuarios finales del software.
8	El esquema de navegación del software debe ser simple de entender.
9	En caso de registrarse errores al momento de operación, el software debe mostrar los mensajes respectivos que orienten al usuario a corregir sus acciones.
10	Los campos de formularios deben estar descritos lo suficiente como para que los usuarios finales entiendan los datos que deben introducir.
	Fiabilidad
11	El software debe ser accesible desde una red privada institucional o Internet para que los usuarios finales puedan realizar sus actividades.
12	En caso de caída de sistema, el software debe recuperarse automáticamente al restaurar los servicios del centro de datos.
	Seguridad
13	Debe controlar los accesos mediante un usuario y contraseña.
14	Debe incorporar un código de verificación CATPCHA para evitar la intrusión de robots o software malicioso en el sistema durante el acceso de usuarios.
15	Las contraseñas de usuario no deben estar almacenados en texto plano, por tanto debe utilizarse algún método de codificación para que las claves de los usuarios no sean legibles.
16	Para controlar la integridad de los datos, se hará uso de un esquema de control basado en roles para evitar inconsistencias.
17	No se hará un borrado físico de los datos con el fin de preservar un historial de datos, solo se empleará el borrado lógico, esto con fines de auditoría.
18	Se registrará un log de operaciones sobre ciertas tablas cruciales para el negocio, y no así en toda la base de datos, con fines de auditoría.
19	Se debe registrar un identificador que permita detectar quien realizó alguna modificación a los datos.
20	Se debe emplear el estándar de verificación de aplicaciones Web de OWASP para verificar el nivel de seguridad del software.
	Mantenibilidad
21	Se debe documentar el código fuente para futuras modificaciones al sistema.
22	Debe respetarse la estructura del proyecto para no generar retrasos en caso de realizar futuras modificaciones.
	Portabilidad
23	El software debe permitir su instalación en un centro de datos institucional.
24	Debe facilitarse una guía de instalación

26	No debe ser dependiente de una plataforma en particular, como por ejemplo Windows, Mac o Gnu/Linux.
----	---

Adicionalmente, para que el software opere correctamente se ha definido un conjunto de requerimientos mínimos para: el servidor que albergará el código fuente, y también las terminales que accederán al software a través de Internet o una red privada. En la Tabla 5 se describen los requerimientos mínimos para el servidor, y en la Tabla 6 se describen los requerimientos mínimos para las terminales de usuario.

Tabla 5 Requerimientos mínimos para el servidor.

REQUERIMIENTOS MINIMOS PARA EL SERVIDOR	
MEMORIA RAM	4GB
ALMACENAMIENTO	50GB
CONEXIÓN A INTERNET	10Mbps
IP – DOMINIO	IP dedicado o dominio o subdominio
TIPO DE SERVIDOR	1 Servidor Virtual, con sistema Gnu/Linux de preferencia.

Tabla 6. Requerimientos mínimos para las terminales de usuario.

REQUERIMIENTOS MINIMOS PARA LAS TERMINALES DE USUARIO	
PROCESADOR	Intel i3 o más reciente.
MEMORIA RAM	4GB
ALMACENAMIENTO	50GB
CONEXIÓN A INTERNET	2Mbps
SISTEMA OPERATIVO	Windows 7, 8 o 10.
NAVEGADOR WEB	Google Chrome v. 87+, Firefox v. 55+

3.3. Diseño

3.3.1. Modelo de contenido

El modelo de contenido es elaborado con base a la notación del método UWE permite abstraer y representar la información referente a los procedimientos analizados para dar respuesta a los requerimientos identificados. El modelo de contenido se muestra en la Figura 38, e incluye 24 clases que permiten definir las

tablas de datos del sistema. En los atributos de las clases del modelo de contenido no se incluyen los atributos que van a permitir la trazabilidad de los registros en la base de datos, solo se muestran los atributos relevantes para la lógica del negocio.

3.3.2. Modelo de navegación

El modelo de navegación es elaborado con base a la notación del método UWE, y permite abstraer la forma en la cual cada actor identificado interactúa con el sistema. A su vez, los diagramas de navegación permiten definir el acceso a funcionalidades que cada rol tiene, estos diagramas se dividen por roles en: diagrama de navegación para administrador de sistema (Figura 39), diagrama de navegación para el encargado de caja (Figura 40), diagrama de navegación para el encargado de insumos (Figura 41), y diagrama de navegación para el personal de servicio (Figura 42).

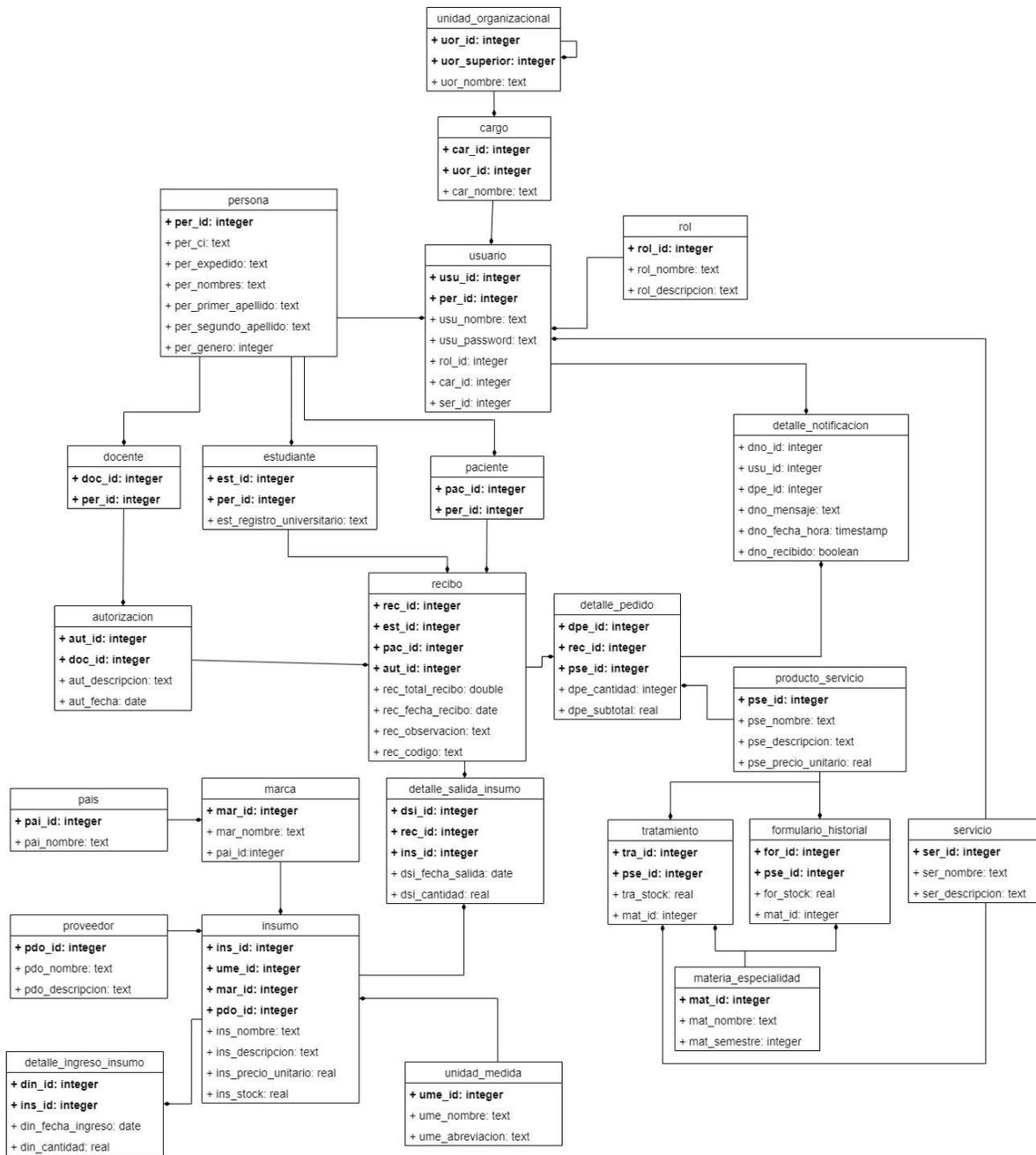


Figura 38. Modelo de contenido, representa los datos que estarán presentes en la base de datos

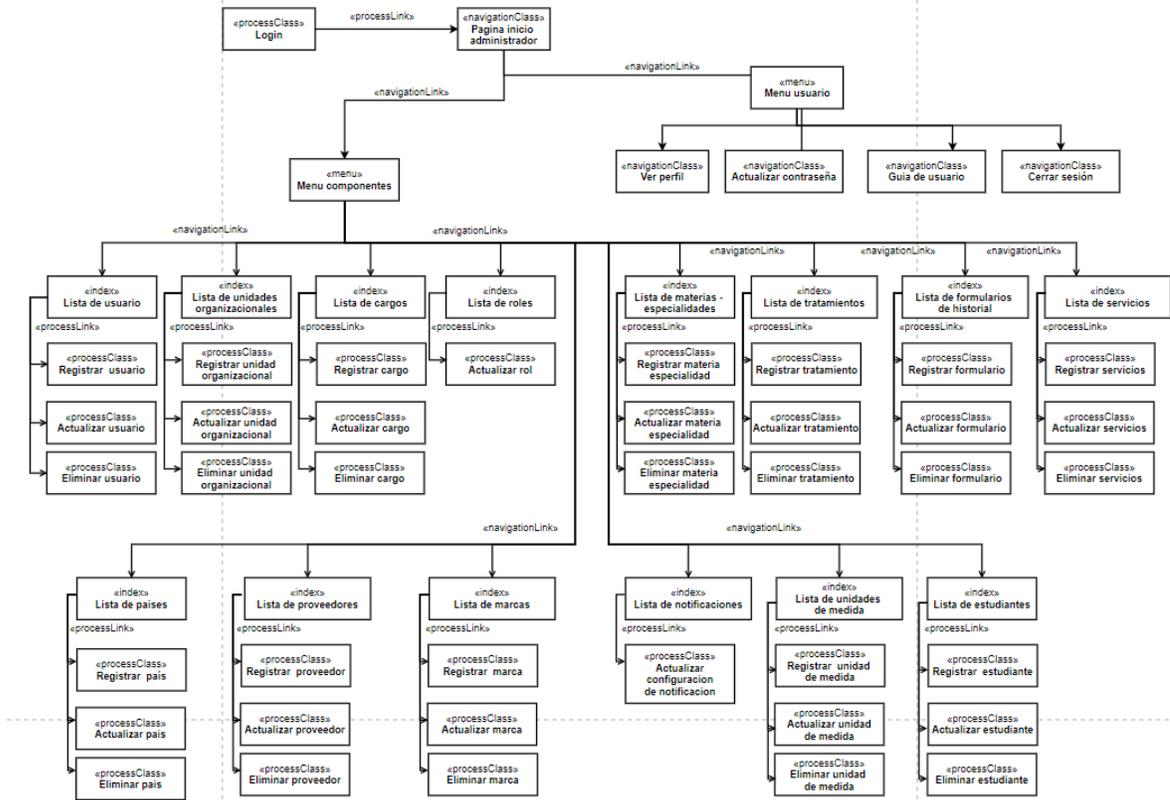


Figura 39. Diagrama de navegación para el usuario con rol de Administrador

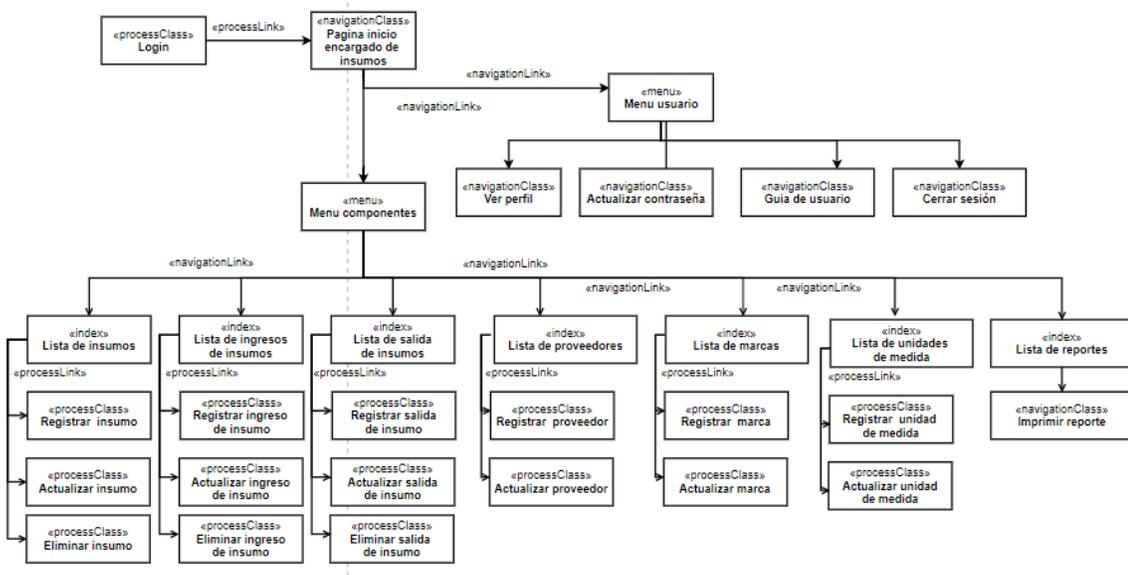


Figura 40. Diagrama de navegación para el usuario con rol de Encargado de insumos.

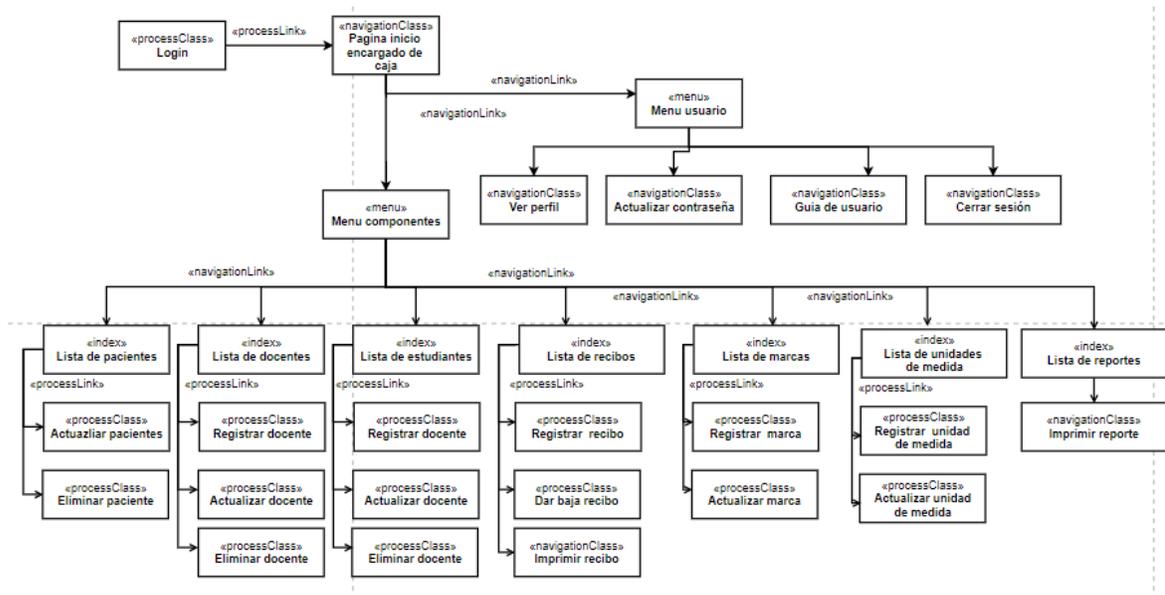


Figura 41 Diagrama de navegación para el usuario con rol de Encargado de caja

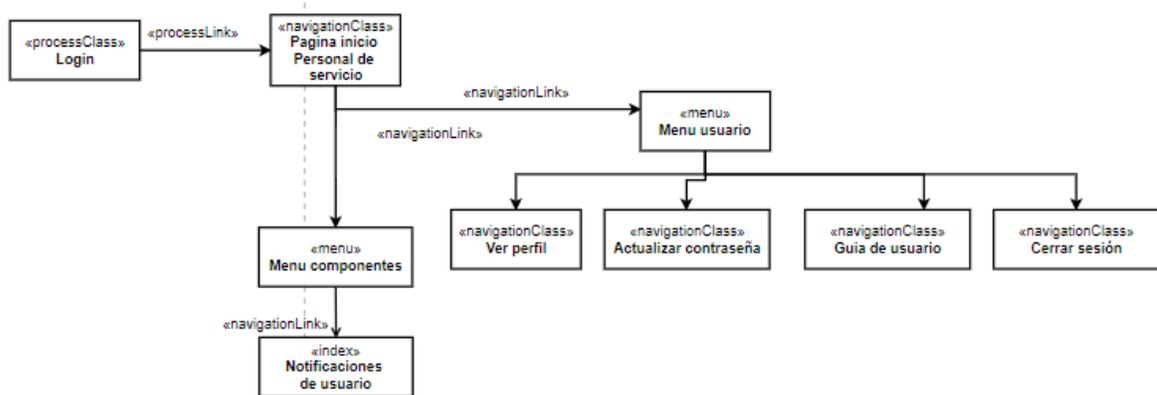


Figura 42 Diagrama de navegación para el usuario con rol Personal de servicio

3.3.3. Modelo de proceso

El modelo de proceso es elaborado con base a la notación del método UWE, y permite afinar el diseño de los procesos del sistema con diagramas de actividades UML. Adicionalmente, se le agregan los carriles al diagrama de actividades para tener un diseño más claro. Los escenarios que se han modelado corresponden a escenarios como ser: a) acceder al sistema, donde interactúan todos los usuarios del sistema, el sistema y el servicio Google Recaptcha (Figura 43), b) listar ítem,

donde interactúa el personal administrativo de acuerdo a la jerarquía de actores y que hace referencia a todos los casos de uso donde se hace un lista de algún ítem o conjunto de ítems extraídos de la base de datos (Figura 44), c) registrar ítem, donde interactúa el personal administrativo, y hace referencia a todos los casos de uso donde se registra algún ítem en la base de datos (Figura45), d) actualizar ítem, donde interactúa el personal administrativo para actualizar algún ítem en la base de datos (Figura 46), y e) dar baja ítem, donde interactúa el personal administrativo para dar de baja algún ítem en la base de datos (Figura 47).

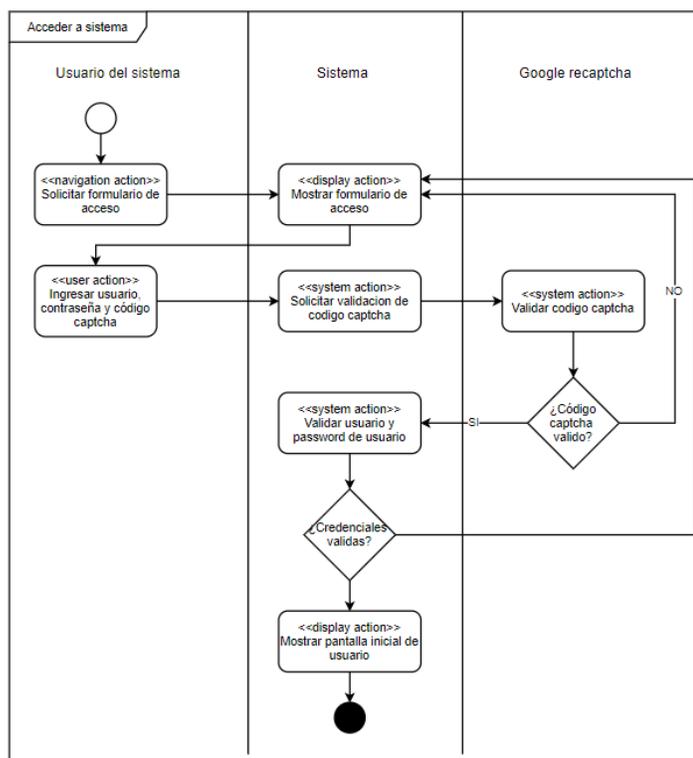


Figura 43 Diagrama de actividad para especificar el procedimiento de acceso al sistema

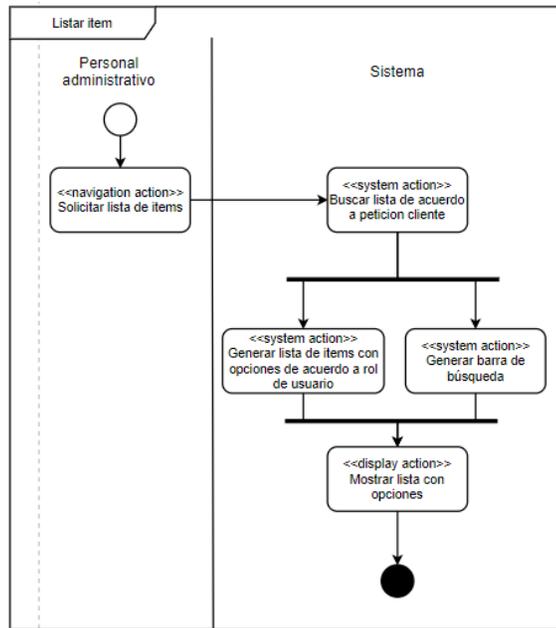


Figura 44 Diagrama de actividad para especificar el procedimiento de listar ítems

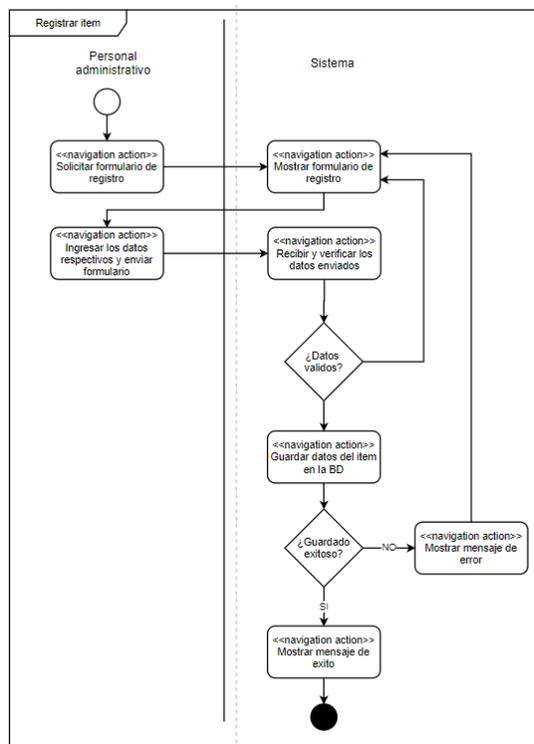


Figura 45 Diagrama de actividad para especificar el procedimiento de registro de un ítem

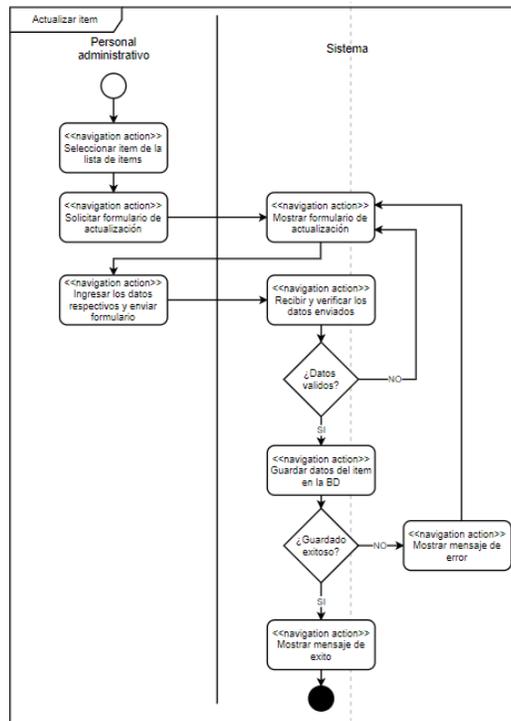


Figura 46 Diagrama de actividad para especificar el procedimiento de actualizar un ítem

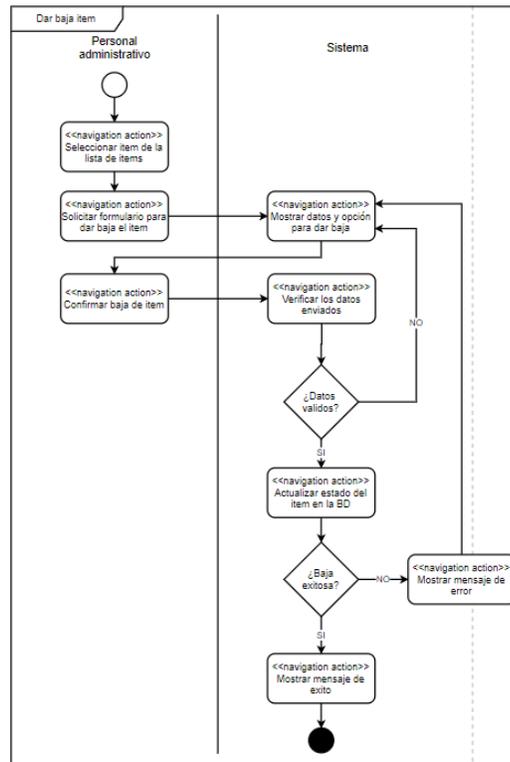


Figura 47 Diagrama de actividad para especificar el procedimiento de dar baja un ítem

3.4. Construcción

Para la etapa de construcción del software se gestionaron las tareas con el método Kanban, y se ha establecido un WIP de 1 tarea, dado que solo una persona ha trabajado en el proyecto. Las tareas se dividieron por casos de uso y se utilizó un tablero Kanban para hacer el seguimiento del avance de las tareas. Se muestra en el Anexo B el tablero Kanban utilizado, donde ha definido cuatro etapas, como ser: tareas en espera, maquetación, programación, pruebas y tareas concluidas.

3.4.1. Arquitectura del sistema

El modelo de arquitectura empleado para desarrollar el presente proyecto de grado es Cliente – Servidor. Este modelo permite organizar un sistema en conjuntos de servicios y servidores asociados más unos clientes que acceden y usan los servicios (Sommerville, 2005). Dado que los usuarios finales acceden al software almacenado en un servidor mediante terminales o computadoras a través de un protocolo de red (como ser HTTP), el modelo cliente – servidor es adecuado para los requerimientos.

Por otra parte, en el proyecto se hace uso del framework PHP Codeigniter para desarrollar el código fuente. Codeigniter implementa el patrón de arquitectura Modelo – Vista – Controlador, el cual al separar la lógica de negocio de la interfaz de usuario permite tener código fuente reutilizable y flexible, tal como se describe en anteriores secciones del presente trabajo. A su vez, éste framework permite conectarse a la base de datos mediante rutinas y protocolos estándar.

Adicionalmente, el software hace uso del servicio de verificación de código Captcha de Google, denominado Recaptcha. Dichas verificaciones para el acceso de usuario se hacen a través de una Interfaz de Programación de Aplicaciones (API), el cual permite acceder al servicio mediante un pequeño programa cliente que es parte del sistema y que hace consultas al servicio externo mediante protocolo HTTP.

En síntesis, la arquitectura de sistema utilizado en el presente proyecto de grado (Figura 48) adopta el modelo cliente servidor, el cliente (navegador Web) accede a

las funcionalidades del software desarrollados con base al patrón de arquitectura MVC almacenado en el servidor, y éste último a su vez puede hacer uso de servicios externos comunicándose a través de protocolos estándar, como HTTP.

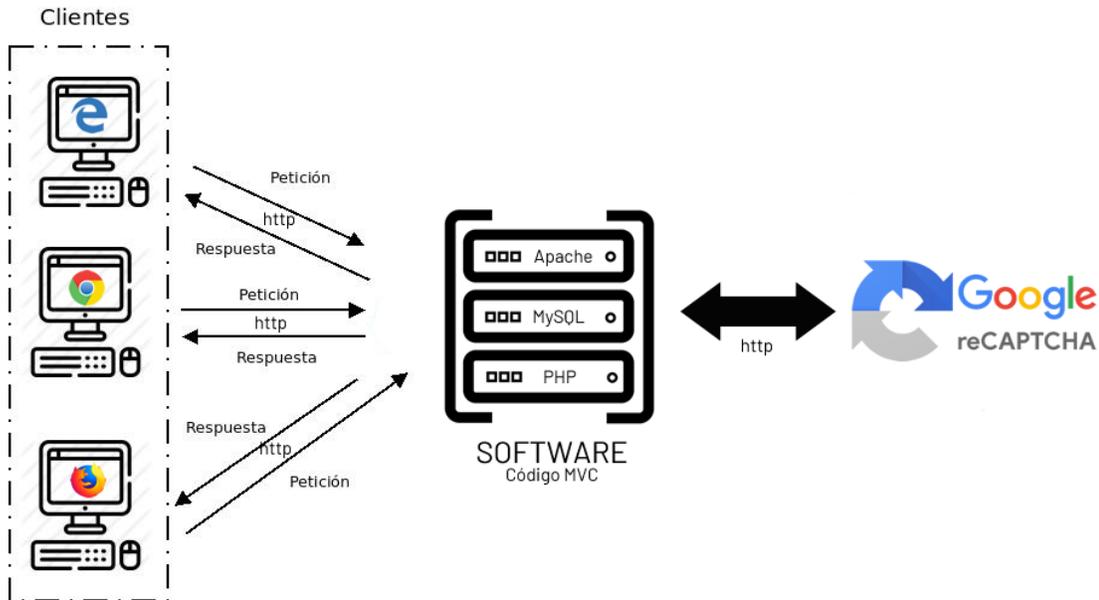


Figura 48. Arquitectura del sistema
Fuente. Elaborado con base a (Sommerville, 2005)

3.4.2. Base de datos

De acuerdo al modelo de contenido presentado en la Figura 38, se procede con la programación del script en formato SQL para la creación del esquema de base de datos. Como se pudo evidenciar en el modelo de contenido diseñado con base al método UWE, las tablas de la base de datos no tienen definidos los atributos necesarios para la trazabilidad eficiente de registros, por lo que se deben incluir en cada tabla cinco atributos que se definen en la Tabla 7. El diccionario de datos que describe cada uno de las tablas y atributos del esquema de la base de datos se presenta en el Anexo C.

Tabla 7 Atributos necesarios para la trazabilidad de registro de la base de datos

Atributo	Tipo de dato	Valor por defecto	Descripción
created_at	Timestamp	CURRENT TIMESTAMP	Es un atributo de tipo timestamp, hace referencia al momento exacto en el cual fue creado el registro y su valor se genera automáticamente en la base de datos.
updated_at	Timestamp	CURRENT TIMESTAMP	Es un atributo de tipo timestamp, hace referencia al momento exacto en el cual fue actualizado el registro. Su valor inicial coincide con la creación del registro, y en caso de actualización el valor los establece el sistema.
deleted_at	Timestamp	NULL	Es un atributo de tipo timestamp, hace referencia al momento exacto en el cual fue eliminado el registro. Su valor inicial es nulo. El valor en caso de eliminación de registro se establece por sistema.
deleted	Boolean	False	Es un atributo de tipo booleano, permite definir si un atributo fue eliminado o no, es decir que no se realiza un eliminado físico, sino se hace un cambio de estado al registro que el sistema interpreta como eliminado. Su valor inicial es false (no eliminado), y en caso de que el usuario decida borrar el registro se cambia el valor a true (eliminado).
usr_at	Integer	0	Es un atributo de tipo entero, hace referencia al ID de usuario que está efectuando la última operación sobre el registro, ya sea crear, actualizar o eliminar.

3.4.3. Interfaces gráficas de usuario

Con base al diseño del sistema realizado con el método UWE, se procede a la maquetación de las interfaces gráficas de usuario y se procede a escribir el código fuente en el lenguaje PHP para que las funcionalidades cumplan con los requerimientos establecidos.

3.4.3.1. Acceso y pantalla de bienvenida

En primera instancia, se realiza la maquetación del formulario de acceso al sistema (Figura 49) el cual presenta tres campos que son obligatorios de llenar, como ser: usuario, contraseña y código Captcha. Una vez validadas las credenciales del usuario, se redirige al usuario a su pantalla de bienvenida correspondiente al rol que cumple en el sistema, por ejemplo, el administrador de sistema (Figura 50). La pantalla de bienvenida del software tiene como elementos base un menú en la parte superior desde donde se puede acceder a todos los módulos disponibles del

sistema. En la parte superior derecha está el logotipo del sistema, y en la parte derecha un menú desplegable asociado a la cuenta de usuario. En la parte media izquierda se puede observar una lista de los módulos disponibles con iconografía, y en la parte derecha se observa un resumen sobre los estados financieros a la fecha.

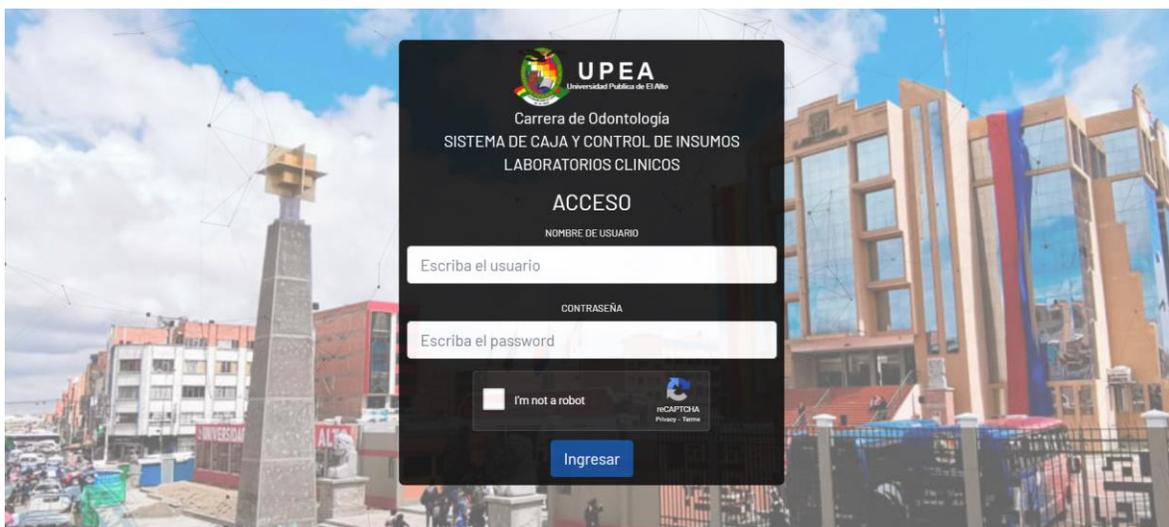


Figura 49. Formulario de acceso al sistema



Figura 50. Página de bienvenida al usuario

3.4.3.2. Gestión de unidades organizacionales

Para la gestión de unidades organizacionales se tienen cuatro escenarios que corresponde a: la lista de unidades organizacionales, el registro de nueva unidad organizacional, la actualización de una unidad organizacional y la baja de una unidad organizacional. En la Figura 51 se ilustra estas interfaces gráficas de usuario respectivas.

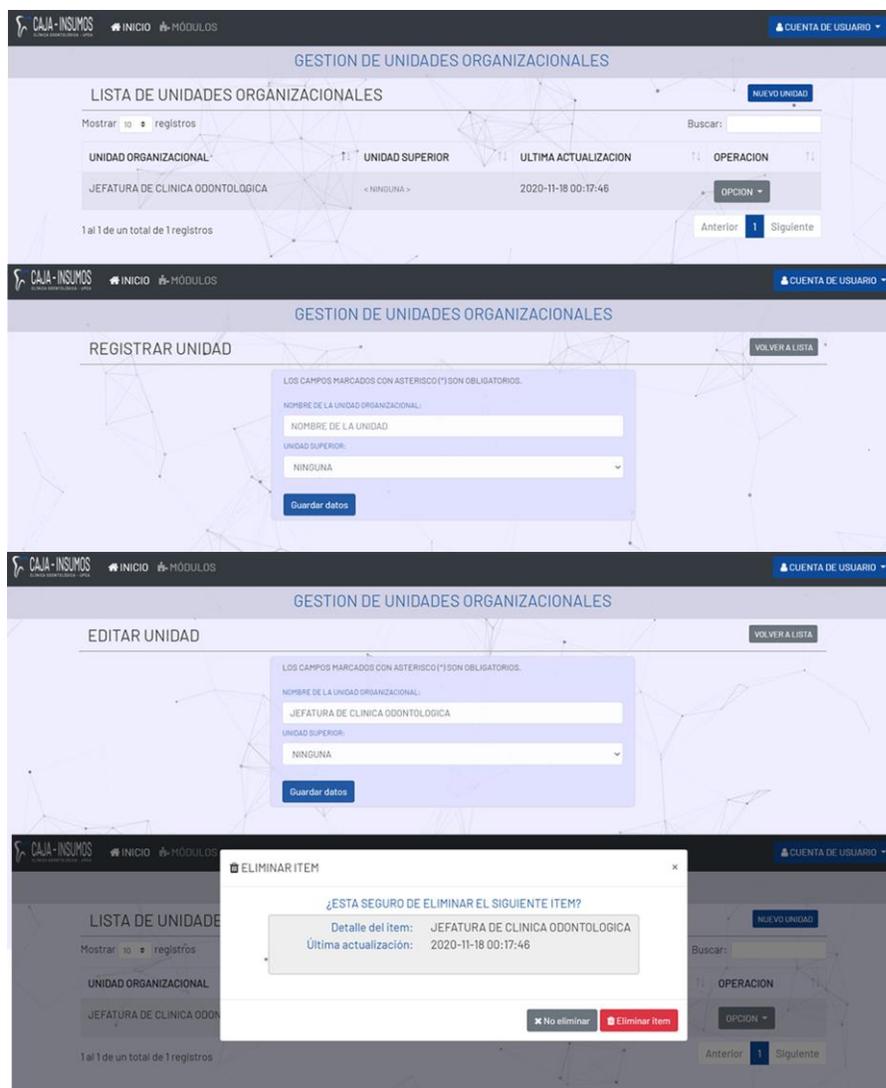


Figura 51. Interfaces gráficas para la gestión de unidades organizacionales

3.4.3.3. Gestión de cargos

Para la gestión de cargos se tienen cuatro escenarios que corresponde a: la lista de cargos, el registro de nuevo cargo, la actualización de datos de un cargo y la baja de un cargo. En la Figura 52 se ilustra estas interfaces gráficas de usuario respectivas.

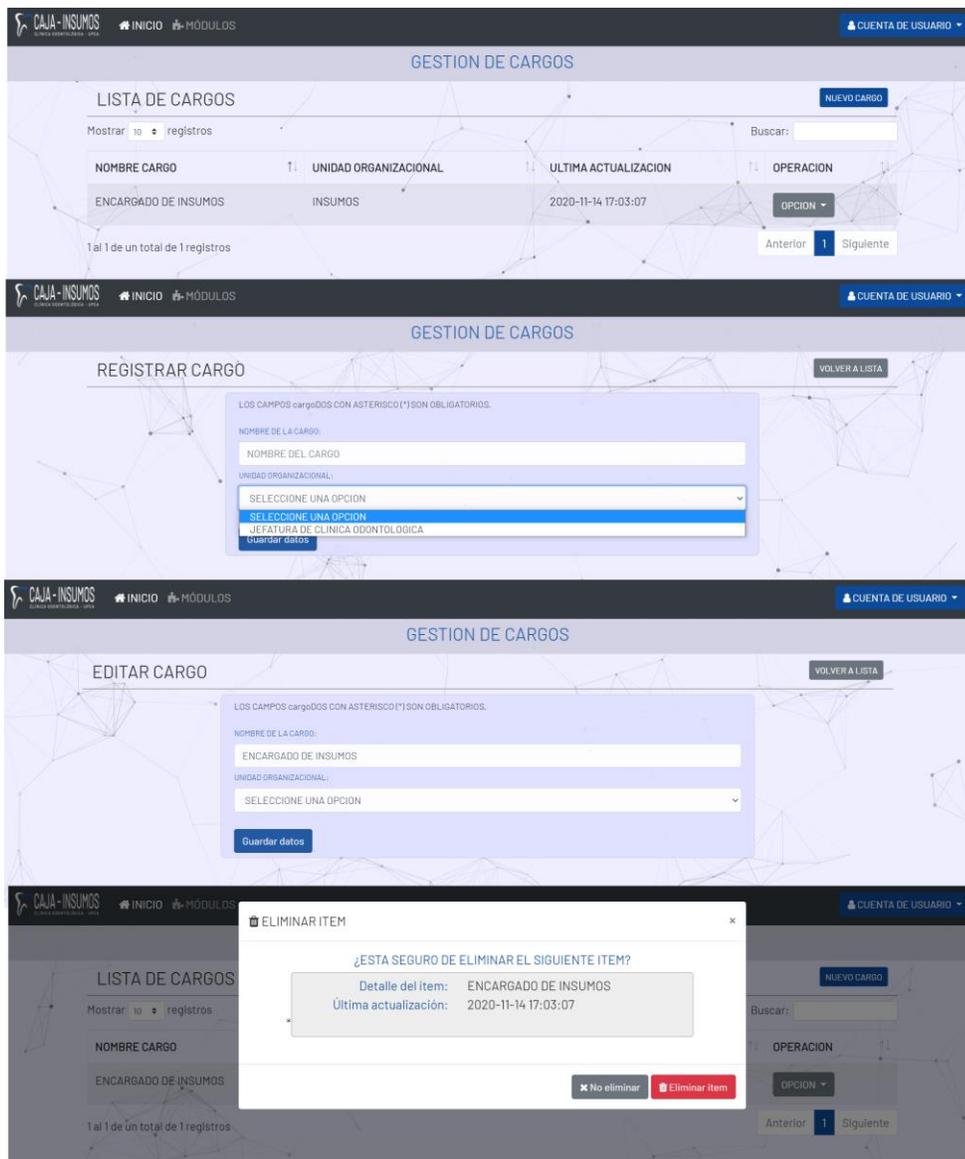


Figura 52. Interfaces gráficas para gestionar cargos

3.4.3.4. Gestión de usuarios

Para la gestión de usuarios se tienen cuatro escenarios que corresponde a: la lista de usuarios, el registro de nuevo usuario, la actualización de datos de un usuario y la baja de un usuario. En la Figura 53 se ilustra estas interfaces gráficas de usuario respectivas.

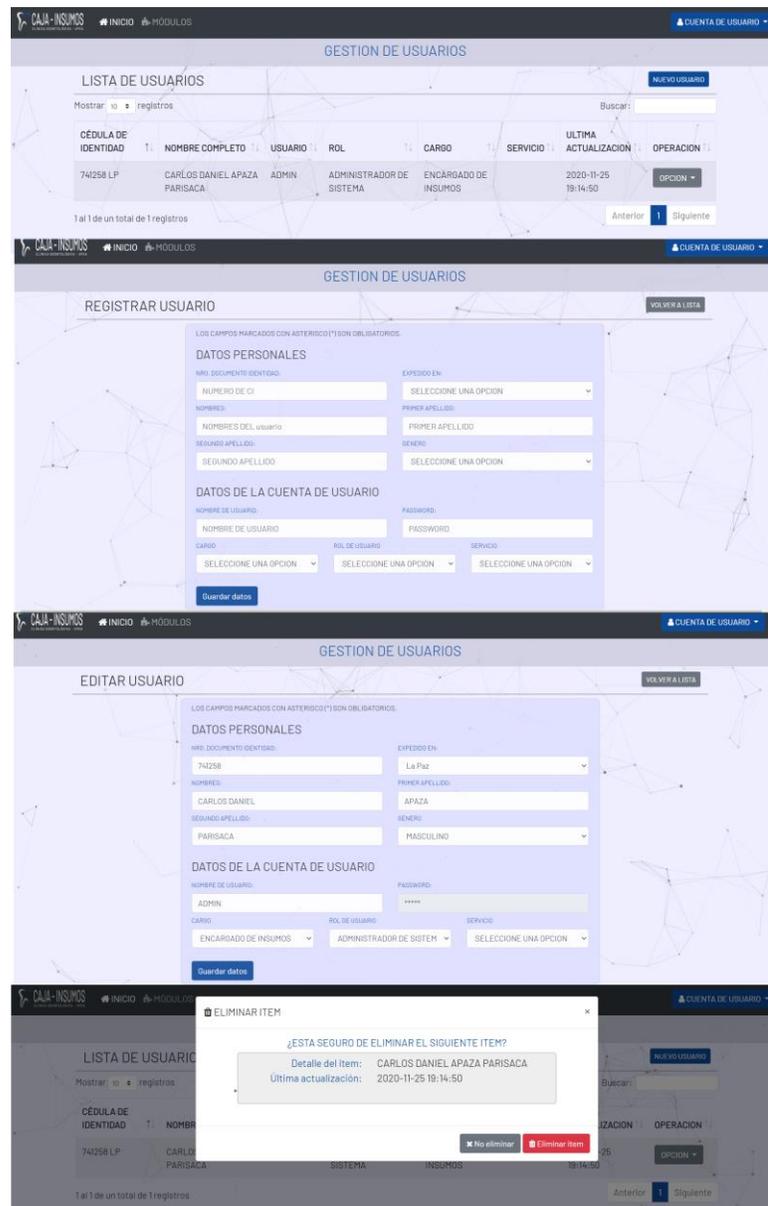


Figura 53. Interfaces gráficas para gestionar usuarios

3.4.3.5. Gestión de estudiantes

Para la gestión de estudiantes se tienen cuatro escenarios que corresponde a: la lista de estudiantes, el registro de nuevo estudiante, la actualización de datos de un estudiante y la baja de un estudiante. En la Figura 54 se ilustra estas interfaces gráficas de usuario respectivas.

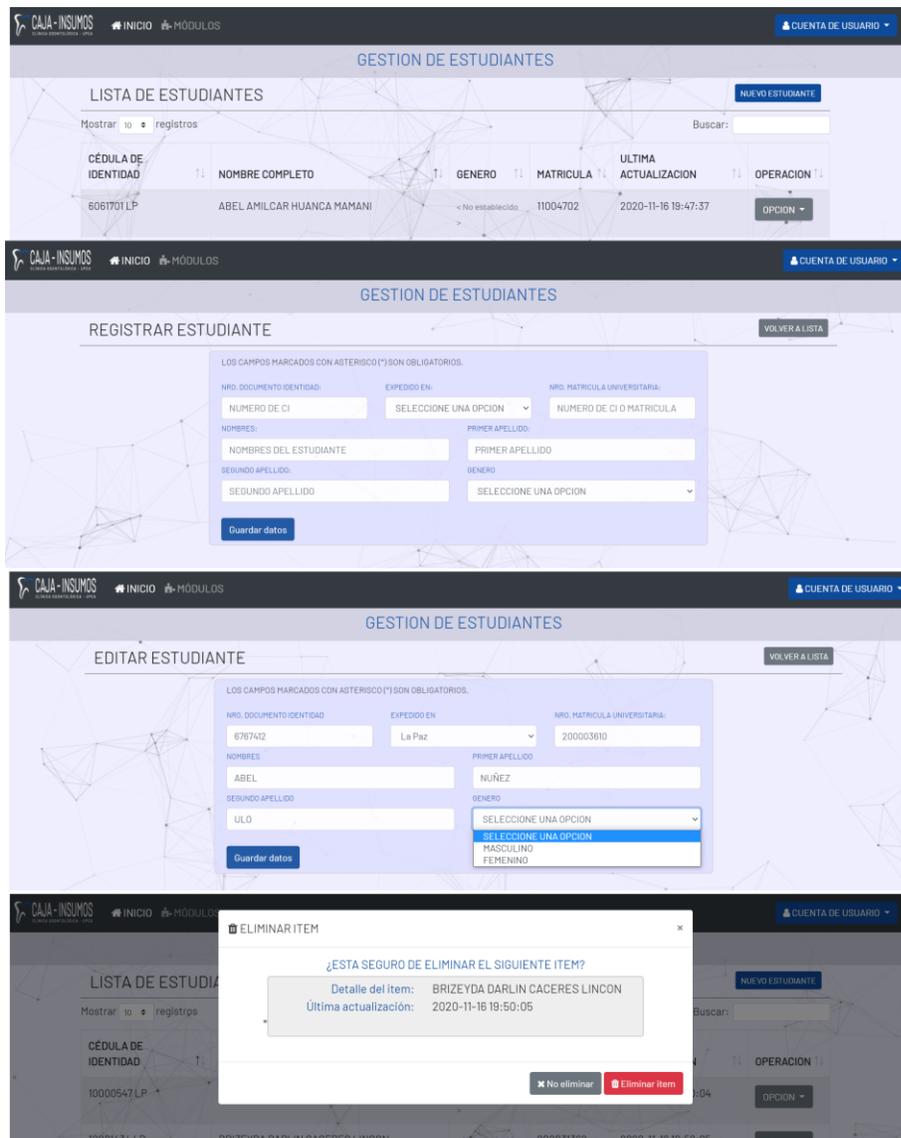


Figura 54. Interfaces gráficas para gestionar estudiantes

3.4.3.5. Gestión de docentes

Para la gestión de docentes se tienen cuatro escenarios que corresponde a: la lista de docentes, el registro de nuevo docentes, la actualización de datos de un docente y la baja de un docente. En la Figura 55 se ilustra estas interfaces gráficas de usuario respectivas.

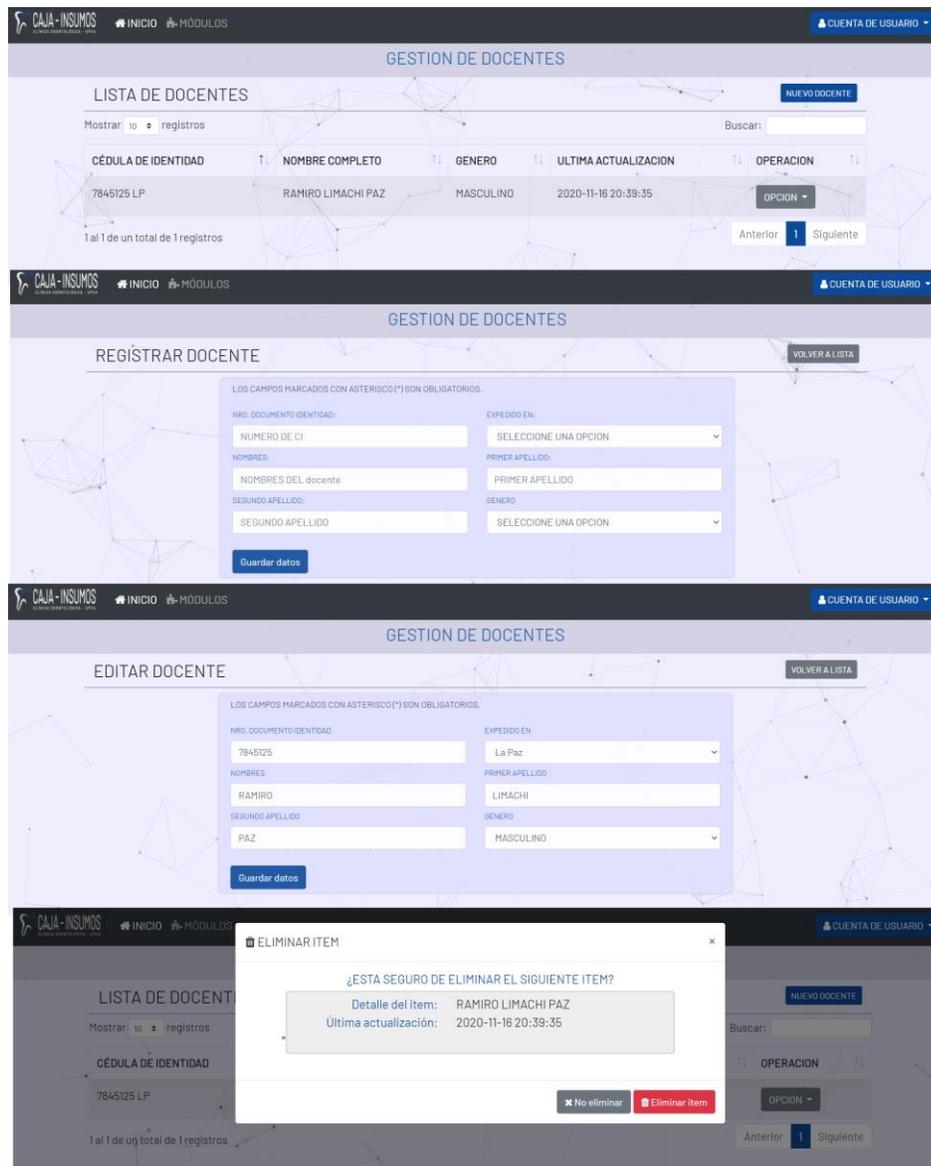


Figura 55. Interfaces gráficas para gestionar docentes.

3.4.3.6. Gestión de materias - especialidades

Para la gestión de materias - especialidades se tienen cuatro escenarios que corresponde a: la lista de materias, el registro de nueva materia, la actualización de datos de una materia y la baja de una materia. En la Figura 56 se ilustra estas interfaces gráficas de usuario respectivas.

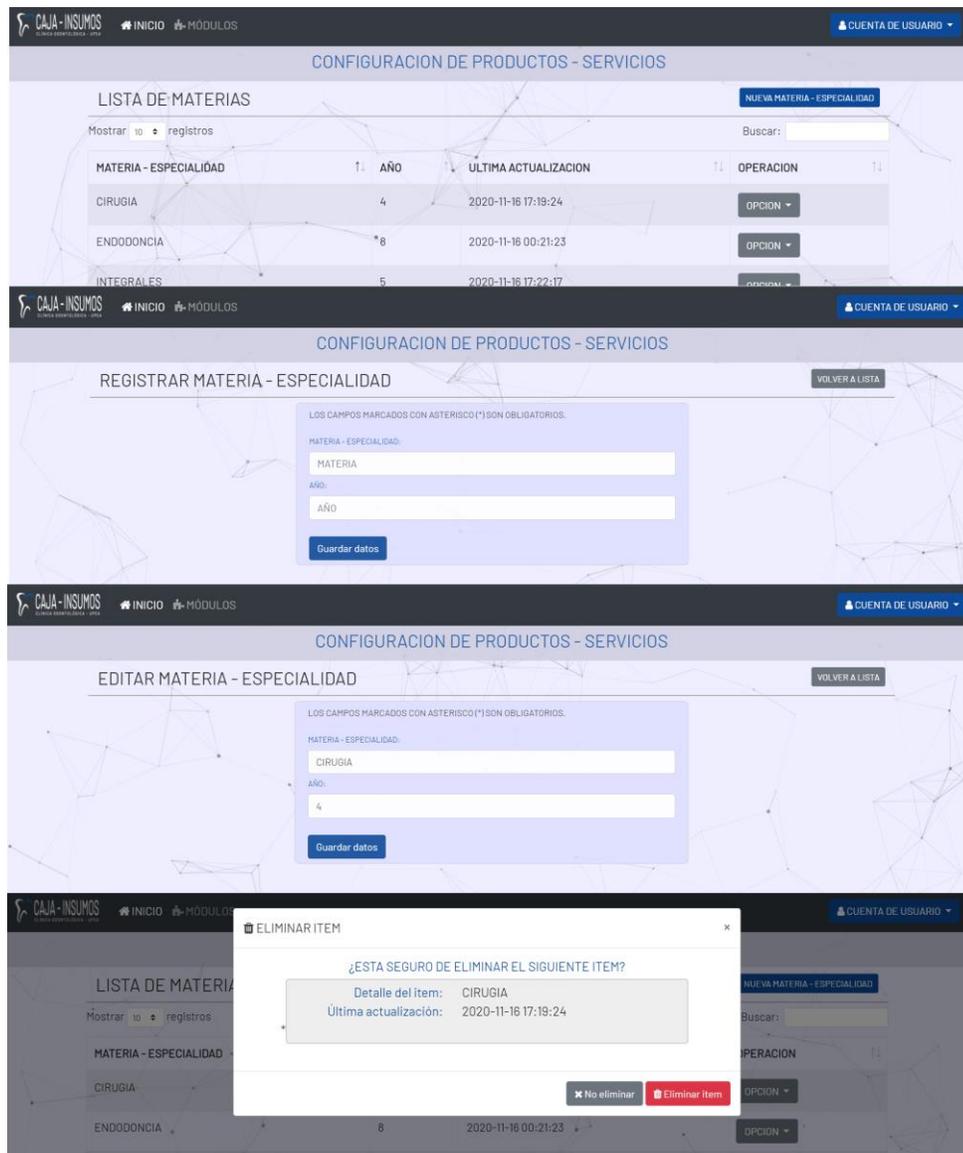


Figura 56. Interfaces gráficas para gestionar materias o especialidades

3.4.3.7. Gestión de tratamientos

Para la gestión de tratamientos se tienen cuatro escenarios que corresponden a: la lista de tratamientos, el registro de nuevo tratamiento, la actualización de datos de un tratamiento y la baja de un tratamiento. En la Figura 57 se ilustra estas interfaces gráficas de usuario respectivas.

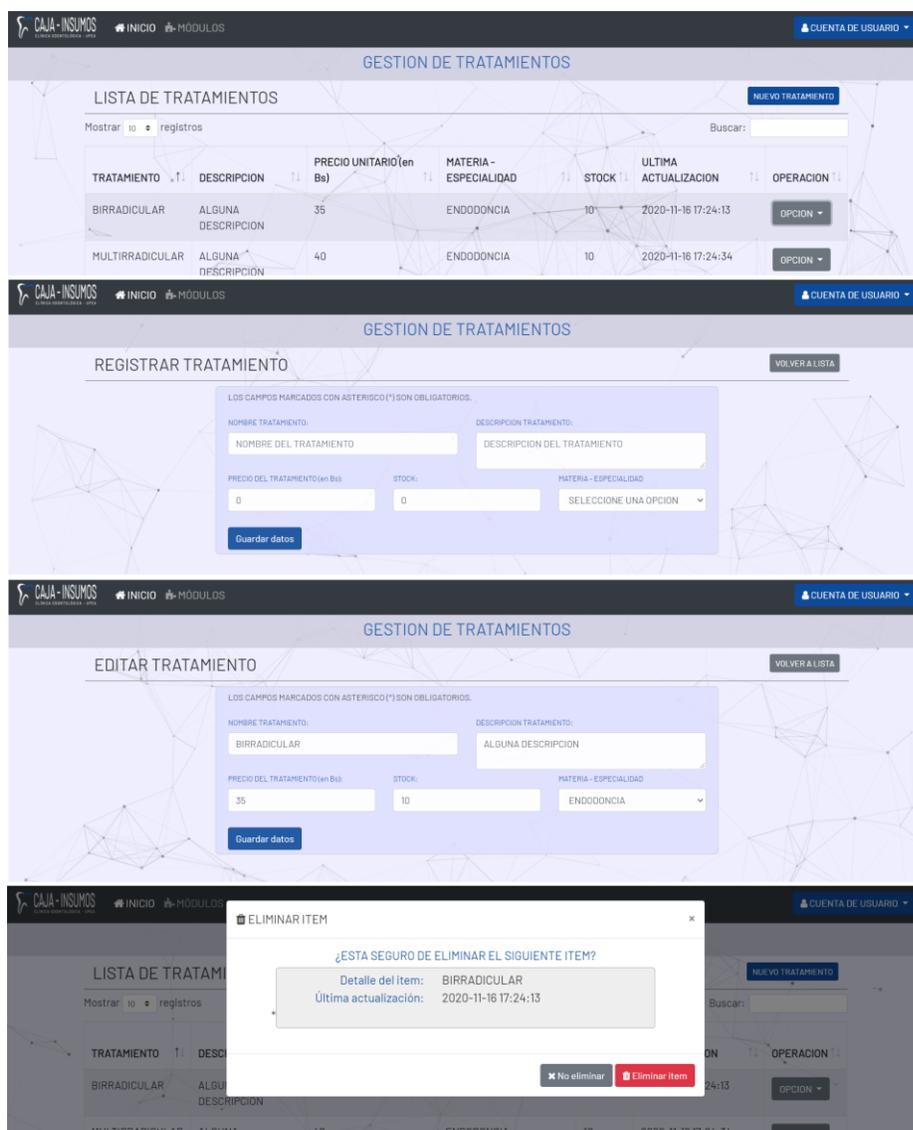


Figura 57. Interfaces gráficas para gestionar tratamientos

3.4.3.8. Gestión de formularios de historial clínico

Para la gestión de formularios de historial clínico se tienen cuatro escenarios que corresponden a: la lista de formularios, el registro de nuevo formulario, la actualización de datos de un formulario y la baja de un formulario. En la Figura 58 se ilustra estas interfaces gráficas de usuario respectivas.

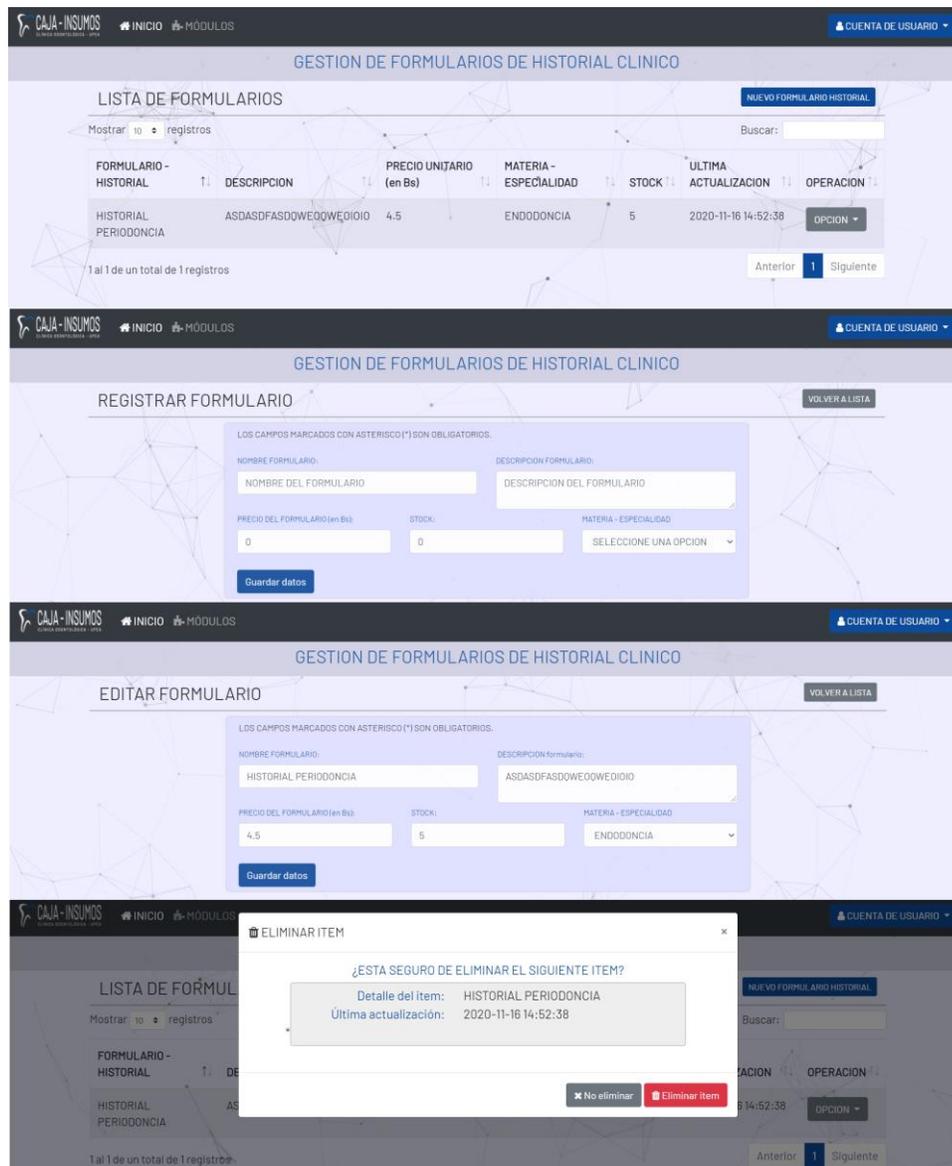


Figura 58. Interfaces gráficas para gestionar formularios de historial clínico

3.4.3.9. Gestión de marcas

Para la gestión de marcas se tienen cuatro escenarios que corresponden a: la lista de marcas, el registro de nueva marca, la actualización de datos de una marca y la baja de una marca. En la Figura 59 se ilustra estas interfaces gráficas de usuario respectivas.

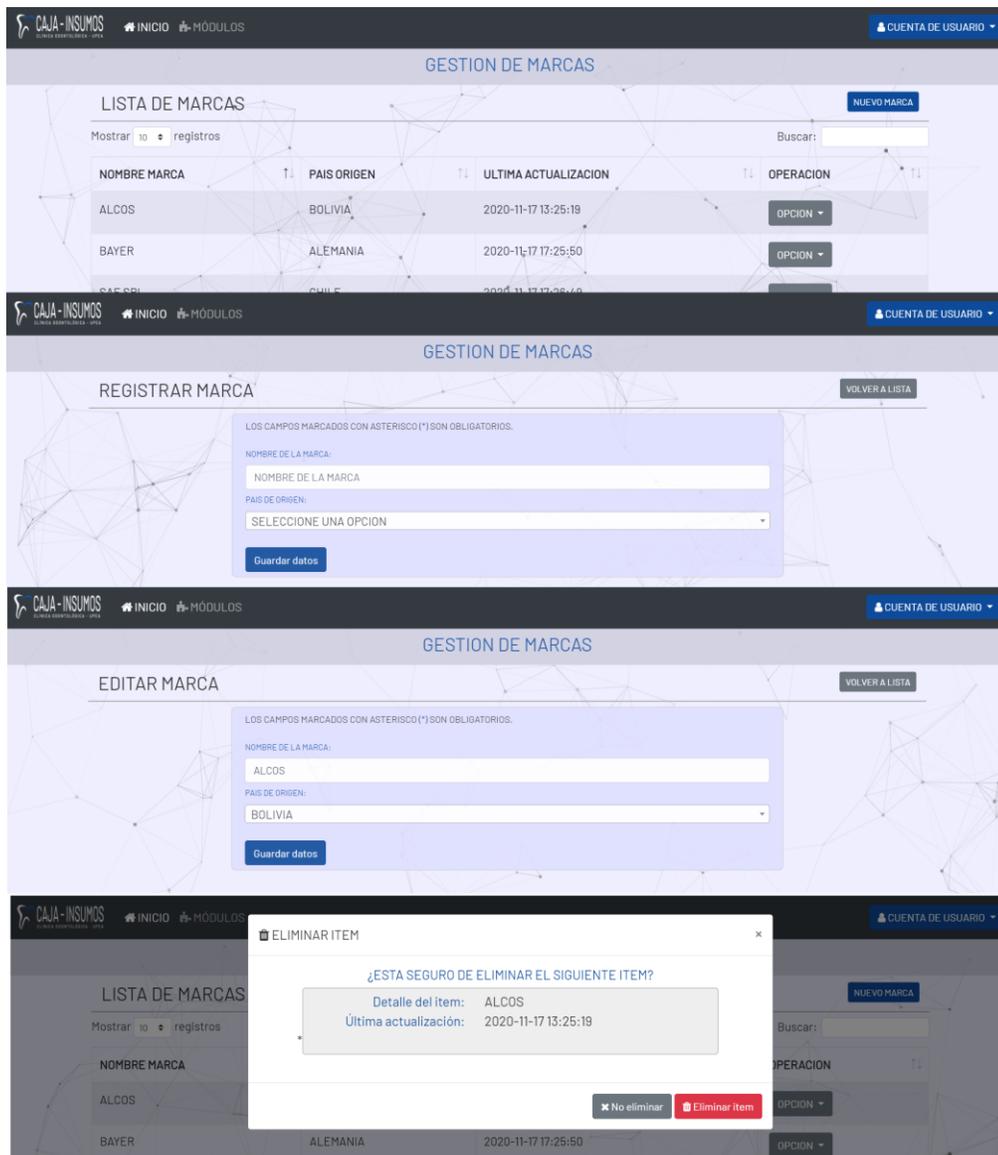


Figura 59. Interfaces gráficas para gestionar marcas de insumos médicos

3.4.3.10. Gestión de proveedores

Para la gestión de proveedores se tienen cuatro escenarios que corresponden a: la lista de proveedores, el registro de nuevo proveedor, la actualización de datos de un proveedor y la baja de un proveedor. En la Figura 60 se ilustra estas interfaces gráficas de usuario respectivas.

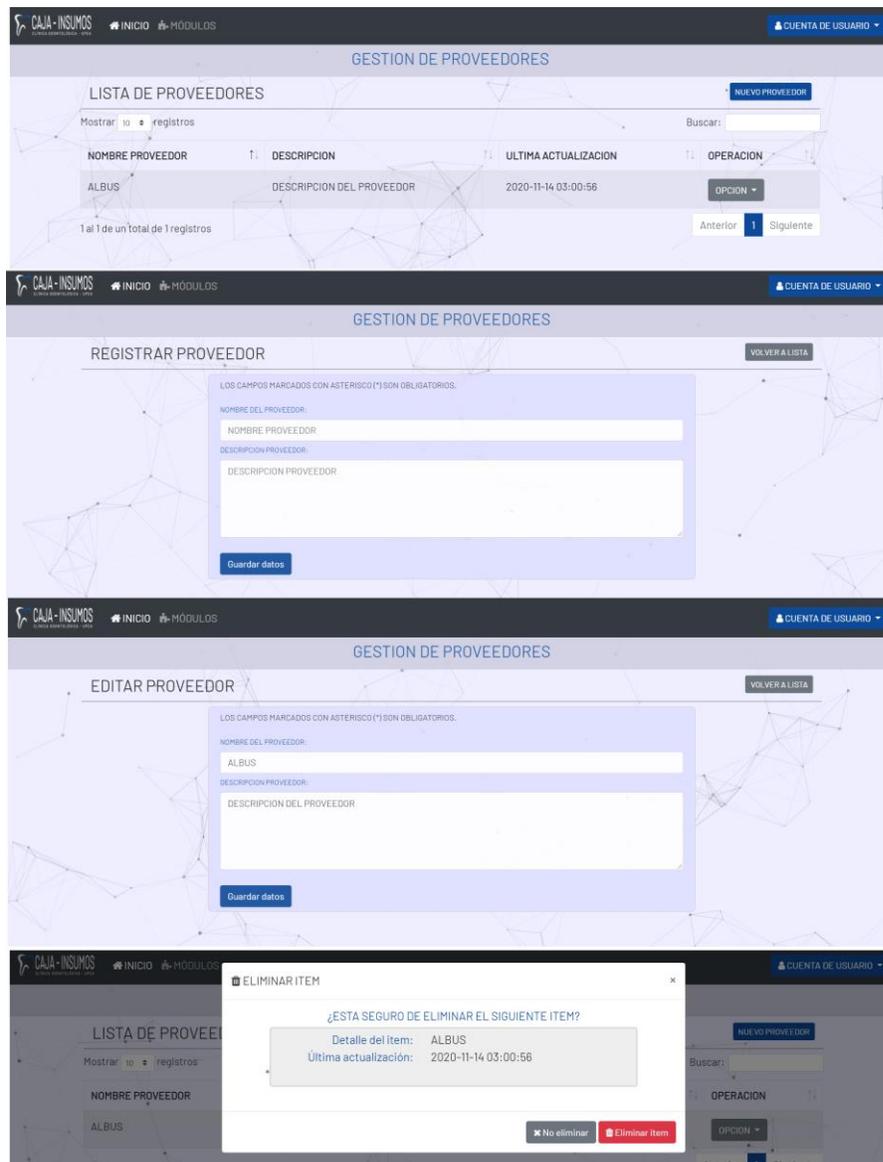


Figura 60. Interfaces gráficas para gestionar proveedores de insumos

3.4.3.11. Gestión de unidades de medida

Para la gestión de unidades de medida se tienen cuatro escenarios que corresponden a: la lista de unidades de medida, el registro de nueva unidad de medida, la actualización de datos de una unidad de medida y la baja de una unidad de medida. En la Figura 61 se ilustra estas interfaces gráficas de usuario respectivas.

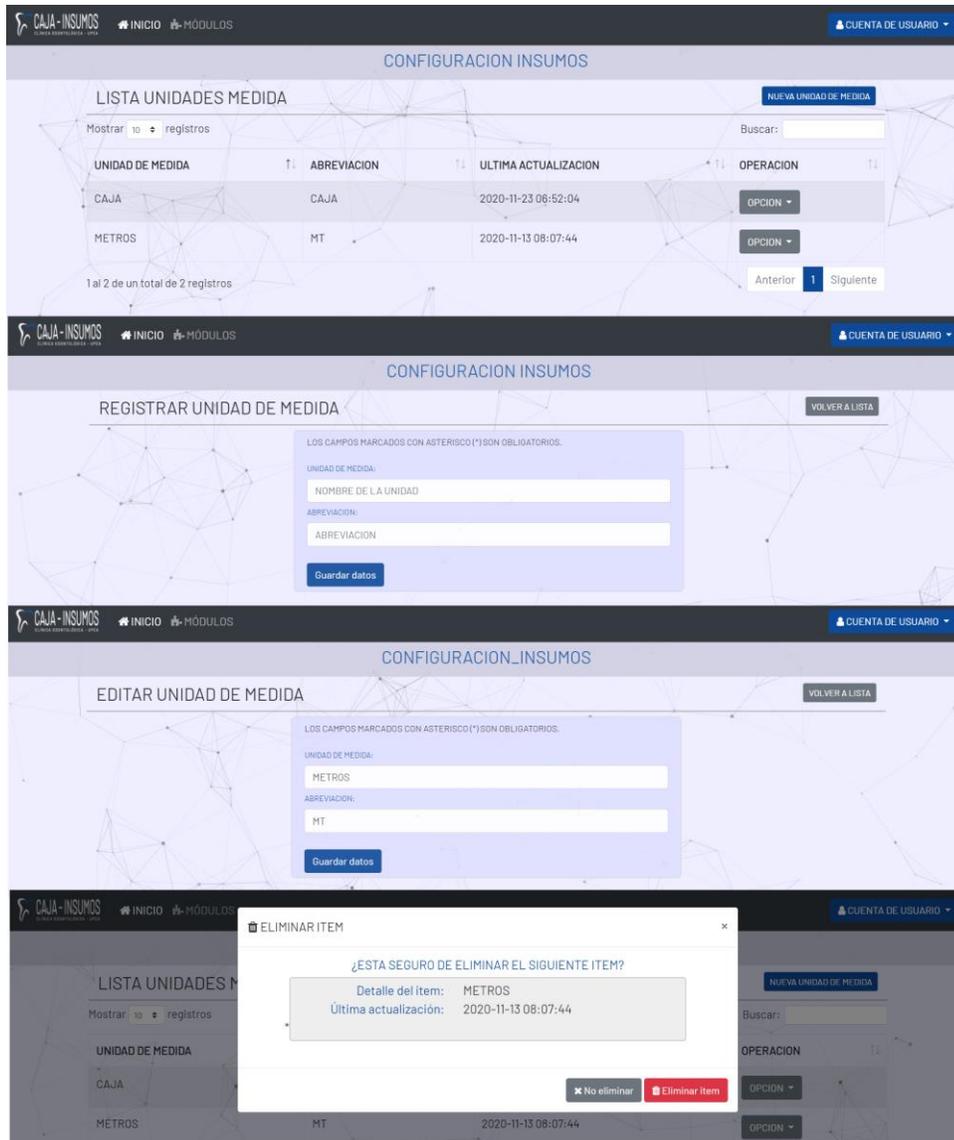


Figura 61. Interfaces gráficas para gestionar unidades de medida

3.4.3.12. Gestión de insumos

Para la gestión de insumos se tienen cuatro escenarios que corresponden a: la lista de insumos, el registro de nuevo insumos, la actualización de datos de un insumo y la baja de un insumo. A su vez, se tiene los escenarios para el registro de entradas y salidas de insumos. En la Figura 62 se ilustra estas interfaces gráficas de usuario respectivas.

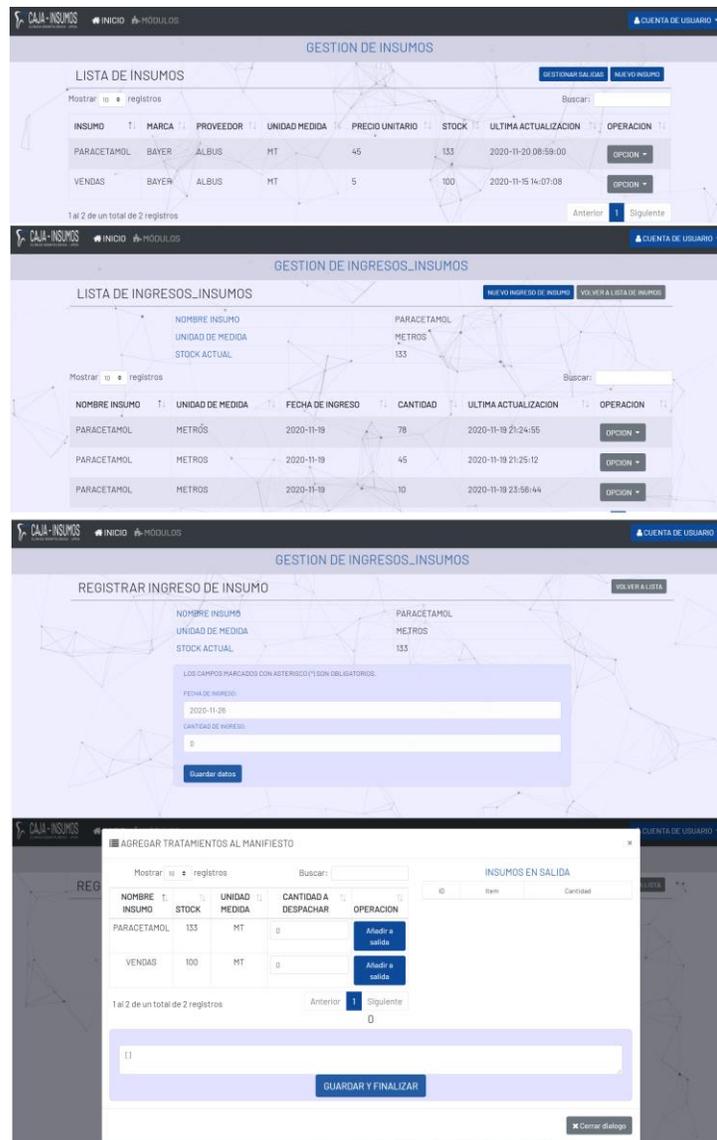


Figura 61. Interfaces gráficas para gestionar los insumos, sus ingresos y salidas

3.4.3.13. Gestión de pedidos y recibos

Para la gestión de marcas se tienen cuatro escenarios que corresponden a: la lista de recibos, el registro de nuevo recibo y pedido, y la baja o anulación de un recibo. En la Figura 63 se ilustra estas interfaces gráficas de usuario usadas por el encargado de caja.

The figure displays three screenshots of the 'Gestion de Pedidos y Recibos' system interface. The top screenshot shows a 'LISTA DE RECIBOS' table with columns for code, student, patient, date, total, and last update. The middle screenshot shows the 'REGISTRAR RECIBO' form with sections for patient data, authorization, and a 'DETALLE DEL PEDIDO' table. The bottom screenshot shows the 'IMPRIMIR RECIBO' interface with a receipt header, patient information, a QR code, and a detailed item list.

CODIGO RECIBO	ESTUDIANTE	PACIENTE	FECHA RECIBO	TOTAL RECIBO	ULTIMA ACTUALIZACION	OPERACION
RC-2-2020	MARIA LUISA APAZA CHOOUE	JUAN CARLOS POMIER ROCHA	2020-11-17	70	2020-11-17 12:57:21	OPCION
RC-3-2020	REYNILDA FLORES LEQUELEQUE	FABIOLA MEJIA MORALES	2020-11-17	105	2020-11-17 17:56:59	OPCION
RC-4-2020	REYNILDA FLORES LEQUELEQUE	GUALBERTO POMACUSI RIVERO	2020-11-17	44.5	2020-11-17 18:19:22	OPCION

ID	Item	Precio Unitario (en Bs)	Cantidad	Subtotal (en Bs)
FORMULARIOS				
ID	Item	Precio Unitario (en Bs)	Cantidad	Subtotal (en Bs)
TOTAL RECIBO (Bs): 0				

Item	Precio Unitario (en Bs)	Cantidad	Subtotal (en Bs)
BIRRADICULAR	35	2	70
TOTAL RECIBO Bs.- 70			

Figura 63. Interfaces gráficas para gestionar recibos

3.4.3.14. Gestión de reportes

Para la gestión de reportes se presentan varios escenarios que corresponden a los reportes requeridos. En la Figura 64 se ilustra estas interfaces gráficas de usuario correspondiente al listado de reportes y el reporte general de ingresos.

The figure consists of two screenshots of a web application interface for report management. The top screenshot shows a list of financial reports under the heading 'GESTION DE REPORTES'. The list has columns for '#', 'NOMBRE DEL REPORTE', and 'OPERACION'. There are four rows of reports, each with a 'GENERAR REPORTE' button. The bottom screenshot shows a detailed report titled 'REPORTE INGRESO ECONOMICO GENERAL (POR DIA)'. It includes the university logo and name: 'UNIVERSIDAD PÚBLICA DE EL ALTO - UPEA', 'Carrera de Odontología', and 'CLINICA ODONTOLÓGICA'. The report details include: 'REPORTE: REPORTE INGRESO ECONOMICO GENERAL (POR DIA)', 'GENERADO CON DATOS DE FECHA: 2020-11-17', and 'FECHA Y HORA EMISION DE REPORTE: 2020-11-26, 13:11:24'. A table lists receipts with columns 'CODIGO RECIBO' and 'MONTO (en Bs)'. The total revenue is 219.5.

CODIGO RECIBO	MONTO (en Bs)
RC-2-2020	70
RC-3-2020	105
RC-4-2020	44.5
TOTAL RECAUDADO:	219.5

Figura 64. Interfaces gráficas para gestionar unidades de medida

3.5. Pruebas

Las pruebas del producto software son un conjunto de actividades dentro del proceso de desarrollo de aplicaciones, cuyo objetivo es proporcionar información sobre la calidad del producto en términos de: funcionalidad, integración, compatibilidad, seguridad, portabilidad, entre otros (Meucii & Muller, 2016). Los

requerimientos del software ser revisan usando técnicas de diseño de casos de prueba de caja negra (Pressman, 2010).

Esmite, Farías, Farías & Perez (2007) define un método para efectuar pruebas de software que se resumen básicamente en tres fases, como ser: a) definición de casos de prueba, ejecución de casos de prueba y evaluación de los casos de prueba. Un caso de prueba debe estar enfocado en un elemento de prueba, y un resultado esperado. Gutierrez (2006) recomienda elaborar y aplicar los casos de prueba con base a todos casos de uso del software y los requerimientos no funcionales. En la Tabla 8 se define un conjunto de casos de prueba que deben ser aplicados al sistema.

Tabla 8 Definición de casos de prueba de caja negra

Nro	Caso de prueba	Elemento de prueba	Indicador	Resultado esperado
1	Verificar el estado de los enlaces	Todos los enlaces	Nro de enlaces correctos	Enlace no roto y que direcciona correctamente a su destino
2	Verificar que la interfaz gráfica que muestra el software coincide con la referencia y datos que indica el enlace	Todas las páginas	Nro de páginas que coinciden con lo indicado en el enlace que los referencia	Interfaz gráfica que coincide con lo indicado en el enlace de referencia
3	Verificar que todos los campos requeridos son validados correctamente en los formularios	Todos los formularios	Nro de formularios que validan correctamente campos requeridos	Formulario que valida correctamente los campos requeridos
4	Verificar que todos los campos en los formularios no sean nulos o vacios	Todos los formularios	Nro de formularios que validan correctamente campos nulos o vacios	Formulario que valida correctamente campos vacios
5	Verificar que los datos de los formularios se envían correctamente	Todos los formularios	Nro de formularios que envían correctamente datos	Formulario que envía correctamente datos

6	Verificar que los datos de los formularios se guardan correctamente en la base de datos.	Todos los formularios	Nro de formularios que guardan correctamente datos en la base de datos	Formularios que guardan correctamente datos a la base de datos.
7	Verificar que todos los ítems en las listas de ítems se muestran correctamente los datos	Todas las listas de ítems	Nro de listas de ítems que muestran correctamente los datos	Lista de ítems que muestran correctamente sus datos
8	Verificar que todos los reportes se generan correctamente	Todos los reportes parametrizados	Nro de reportes que se generan correctamente	Reportes que se generan conforme a los parámetros establecidos por el usuario
9	Verificar que todos los reportes conservan un formato adecuado para impresión	Todos los reportes con opción a impresión	Nro de reportes que conservan el formato adecuado para impresión	Reportes que conservan el formato de impresión
10	Verificar que la respuesta promedio de carga de las paginas es menor a 5 segundos.	Todas las páginas del sistema	Nro de páginas que responden en menos de 5 segundos	Páginas que responden en menos de 5 segundos
11	Verificar que los recursos del proyecto estén optimizados	Todos los recursos que no pertenecen al código de la lógica del negocio	Nro de recursos optimizados	Recursos optimizados para agilizar la carga de la página.
12	Verificar que las peticiones al servicio Google Recaptcha se efectúan correctamente	Todas las funciones que realizan peticiones al servicio Google Recaptcha	Nro de funciones que realizan correctamente peticiones al servicio Google Recaptcha	Funciones o métodos que realizan correctamente peticiones al servicio de Google Recatpcha
13	Verificar que el sistema emita mensajes informativos al operar registros la base de datos	Todos los formularios	Nro de formularios que emiten mensajes informativos al operar con registros base de datos	Formulario que emite mensajes informativos para el usuario
14	Verificar que la descripción de los campos de	Todos los formularios	Nro de formularios que describen	Formulario que describe correctamente sus campos

	formulario sean comprensibles		correctamente los campos de formulario	
--	-------------------------------	--	--	--

Una vez definidos los casos de prueba, se efectúan las pruebas al software en reiteradas oportunidades con el fin de mejorar la calidad. Los resultados en una tercera iteración de las pruebas se muestran en la Tabla 9.

Tabla 9 Resultados tras haber efectuado los casos de prueba de caja negra después de la tercera iteración o revisión del sistema

Nro	Caso de prueba	Indicador	Valor referencial ¹	Valor obtenido	% de cumplimiento
1	Verificar el estado de los enlaces	Nro de enlaces correctos	104	104	100%
2	Verificar que la interfaz gráfica que muestra el software coincide con la referencia y datos que indica el enlace	Nro de páginas que coinciden con lo indicado en el enlace que los referencia	106	106	100%
3	Verificar que todos los campos requeridos son validados correctamente en los formularios	Nro de formularios que validan correctamente campos requeridos	43	42	97.6%
4	Verificar que todos los campos en los formularios no sean nulos o vacíos	Nro de formularios que validan correctamente campos nulos o vacíos	43	43	100%
5	Verificar que los datos de los formularios se envían correctamente	Nro de formularios que envían correctamente datos	43	43	100%
6	Verificar que los datos de los formularios se guardan correctamente en la base de datos.	Nro de formularios que guardan correctamente datos en la base de datos	43	43	100%

¹ Corresponde al valor obtenido al hacer el conteo de los elementos que son sometidos a la prueba, por ejemplo: la cantidad de enlaces en toda la aplicación Web.

7	Verificar que todos los ítems en las listas de ítems se muestran correctamente los datos	Nro de listas de ítems que muestran correctamente los datos	16	16	100%
8	Verificar que todos los reportes se generan correctamente	Nro de reportes que se generan correctamente	12	12	100%
9	Verificar que todos los reportes conservan un formato adecuado para impresión	Nro de reportes que conservan el formato adecuado para impresión	12	12	100%
10	Verificar que los recursos del proyecto estén optimizados	Nro de recursos optimizados	65	61	93.8%
11	Verificar que las peticiones al servicio Google Recaptcha se efectúan correctamente	Nro de funciones que realizan correctamente peticiones al servicio Google Recaptcha	1	1	100%
12	Verificar que el sistema emita mensajes informativos al operar registros la base de datos	Nro de formularios que emiten mensajes informativos al operar con registros base de datos	43	43	100%
13	Verificar que la descripción de los campos de formulario sean comprensibles	Nro de formularios que describen correctamente los campos de formulario	43	43	100%

Los casos de prueba aplicados hasta el momento tienen como finalidad medir características como la adecuación funcional, la usabilidad, fiabilidad y la compatibilidad. Para probar la eficiencia de desempeño y determinar un tiempo de respuesta del software, se hace uso de servicios especializados, como ser: Webtestpage, GMetrix y PageSpeed de Google que determinan el desempeño y

tiempo de respuesta de aplicaciones Web aplicando técnicas de prueba de caja negra automatizadas.

Para probar el desempeño, se evaluarán tres indicadores, como ser: a) la primera impresión de la página o First Contentful Paint (FCP), que hace referencia al tiempo que transcurre desde el momento en que se solicita la página Web hasta que aparece en pantalla el primer fragmento de contenido; b) la impresión más larga de la página o Largest Contentful Paint (LCP), que hace referencia al cargado de la página Web con los recursos más grandes disponibles y c) el tiempo de interactivo o Time to interactive (TTI), que hace referencia al tiempo que tarda una página Web en ser visible e interactivo para el visitante (Google Developers, 2020). Los resultados obtenidos se muestran en la Tabla 10.

Tabla 10 Resultados de las pruebas de desempeño del sistema muestran un tiempo de respuesta menor a 5 segundos como esta especificado en los requerimientos

Indicador	Google PageSpeed (en segundos)	GMetrix (en segundos)	Webtestpage (en segundos)	Promedio (en segundos)
FCP	0.3	0.36	0.9	0.52
LCP	1.2	0.8	1.6	1.2
TTI	1.3	0.9	3.2	1.45

CAPITULO 4 CALIDAD Y SEGURIDAD

Resumen

En este capítulo se desarrolla la evaluación de la calidad al producto software con base al estándar ISO/IEC 9126. Asimismo, se procede con la descripción de las medidas de seguridad aplicadas al sistema y también se muestra la evaluación de seguridad Web del sistema con base al estándar de verificación de seguridad en aplicaciones Web de OWASP.

4.1. Evaluación de la calidad

La evaluación de calidad del sistema se realiza con base al estándar ISO/IEC 9126, el cual define el modelo de calidad de producto software. Se han evaluado cuatro dimensiones del modelo de calidad, que son: funcionalidad, confiabilidad, mantenibilidad y usabilidad (Anexo D). Se procedió con la evaluación de usabilidad aplicando un cuestionario a dos (2) actores que intervienen en los casos de uso identificados, como ser: encargado de caja y encargado de entrega de insumos. Además, se aplicó el cuestionario a un (1) informático del área de TIC con la finalidad de obtener un punto de vista neutral respecto al negocio. Los resultados de la evaluación aplicada se presentan en la Tabla 11. Los resultados en promedio dan un resultado promedio de 89.7 sobre 100.

Tabla 11 Resultados de la evaluación de calidad del sistema realizado con base al estándar ISO/IEC 9126.

DIMENSION DE CALIDAD	EVALUACION SOBRE EL 100%
Funcionalidad	85.9
Confiabilidad	84
Mantenibilidad	95
Usabilidad	94
PROMEDIO	89.7

4.2. Medidas de seguridad del sistema

4.2.1. Autenticación

El método de autenticación del sistema emplea de un nombre de usuario, una contraseña y un código captcha² como está establecido en los requerimientos y así para identificar a cada usuario durante su acceso. El nombre usuario es un valor alfanumérico único³ en la base de datos, mientras que contraseña hace uso del algoritmo HASH Bcrypt que genera una cadena de 60 caracteres, por tanto las contraseñas no se almacenan en texto plano y no son legibles para los humanos en caso de intrusión a la base de datos.

Por otra parte, se hace uso del servicio Google reCaptcha para evitar el acceso de programas maliciosos o robots al sistema. Para hacer uso de este servicio el sistema se conecta con una interfaz de Google a través de un token y contraseña para la validación del código enviado por el usuario al verificar la casilla No soy robot.

4.2.2. Control de acceso basado en roles

El control de acceso basado en roles es una función de seguridad para controlar el acceso de usuarios a tareas o funcionalidades del sistema. El sistema hace uso de este tipo de control, otorgando y restringiendo el acceso a funciones con base a las interacciones que un actor tiene con los casos de uso. Por tanto, para el sistema hay una equivalencia entre actor y rol en el sistema, por lo cual se ha definido un mapa de control de accesos en el Anexo E.

4.2.3. Políticas de seguridad de la base de datos

- Como política de seguridad, no se debe conceder acceso a la base de datos a otro usuario que no sea el sistema. El usuario otorgado al sistema no debe tener

² Son las siglas de Completely Automated Public Turing test to tell Computers and Humans apart, son pruebas desafío-respuesta controladas por maquinas que se utilizan para determinar si un usuario es humano o un programa automático.

³ Este atributo se define como UNIQUE en la base de datos

permisos para modificar la estructura de la base de datos debe tener permitido borrado físico de registros.

- Con el fin de garantizar la continuidad del negocio, se debe realizar una copia de respaldo manual de la base de datos cada semana. Sin embargo, si se pudiese acortar este tiempo, sería adecuado para no perder información útil del negocio.
- El sistema registra información acerca de las operaciones realizadas sobre los registros de la base de datos, almacenando las fechas de creación, actualización, eliminación de cada registro y también el usuario del sistema que realiza cada acción.

4.3. Evaluación de seguridad Web

La evaluación de seguridad del sistema se realiza con base al Estándar de Verificación de Seguridad de Aplicaciones Web 3.0.1 publicado en idioma español en abril de 2017 por la organización internacional OWASP la cual se dedica a la seguridad informática. Como se describió en anteriores secciones, el estándar define tres niveles de evaluación del software, donde el nivel 1 apunta a todo tipo de software, el nivel 2 a software con información muy sensible y el nivel 3 a software con niveles críticos. La evaluación de seguridad realizada al sistema apunta a un nivel 1 de la escala del estándar, donde se analiza si el software responde de manera adecuada contra vulnerabilidades de seguridad que incluyen el Top10 de vulnerabilidades de OWASP. Para esto, se hace uso de la herramienta Zed Attack Proxy versión 2.9, la cual permite realizar pruebas automáticas de seguridad. Los resultados del reporte emitido por la mencionada herramienta se muestran en la Figura 65. El reporte detallado se muestra en el Anexo F.

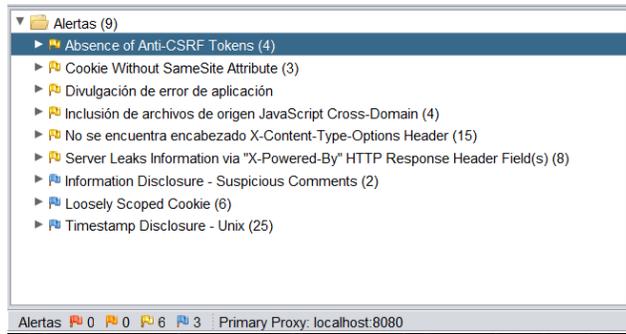


Figura 65 Resultados obtenidos tras analizar la seguridad del sistema con Zed Attack Proxy

CAPITULO 5 COSTOS Y BENEFICIOS

Resumen

En este capítulo se realiza la estimación del costo de software con base al método puntos de casos de uso propuesto por Karner y Jacobson. En primera instancia se procedió con la ponderación de actores y casos de uso, y posteriormente se procedió con el cálculo de los indicadores utilizados en el método, con los cuales se obtiene el esfuerzo de desarrollo del software. Por último, se realiza el cálculo del costo con base a un salario mínimo de 500 USD y también con base al salario mínimo nacional.

5.1. Estimación de costos con base al método de Puntos de Casos de Uso

Para realizar la estimación del costo de software con base al método de puntos de casos de uso en primer lugar se debe tener identificado los actores y los casos de uso del sistema, los cuales ya se han sido identificados en el capítulo 3 del presente proyecto empleando el método UWE.

El primer indicador que se va a calcular es el Factor de peso de los actores sin ajustar (UAW), que se obtiene de haber identificado la complejidad de interacción de cada actores, donde simple corresponde a otro sistema que interactúa con el software a través de una API, medio corresponde a otro sistema interactuando a través de TCP/IP o bien un humano interactuando en modo texto, y complejo corresponde a una persona interactuando con el sistema mediante una interfaz gráfica (Diaz & Robiolo, 2014; Lencina et al., 2016). La ponderación para cada uno de estos actores es de 1 para un actor simple, 2 para un actor medio, y 3 para un actor complejo. En la Tabla 12 se muestra el resultado calculado.

Tabla 12 Cálculo del Factor de peso de los actores sin ajustar

Tipo de actor	(A) # de actores	(B) Ponderación	A*B
Simple	1	1	1
Medio	0	2	0
Complejo	4	3	12
UAW = SUMATORIA (A*B)			13

El segundo indicador que se va a calcular es el Factor de peso de los casos de uso sin ajustar (UUCW), que se obtiene de haber identificado la complejidad de interacción de cada caso de uso, donde simple corresponde a un caso de uso con menos de 3 transacciones, medio corresponde a un caso de uso con 4 a 7 transacciones, y complejo corresponde a un caso de uso con más de 7 transacciones (Díaz & Robiolo, 2014; Lencina et al., 2016). La ponderación para cada uno de estos actores es de 5 para un caso de uso simple, 10 para un caso de uso de nivel medio, y 15 para un caso de uso complejo. En la Tabla 13 se muestra el resultado calculado.

Tabla 13 Cálculo del Factor de peso de los casos de uso sin ajustar

Tipo de caso de uso	(A) # de casos de uso	(B) Ponderación	A*B
Simple	100	5	500
Medio	1	10	10
Complejo	0	15	0
UUCW = SUMATORIA (A*B)			510

Hasta aquí, se puede calcular los puntos de casos de uso sin ajustar mediante la fórmula $UUCP = UAW + UUCW$, obteniendo un valor de 523. Luego se procede a evaluar los factores técnicos del software como indica Lencina et al. (2016). La evaluación de los factores técnicos realizada al sistema se muestra en la Tabla 14.

Tabla 14 Cálculo de los factores técnicos

Factor	Descripción	(A) Ponderación	(B) Evaluación	A*B
T1	Sistema distribuido.	2	0	0
T2	Objetivos de performance o tiempo de respuesta.	1	3	3
T3	Eficiencia del usuario final.	1	5	5
T4	Procesamiento interno complejo.	1	3	3
T5	El código debe ser reutilizable.	1	3	3
T6	Facilidad de instalación.	0.5	3	1.5

T7	Facilidad de uso.	0.5	4	2
T8	Portabilidad.	2	3	6
T9	Facilidad de cambio.	1	5	5
T10	Concurrencia.	1	3	3
T11	Incluye objetivos especiales de seguridad.	1	3	3
T12	Provee acceso directo a terceras partes.	1	0	0
T13	Se requiere facilidades especiales de entrenamiento a usuario.	1	1	1
TFactor = Sumatoria(A*B)				35.5
TCF = 0.6 + (0.01*TFactor)				0.955

A continuación, se procede a evaluar los factores ambientales del software como indica Lencina et al. (2016). La evaluación de los factores ambientales realizada al sistema se muestra en la Tabla 15.

Tabla 15 Cálculo de los factores ambientales

Factor	Descripción	(A) Ponderación	(B) Evaluación	A*B
E1	Familiaridad con el modelo de proyecto utilizado.	1.5	5	7.5
E2	Experiencia en la aplicación.	0.5	4	2
E3	Experiencia en orientación a objetos.	1	4	4
E4	Capacidad del analista líder.	0.5	3	1.5
E5	Motivación.	1	5	5
E6	Estabilidad de los requerimientos	2	4	8
E7	Personal part-time	-1	3	-3

E8	Dificultad del lenguaje de programación	-1	3	-3
EFactor = Sumatoria(A*B)				22
EF = 0.6 + (0.01*TFactor)				0.74

A continuación, se procede con el cálculo de los puntos de casos de uso ajustados mediante la fórmula $UCP = UUCP \times TCF \times EF$, obteniendo un valor de 369.6.

Finalmente se debe hacer el cálculo de Esfuerzo estimado en horas hombre mediante la fórmula $E = UCP \times CF$, donde CF es el esfuerzo en horas que utiliza el personal técnico en resolver un caso de uso. El método de puntos de casos permite tener un valor referencial para este último valor, pero para la estimación de costo de este sistema se considera media jornada laboral ($CF = 4$ horas) para resolver un caso de uso. Tomando en cuenta este último aspecto, el resultado obtenido para E es de 1478 horas/hombre.

Analizando la coyuntura respecto al monto de remuneración que se debe hacer al personal técnico, se tienen dos costos mínimos estimados, uno considera el salario mínimo nacional a la fecha de Bs 2122, y el otro considera el salario de un programador junior en el mercado laboral equivalente a \$USD 500.

Los resultados del costo de software que se presentan en la Tabla 16, muestran que en el caso del salario mínimo nacional, el costo de desarrollo del software es de 13071 Bolivianos, pero considerando un salario en dólares para un programador junior, el costo de desarrollo del software equivale a 3080 dólares.

Tabla 16 Estimación del costo de software

ESFUERZO (en Horas - hombre)	Tiempo equivalente (en meses)	Costo (basado en Salario minimo nacional Bs)	Costo (basado en Salario de un programador junior \$USD 500)
1478	6.1	13071.66 Bs	3080.03 \$USD

CAPITULO 6 CONCLUSIONES Y RECOMENDACIONES

Resumen

En este capítulo se realizan las conclusiones del proyecto de grado, donde se verifica el estado de cumplimiento de los objetivos y los requerimientos del software, además de realizar las recomendaciones respectivas.

6.1. Conclusiones

Los sistemas informáticos son herramientas tecnológicas que permiten a las organizaciones optimizar sus procesos, procedimientos e información, y por ende mejoran la calidad de vida de las personas. En el presente proyecto de grado se desarrolló un sistema informático capaz de mejorar los procesos de emisión de recibos y distribución de insumos médicos en laboratorios clínicos de la Carrera de Odontología de la Universidad Pública de El Alto. Este sistema fue desarrollado con base a métodos y técnicas de ingeniería, permitiendo la entrega de un producto que satisface los requerimientos y expectativas de los interesados.

En cuanto a la calidad del software, se realizó la evaluación con base al estándar ISO/IEC 9126, donde se obtuvo una calificación igual a 89.7 sobre 100, lo cual se considera aceptable, dado que los usuarios finales no se vieron perjudicados durante la operación del sistema durante las pruebas de software, además de que todos los requerimientos establecidos fueron cumplidos. Respecto a la seguridad, se realizó una evaluación con base al Estándar de Verificación de Aplicaciones Web 3.0 de OWASP, y no se encontraron fallas de nivel alto o medio, pero si se encontró fallas de nivel bajo o informativo, los cuales no representan un riesgo para la operación del sistema. Respecto al desempeño y tiempo de respuesta del sistema, también se logró un tiempo de respuesta menor a los establecido en los requerimientos.

En lo concerniente a los objetivos específicos del proyecto, se concluye lo siguiente:

- Se describió el fundamento teórico relacionado a los procesos de emisión de recibos, distribución de insumos médicos y las metodologías o herramientas de software en el Capítulo 2. Estos conceptos permitieron entender mejor los escenarios del negocio y así realizar una mejor abstracción de los requerimientos.
- Se diseñó el sistema informático con base al método UWE, se empleó la notación respectiva para especificar todos los aspectos técnicos del software, garantizando así un diseño documentado y de calidad.
- Se ha desarrollado el sistema con base a métodos y herramientas de ingeniería. Se ha utilizado una arquitectura cliente servidor, y el código se ha organizado siguiendo el patrón MVC. Asimismo, se han utilizado herramientas como HTML5, Javascript, PHP, el gestor de base de datos MySQL y el servidor Web Apache. Por otra parte se ha empleado el método Kanban para gestionar las tareas durante el proceso de desarrollo del software.
- Se han realizado las pruebas técnicas al sistema, empleando casos de prueba que determinan el funcionamiento correcto de los módulos del software. Asimismo, se han efectuado las pruebas respectivas de seguridad con la herramienta Zed Attack Proxy de OWASP, obteniendo los resultados descritos en párrafos anteriores.

6.1. Recomendaciones

6.1.1. Recomendaciones para la institución

- Dar lectura a las guías de usuario con el fin de que las operaciones se efectúen correctamente al trabajar con el sistema.
- Realizar una copia de respaldo de la base de datos cada 2 días o semana, esto con el fin de reducir más aún los riesgos de pérdida de información.
- Respecto a la instalación, procurar el uso de certificados digitales para asegurar una conexión segura a través del protocolo HTTPS.
- Actualizar las contraseñas de usuario al menos cada mes, para evitar intrusiones de terceros al sistema.

6.1.2. Recomendaciones para trabajos futuros

- Desarrollar el módulo de insumos para administrar el almacén central de insumos de clínica odontológica.
- Desarrollar el módulo de seguimiento académico para evaluar los resultados de los estudiantes durante las prácticas en laboratorios clínicos.
- Desarrollar el módulo de gestión documental para organizar los historiales clínicos de los pacientes en forma digitalizada.

REFERENCIAS BIBLIOGRAFICAS

- Actualidad Empresa. (2016). Logística, distribución física y transporte. Revista Electrónica Actualidad Empresa. Recuperado de <http://actualidadempresa.com/logistica-distribucion-fisica-transporte>, [Acceso: Julio de 2020].
- Amaro, K. (2017). Sistema de emisión de comprobantes de pago electrónico en el proceso de facturación de Contasis S.A.C. Universidad Nacional del Centro del Perú, Perú.
- Apache. (2020). Apache HTTP Server Project: Getting Started. Recuperado de <https://httpd.apache.org/docs/2.4/getting-started.html>, [Acceso: Agosto de 2020].
- Arias, R. (2015). Control de inventarios de medicamentos e insumos utilizando la metodología de multicriterio en la bodega de la Unidad Quirúrgica en una IPS de la ciudad de Cali. Universidad Autónoma de Occidente, Santiago de Cali, Colombia.
- Asociación Latinoamericana de Odontopediatría. (2020). Ruta de atención para procedimientos de Odontología Pediátrica durante la etapa de confinamiento o cuarentena de la pandemia COVID-19. Revista de Odontopediatría Latinoamericana, 10(2). Recuperado de <https://www.revistaodontopediatria.org/ediciones/2020/2/art-1/>, [Acceso: Agosto de 2020].
- Caurin, J. (2016). Comprobante de pago. Recuperado de <https://www.economiasimple.net/glosario/comprobante-de-pago>, [Acceso: Julio de 2020].
- Colegio de Odontólogos de La Paz [COLP]. (Mayo, 2020). Protocolo de bio-seguridad para la atención a pacientes y prevención del COVID-19 en clínicas y consultorios odontológicos. Recuperado de https://www.fdiworlddental.org/sites/default/files/media/documents/protocolo_de_bio-seguridad_para_la_atencion_a_pacientes_y_preencion_del_covid-19_en_clinicas_y_consultorios_odontologicos.pdf, [Acceso: Agosto de 2020].
- Diaz, J. & Robiolo, G. (2014). Metodo de estimación de costos de un producto de software Web. 15° Simposio en Ingeniería de Software, Argentina. Recuperado de <http://43jaiio.sadio.org.ar/proceedings/ASSE/02-01-568-2484-1-DR.pdf>, [Acceso: Julio de 2020].
- DIGEMID. (2009). Definición de insumos médicos: Ley de productos farmacéuticos, dispositivos médicos y productos sanitarios. Dirección General de Medicamentos Insumos y Drogas del Perú. Recuperado de <http://www.digemid.minsa.gob.pe/Main.asp?Seccion=760#:~:text=MATERIAL>

%20O%20INSUMO%20M%C3%89DICO%3A%20Sustancia,requieren%20de
%20fuentes%20de%20energ%C3%ADa, [Acceso: Julio de 2020].

Doña, D. (s.f.). Introducción a los sistemas informáticos. Recuperado de <https://www.danieldon.com/informatica%20basica/2%20sistemas%20informaticos.pdf>, [Acceso: Julio de 2020].

Durán, Y. (2012). Administración del inventario: elemento clave para la optimización de las utilidades en las empresas. Revista del Centro de Investigaciones y Desarrollo, 1(1), 55 – 78. Recuperado de <https://www.redalyc.org/pdf/4655/465545892008.pdf>, [Acceso: Julio de 2020].

Escolme. (2012). Documentos Comerciales. Institución Universitaria Escolme, Republica de Colombia. Recuperado de http://www.escolme.edu.co/almacenamiento/oei/tecnicos/doc_comerciales/contenido_u1_4.pdf, [Acceso: Julio de 2020].

ESERP. (2020). Los métodos de control de inventarios mas usados. Fundacion Universitaria ESERP, España. Recuperado de <https://es.eserp.com/articulos/metodo-control-inventarios>, [Acceso: Julio de 2020].

Esmite, I., Farías, M., Farías, N. & Perez, B. (2007). Automatización y Gestión de pruebas funcionales usando Herramientas Open Source. Centro de Ensayos de Software, Universidad de la Republica Montevideo, Uruguay.

Google Developers. (2020). Metrics: Measuring performance and user experience. Recuperado de <https://web.dev/metrics/>, [Acceso: Noviembre de 2020]

Herrera, B. (2011). Diseño e implementación de la factura electrónica como mecanismo para hacer eficiente el proceso de facturación en Certicámara S.A. Universidad de La Salle, Colombia.

HIPODEC. (2018). ¿Qué es un control de inventario?. Universidad Panamericana, Mexico. Recuperado de <https://hipodec.up.edu.mx/blog/que-es-control-inventario#:~:text=El%20control%20de%20inventario%20se,manera%20eficiente%20el%20movimiento%20y>, [Acceso: Agosto de 2020].

IEBS. (2019). ¿Qué es la metodología Kanban y como utilizarla?. Innovation & Entrepreneurship Business School, España. Recuperado de <https://www.iebschool.com/blog/metodologia-kanban-agile-scrum/>, [Acceso: Agosto de 2020].

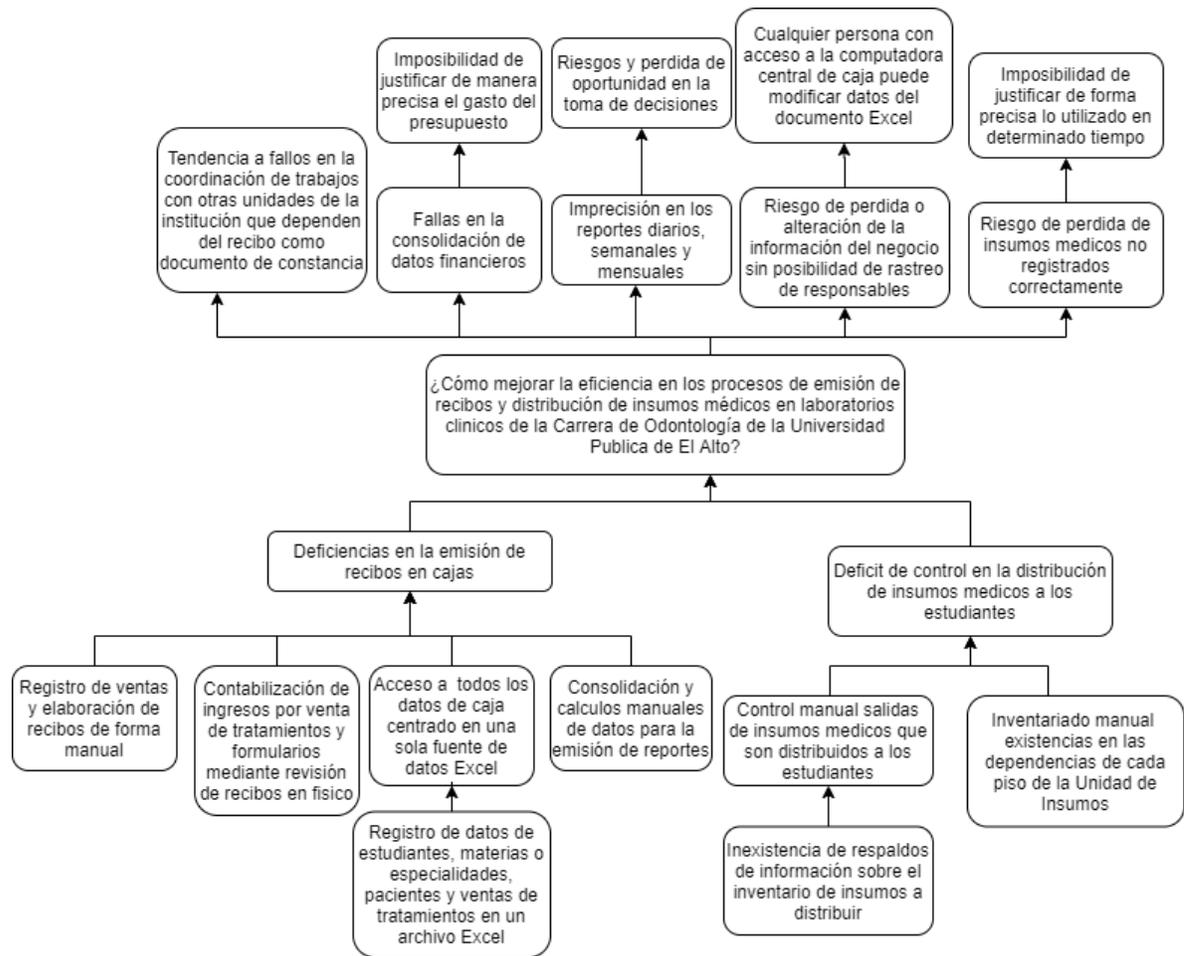
ITCA.(s.f.). Selección de técnicas de ingeniería de software: Factores de calidad del software. Escuela Especializada en Ingeniería. Recuperado de <https://virtual.itca.edu.sv/Mediadores/stis/index.html>, [Acceso: Noviembre de 2020].

- Kanbanize. (2020). ¿Qué es Kanban?: definición, características y ventajas. Recuperado de <https://kanbanize.com/es/recursos-de-kanban/primeros-pasos/que-es-kanban>, [Acceso: Agosto de 2020].
- Karner, G. (1993). Metrics for objectory. University of Linköping, Suecia.
- Lagunas, H. (s.f.). Odontología preventiva. Secretaría de la Defensa Nacional, México. Recuperado de http://www.sedena.gob.mx/pdf/sanidad/odont_preventiva.pdf, [Acceso: Julio de 2020].
- Lapiedra, R., Devece, C. & Guiral, J. (2011). Introducción a la gestión de sistemas de información en la empresa. Universitat Jaume I, España. Recuperado de <https://libros.metabiblioteca.org/bitstream/001/193/8/978-84-693-9894-4.pdf>, [Acceso: Julio de 2020].
- Lencina, A., Medina, Y. & Dapozo, G. (2016). Aplicación para estimar costos en proyectos de software. 17° Simposio en Ingeniería de Software, Argentina. Recuperado de <https://core.ac.uk/download/pdf/301073268.pdf>, [Acceso: Julio de 2020].
- Meucii, M. & Muller, A. (2016). OWASP Testing Guide 4.0. Open Web Application Security Project.
- Monrroy, J. (2014). Sistema Web para el control y administración de recursos humanos CASO: Empresa de Limpieza Industrial TOTES LTDA. Universidad Mayor de San Andrés.
- Nieves, A., Ucán, J. & Menendez, V. (2014). UWE en Sistema de Recomendación de Objetos de Aprendizaje. Aplicando Ingeniería Web: Un método en Caso de Estudio. Revista Latinoamericana de Ingeniería de Software, 2(3), 137 – 143.
- Oracle. (2020). Oracle datasheet: MySQL Enterprise Edition. Recuperado de <https://www.mysql.com/products/enterprise/mysql-datasheet.en.pdf>, [Acceso: Agosto de 2020].
- PHP Group. (2020). ¿Qué es PHP?. Recuperado de <https://www.php.net/manual/es/intro-what-is.php>, [Acceso: Agosto de 2020].
- Rock, R. (2019). ¿Qué es un lenguaje de programación y que tipos existen?. Recuperado de <https://rockcontent.com/es/blog/que-es-un-lenguaje-de-programacion>, [Acceso: Agosto de 2020].
- Rodriguez, A. (2009). Metodologías de diseño usadas en Ingeniería Web, su vinculación con las TIC. Universidad Nacional de La Plata, Argentina.
- Strenamedical. (2010). Insumos médicos, que son y cuales son sus tipos. Recuperado de <https://strenamedical.com.ar/insumos-medicos-que-son-y-cuales-son-sus-tipos>, [Acceso: Julio de 2020].

- Tito, A. (2015). Sistema Web integrado de control, pago e historial de socios basado en biométrico dactilar Caso: Empresa CIMCO LTDA. Universidad Mayor de San Andrés.
- Universidad Pública de El Alto [UPEA]. (s.f.). Clínica Odontológica. Recuperado de https://www.upea.bo/unidad_academica/centro/ver/2, [Acceso: Junio de 2020].
- Valero, Y. (2017). La odontología en nuestros días. Revista odontológica mexicana, 21(3), 154. Recuperado de http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S1870-199X2017000300154&lng=es&tlng=es, [Acceso: Julio de 2020].
- Villca, E. (2018). Aplicación móvil de control de ventas e inventarios con alertas tempranas: caso empresa importadora y distribuidora de alimentos e insumos para mascotas San Gabriel PET. Carrera de Informática, Universidad Mayor de San Andrés.

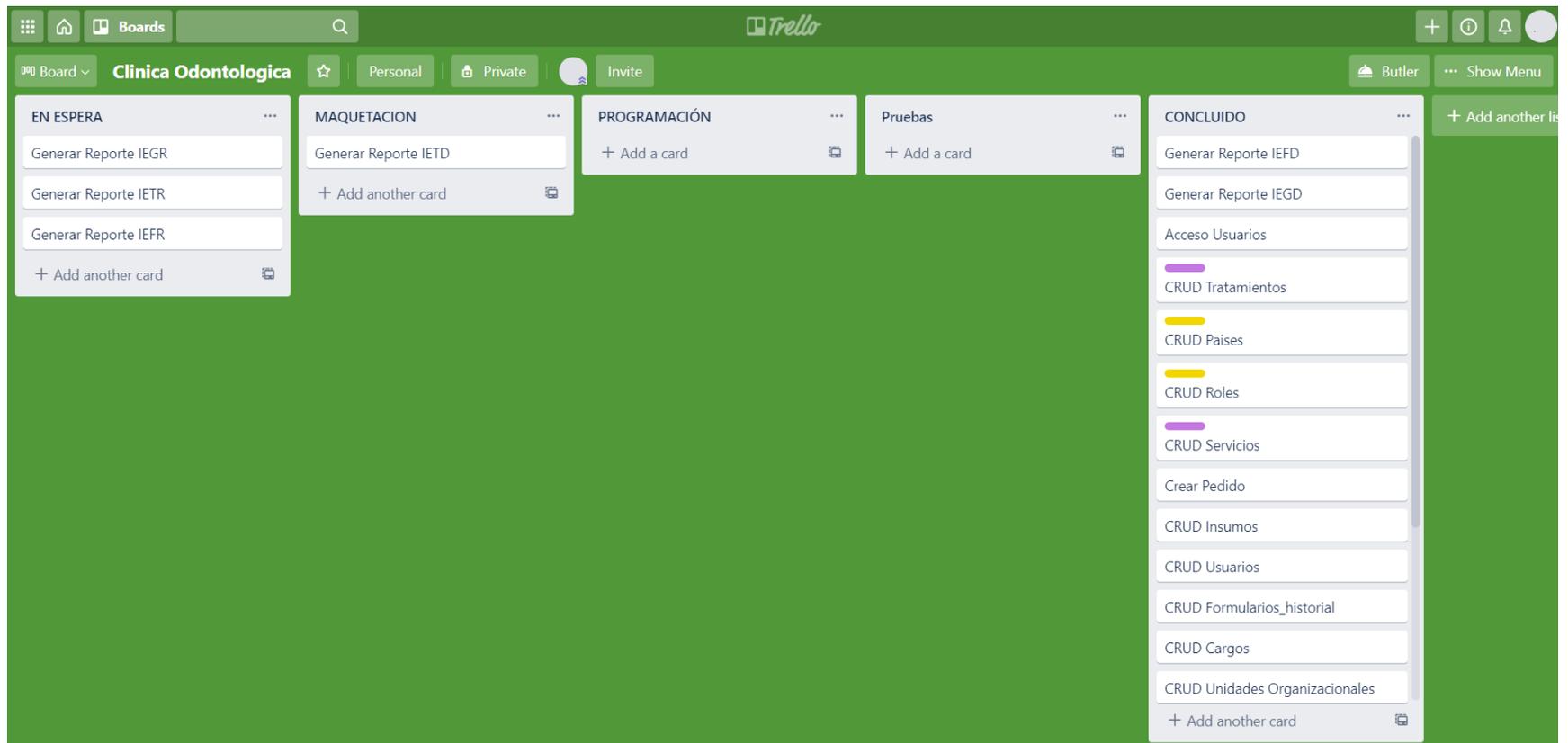
ANEXOS

ANEXO A ÁRBOL DE PROBLEMAS



ANEXO B TABLERO KANBAN

Se presenta el tablero Kanban utilizado durante el desarrollo de proyecto. La ilustración corresponde al momento antes de culminar los reportes del sistema.



ANEXO C DICCIONARIO DE DATOS

NOMBRE DE LA TABLA	DESCRIPCION TABLA	NOMBRE CAMPO	TIPO DE DATO	LONGITUD	RESTRICCION	DESCRIPCION DEL CAMPO
unidad_organizacional	Almacena los datos correspondientes a las unidades organizacionales	uor_id	integer	10	PK	Clave primaria de la tabla
		uor_superior	integer	10	FK	Clave foranea, hace referencia a la unidad organizacional superior de la cual depende la actual unidad organizacional
		uor_nombre	varchar	100	UNIQUE	Nombre de la unidad organizacional
cargo	Almacena los datos correspondientes a los cargos de una unidad organizacional	car_id	integer	10	PK	Clave primaria de la tabla
		uor_id	integer	10	FK	Clave foranea, hace referencia a la unidad organizacional donde el cargo pertenece
		car_nombre	varchar	100	UNIQUE	Nombre del cargo
persona	Almacena los datos generales de una persona	per_id	integer	10	PK	Clave primaria de la tabla
		per_ci	varchar	100	UNIQUE	Número de identificación de la cédula de identidad
		per_ci_expedido	varchar	100	-	Departamento donde se ha expedido la cédula de identidad, los valores son cadenas cuyos valores identifican a los 9 departamentos de Bolivia
		per_nombres	varchar	100	-	Nombres de la persona
		per_primer_apellido	varchar	100	-	Primer apellido de la persona
		per_segundo_apellido	varchar	100	-	Segundo apellido de la persona

		per_genero	integer	10	-	Genero o sexo con el cual la persona se identifica
docente	Almacena los datos correspondientes a un docente. Hereda datos de la tabla persona	doc_id	integer	10	PK	Clave primaria de la tabla
		per_id	integer	10	FK	Clave foranea, hace referencia a los datos de una tupla en la tabla persona
estudiante	Almacena los datos correspondientes a un estudiante. Hereda datos de la tabla persona	est_id	integer	10	PK	Clave primaria de la tabla
		per_id	integer	10	FK	Clave foranea, hace referencia a los datos de una tupla en la tabla persona
		est_registro_universitario	varchar	100	UNIQUE	Identificador del estudiante en la UPEA
paciente	Almacena los datos correspondientes a un paciente. Hereda datos de la tabla persona	pac_id	integer	10	PK	Clave primaria de la tabla
		per_id	integer	10	FK	Clave foranea, hace referencia a los datos de una tupla en la tabla persona
autorizacion	Almacena los datos correspondientes a las autorizaciones que realizan los docentes	aut_id	integer	10	PK	Clave primaria de la tabla
		doc_id	integer	10	FK	Clave foranea, hace referencia al docente que autorizó la compra de un producto o servicio para la emisión del recibo correspondiente
		aut_descripcion	varchar	100	-	Descripcion de la autorizacion
		aut_fecha	date	10	-	Fecha en la cual se efectuó la autorización
recibo		rec_id	integer	10	PK	Clave primaria de la tabla

	Almacena los datos correspondientes a los recibos emitidos por caja	est_id	integer	10	FK	Clave foranea, hace referencia al estudiante que realizó la compra de un producto o servicio
		pac_id	integer	10	FK	Clave foranea, hace referencia al paciente para el cual se realizó la compra de un producto o servicio
		aut_id	integer	10	FK	Clave foranea, hace referencia a la autorización realizada por el docente para la compra del producto o servicio
		rec_total_recibo	float	23	-	Monto total del recibo, corresponde a la sumatoria de los subtotales por item del pedido
		rec_fecha_recibo	date	10	-	Fecha en la cual se realizó el recibo
		rec_observacion	varchar	200	-	Observacion del recibo (si corresponde)
detalle_pedido	Almacena los datos correspondientes al detalle del pedido.	dpe_id	integer	10	PK	Clave primaria de la tabla
		rec_id	integer	10	FK	Clave foranea, hace referencia al recibo al cual corresponde el item de pedido
		pse_id	integer	10	FK	Clave foranea, hace referencia al producto o servicio adquirido
		dpe_cantidad	integer	10	-	Cantidad del producto o servicio adquirido
		dpe_subtotal	float	23	-	Subtotal correspondiente al producto o servicio adquirido

detalle_notificacion	Almacena los datos correspondientes al detalle de las notificaciones realizadas a los usuarios	dno_id	integer	10	PK	Clave primaria de la tabla
		usu_id	integer	10	FK	Clave foranea, hace referencia al usuario que debe recibir la notificacion de compra de producto o servicio
		dpe_id	integer	10	FK	Clave foranea, hace referencia al detalle del producto o servicio en el pedido
		dno_mensaje	varchar	200	-	Mensaje de la notificacion
		dno_fecha_notificacion	date	10	-	Fecha en la cual se realizó la notificación
		dno_recibido	boolean	2	-	Valor booleano, identifica si una notificacion fue recibida o no por el usuario final
usuario	Almacena los datos correspondientes a las cuentas de usuario	usu_id	integer	10	PK	Clave primaria de la tabla
		per_id	integer	10	FK	Clave foranea, hace referencia a la persona a la cual pertenece la cuenta de usuario
		usu_nombre	varchar	100	UNIQUE	Nombre de la cuenta de usuario
		usu_password	varchar	100	UNIQUE	Clave de la cuenta de usuario
		rol_id	integer	10	FK	Clave foranea, hace referencia al rol que tiene el usuario en el sistema
		ser_id	integer	10	-	Relacion debil, hace referencia a si un usuario pertenece a un servicio de clinica odontologica o no

		car_id	integer	10	FK	Clave foranea, hace referencia al cargo que tiene el usuario en la institución
rol	Almacena los datos correspondientes a los roles del sistema	rol_id	integer	10	PK	Clave primaria de la tabla
		rol_nombre	varchar	100	UNIQUE	Nombre del rol en el sistema
		rol_descripcion	varchar	200	-	Descripcion del rol
producto_servicio	Almacena los datos correspondientes a los productos o servicios del negocio	pse_id	integer	10	PK	Clave primaria de la tabla
		pse_nombre	varchar	100	UNIQUE	Nombre del producto o servicio
		pse_descripcion	varchar	200	-	Descripcion del producto o servicio
		pse_precio_unitario	float	23	-	Precio unitario del producto o servicio
tratamiento	Almacena datos correspondientes a tratamientos. Hereda datos de la tabla producto_servicio	tra_id	integer	10	PK	Clave primaria de la tabla
		pse_id	integer	10	FK	Clave foranea, hace referencia a una tupla de la tabla producto servicio (objeto padre)
		tra_stock	float	23	-	Cantidad disponible del tratamiento
		mat_id	integer	10	FK	Clave foranea, hace referencia a la materia o especialidad a la cual pertenece el tratamiento
formulario_historial	Almacena datos correspondientes a formularios de historial del negocio. Hereda	for_id	integer	10	PK	Clave primaria de la tabla
		pse_id	integer	10	FK	Clave foranea, hace referencia a una tupla de la tabla producto servicio (objeto padre)

	datos de la tabla producto_servicio	for_stock	float	23	-	Cantidad disponible de formularios
		mat_id	integer	10	FK	Clave foranea, hace referencia a la materia o especialidad a la cual pertenece el formulario
servicio	Almacena datos correspondientes a los servicios ofrecidos en la clinica odontologica. Hereda datos de la tabla producto_servicio	ser_id	integer	10	PK	Clave primaria de la tabla
		ser_nombre	varchar	100	-	Nombre del servicio
		ser_descripcion	varchar	200	-	Descripcion del servicio
pais	Almacena datos correspondientes a los paises	pai_id	integer	10	PK	Clave primaria de la tabla
		pai_nombre	varchar	100	UNIQUE	Nombre del país
marca	Almacena datos correspondientes a las marcas de productos o insumos	mar_id	integer	10	PK	Clave primaria de la tabla
		mar_nombre	varchar	100	UNIQUE	Nombre de la marca de producto o insumo médico
		pai_id	integer	10	FK	Clave foranea, hace referencia al pais de donde proviene la marca
proveedor	Almacena datos correspondientes a los proveedores de insumos o productos	pdo_id	integer	10	PK	Clave primaria de la tabla
		pdo_nombre	varchar	100	UNIQUE	Nombre del proveedor de productos o insumos medicos
		pdo_descripcion	varchar	200	-	Descripcion del proveedor
detalle_ingreso_insumo	Almacena los datos correspondientes	din_id	integer	10	PK	Clave primaria de la tabla
		ins_id	integer	10	FK	Clave foranea, hace refencia a un insumo medico

	a los ingresos al inventario de insumos	din_fecha_ingreso	date	10	-	Fecha del ingreso de un insumo al inventario
		din_cantidad	float	23	-	Cantidad ingresada al inventario
unidad_medida	Almacena los datos de las unidades de medida utilizadas en el sistema	ume_id	integer	10	PK	Clave primaria de la tabla
		ume_nombre	varchar	100	UNIQUE	Nombre de la unidad de medida
		ume_abreviacion	varchar	100	UNIQUE	Abreviación de la unidad de medida
insumo	Almacena los datos de los insumos en inventario	ind_id	integer	10	PK	Clave primaria de la tabla
		ume_id	integer	10	FK	Clave foranea, hace referencia a la unidad de medida que utiliza el producto o insumo en inventario
		mar_id	integer	10	FK	Clave foranea, hace referencia a la marca del producto
		pdo_id	integer	10	FK	Clave foranea, hace referencia al proveedor del producto
		ins_nombre	varchar	100	UNIQUE	Nombre del producto
		ins_descripcion	varchar	200	-	Descripcion del producto
		ins_precio_unitario	float	23	-	Precio unitario del producto
		ins_stock	float	23	-	Cantidad disponible del producto
detalle_salida_insumo	Almacena los datos correspondientes a las salidas del inventario de insumos	dsi_id	integer	10	PK	Clave primaria de la tabla
		rec_id	integer	10	FK	Clave foranea, hace referencia al recibo al cual corresponde la salida de insumo
		ins_id	integer	10	FK	Clave foranea, hace referencia al insumo que fue despachado para conformar un tratamiento

		dsi_fecha_salida	date	10	-	Fecha de la salida de inventario
		dsi_cantidad	float	23	-	Cantidad del producto saliente que corresponde a un recibo por tratamiento
materia_especialidad	Almacena los datos correspondientes a la materias o especialidades de clinica odontológica	mat_id	integer	10	PK	Clave primaria de la tabla
		mat_nombre	varchar	100	UNIQUE	Nombre de la materia o especialidad
		mat_semestre	integer	10	-	Número de semestre o año al cual pertenece la especialidad

ANEXO D EVALUACION DE CALIDAD DEL SISTEMA

Se evaluará la calidad del sistema de la calidad del software con base al modelo de calidad ISO – 9126.

1. Objetivo de la evaluación

Determinar el nivel de calidad del producto software en las dimensiones de funcionalidad, confiabilidad, mantenibilidad y usabilidad.

2. Tipo de producto a evaluar

Producto Final

3. Modelo de calidad aplicado

Modelo de calidad para la calidad externa ISO/IEC 9126-1. Se presenta a continuación una representación del modelo de calidad adoptado.



Fuente: (ITCA, s.f.).

4. Procedimiento de evaluación

4.1. Evaluación de funcionalidad

La funcionalidad no se puede medir directamente porque corresponde derivar medidas directas, como el punto función que cuantifica el tamaño y la complejidad del sistema en términos de las funciones del usuario, determinando las características del dominio, como ser: número de entrada usuario, número salida usuario, número petición usuario, número de archivos, número interfaces externas (Pressman, 2010).

Para realizar el cálculo de la cuenta total de puntos, se consideran los factores medios de ponderación, como se muestra en la siguiente tabla.

Cálculo de puntos función con factor medio sin ajustar.

	Parámetro de Medición	Cuenta	Factor medio	Total
1	Número de Entrada del Usuario	41	4	164
2	Número de Salidas del Usuario	39	5	195
3	Número de Petición del Usuario	18	4	72
4	Número de archivos	17	10	170
5	Número de interfaces externas	1	7	7
CT	Cuenta Total			608

Fuente: Elaborado con base a (Pressman, 2010).

Para calcular los valores de ajuste de complejidad, se deben aplicar las siguientes preguntas.

	FACTOR	0	1	2	3	4	5	Fi
1	¿Requiere el sistema copias de seguridad y recuperación?						X	5
2	¿Se requiere comunicación de datos?						X	5
3	¿Existen funciones de procesos distribuido?				X			3
4	¿Es crítico el rendimiento?				X			3
5	¿Se ejecutaría el sistema en un entorno operativo existente y fuertemente utilizado?				X			3
6	¿Requiere el sistema entrada de datos interactiva?					X		4
7	¿Requiere la entrada de datos interactiva que las transacciones de entrada se lleven a cabo sobre múltiples pantallas o variadas operaciones?				X			3
8	¿Se actualizan los archivos maestros de forma interactiva?						X	5
9	¿Son complejas las entradas, salidas, archivos o peticiones?				X			3
10	¿Es complejo el procesamiento interno?				X			3
11	¿Se diseña el código para ser reutilizable?						X	5
12	¿Están incluidas en el diseño la conversión y la instalación?			X				2
13	¿Se ha diseñado el sistema para soportar múltiples instalaciones?				X			3
14	¿Se ha diseñado para facilitar cambios y ser fácilmente utilizada por el usuario?					X		4

FACTOR DE COMPLEJIDAD (FCT)	51
------------------------------------	-----------

Fuente: Elaborado con base a (ITCA,s.f).

Para calcular Punto Función (PF), usamos la siguiente relación:

$$PF = cuenta\ total * (0.65 + 0.01 * \sum Fi)$$

Calculando el punto función real:

$$PF = 608 * (0.65 + 0.01 * 51)$$

$$PF = 705.28$$

Al 100% el nivel de confianza consideramos la sumatoria de $F_i = 70$ como valor máximo de ajuste de complejidad sería:

$$PF_{max} = 608 * (0.65 + 0.01 * 70)$$

$$PF_{max} = 820.8$$

Por tanto, el grado de funcionalidad está dado por:

$$Funcionalidad = 705/820 = 0.859$$

La funcionalidad del sistema es de 85.9% sin riesgo a fallo y operatividad y 14.1% de colapso del sistema.

4.2. Evaluación de confiabilidad

Para determinar la confiabilidad de un software especificamos desde el instante que empieza a funcionar es decir $t_0 = 0$, a partir de este momento se realiza las observaciones pertinentes. P es la probabilidad de encontrar una falla en el sistema considerando el tiempo de falla, y como t_1 como intervalo entre ambos tiempos es una variable continua de función continua, que nos da la confiabilidad en términos probabilísticos, mediante.

$$P(T \leq t) = F(t) \quad \text{Probabilidad de fallos}$$

$$P(T > t) = 1 - F(t) \quad \text{Probabilidad de éxito}$$

Para el cálculo de las probabilidades se tomó la distribución exponencial, por la existencia de intervalos continuos.

$$F(t) = 1 - e^{-\lambda * t}$$

Para calcular el índice de error se considera que de 60 ejecuciones realizadas con el sistema 2 fueron fallidas en un periodo de una semana (t=6), y reemplazando tenemos:

$$F(t) = 1 - e^{-[0.03*6]}$$

$$F(t) = 0,164$$

Reemplazando en las fórmulas de probabilidades:

$$P(T \leq t) = F(t)$$

$$P(T \leq t) = 0,164$$

Probabilidad de fallos

$$P(T > t) = 1 - F(t)$$

$$P(T > t) = 1 - 0,164$$

$$P(T > t) = 0,836$$

Probabilidad de éxito

Siendo la probabilidad de fallo del 16% y la probabilidad de éxito de un 84%.

4.3. Evaluación de mantenibilidad

Se desarrolla para mejorar el sistema a los nuevos requerimientos que tengan, el estándar IEEE 982.1-1988 sugiere el índice de madurez del software como métrica específica de mantenimiento. Proporciona un indicador en la estabilidad de un producto software mediante el cálculo de:

$$M = \frac{(Mt - (Fa + Fc + Fd))}{Mt}$$

Mt = Número de módulos actual

Fa = Número de módulos que se añadió

Fc = Número de módulos actual que se cambió

Fd = Número de módulos anterior que se borró en lo actual

Por tanto, se calcula la M (madurez), considerando modificaciones en 2 módulos del sistema, debido a requerimientos finales del usuario.

$$M = (21 - (0 + 2 + 0)) / 21$$

$$M = 0.952$$

Se puede concluir que el software tiene una estabilidad de 95%, fácil mantenimiento y el 5% restante sería el margen de error por los cambios y modificaciones.

4.4. Evaluación de la usabilidad

El grado de usabilidad muestra la facilidad de uso del producto software. Para evaluar el grado, se aplica el siguiente cuestionario.

Preguntas de usabilidad

#	Preguntas
1	¿Puede utilizar con facilidad el sistema?
2	¿Puede controlar operaciones que el sistema solicita?
3	¿Las respuestas del sistema son complicadas?
4	¿El sistema permitió la retroalimentación de información?
5	¿El sistema cuenta con interfaces agradables a la vista?
6	¿La respuesta del sistema es satisfactorio?
7	¿Los resultados que proporciona el sistema facilitan el trabajo?

Fuente: Elaborado con base a (Villca, 2018)

Cada pregunta debe tener respuesta afirmativa o negativa. El cuestionario de fue aplicado mediante Google Forms con el fin de que las persona que evalúan el software lo hagan vía Internet, esto para no perjudicar sus actividades, además como prevención ante la situación actual de la pandemia por el COVID-19. La columna R1 corresponde a las respuestas del encargado de insumos, la columna R2 corresponde a las respuestas del encargado de caja, y R3 corresponde a las respuestas del técnico informático.

Para el cálculo de la usabilidad se debe considerar la siguiente fórmula:

$$U = ((\text{Sum } (x_i)/N)*100)$$

Donde x_i resulta de comparar la cantidad de respuestas afirmativas de acuerdo a cada pregunta, y N es la cantidad de respuestas recibidas.

#	Preguntas	R1		R2		R3		x_i
		SI	NO	SI	NO	SI	NO	
1	¿Puede utilizar con facilidad el sistema?	X		X		X		1
2	¿Puede controlar operaciones que el sistema solicita?	X			X	X		0.6
3	¿Las respuestas del sistema son complicadas?		X		X		X	1
4	¿El sistema permitió la retroalimentación de información?	X		X			X	1
5	¿El sistema cuenta con interfaces agradables a la vista?	X		X		X		1
6	¿La respuesta del sistema es satisfactorio?	X		X		X		1
7	¿Los resultados que proporciona el sistema facilitan el trabajo?	X		X		X		1
USABILIDAD								0.94

Aplicando la fórmula, se obtiene una usabilidad del 94% de acuerdo a los datos.

ANEXO E MAPA DE CONTROL DE ACCESO BASADO EN ROLES

Para realizar la asignación de funcionalidades, la tabla de control de acceso basado en roles fue elaborada con base a los casos de uso identificados durante el análisis de requerimientos del sistema. Se marca con 1 los accesos otorgados y con vacío los accesos no permitidos.

FUNCIONALIDADES	ROLES			
	ADMINISTRADOR DE SISTEMA	ENCARGADO DE CAJA	ENCARGADO DE INSUMOS	ENCARGADO DE SERVICIO
GESTION DE USUARIO				
Listar cuentas de usuario	1			
Registrar cuenta de usuario	1			
Actualizar cuenta de usuario	1			
Dar baja cuenta de usuario	1			
Acceder al sistema	1	1	1	1
GESTIONAR UNIDADES ORGANIZACIONALES				
Listar unidades organizacionales	1			
Registrar unidad organizacional	1			
Actualizar Unidad organizacional	1			
Dar baja unidad organizacional	1			
GESTIONAR CARGOS				
Listar cargos	1			
Registrar cargo	1			
Actualizar cargo	1			
Dar baja cargo	1			
GESTIONAR ROLES				
Listar roles	1			
Actualizar rol	1			
GESTIONAR PAISES				
Listar paises	1			
Registrar pais	1			
Actualizar pais	1			
Dar baja pais	1			

GESTIONAR PROVEEDOR				
Listar proveedores	1		1	
Registrar proveedor	1		1	
Actualizar proveedor	1		1	
Dar baja proveedor	1			
GESTIONAR MARCA				
Listar marcas	1		1	
Registrar marca	1		1	
Actualizar marca	1		1	
Dar baja marca	1			
GESTIONAR UNIDADES DE MEDIDA				
Listar unidades de medida	1		1	
Registrar unidades de medida	1		1	
Actualizar unidad de medida	1		1	
Dar baja unidad de medida	1			
GESTIONAR NOTIFICACIONES				
Listar notificaciones	1			1
Actualizar configuracion notificaciones	1			
GESTIONAR SERVICIOS				
Listar servicios	1			
Registrar servicio	1			
Actualizar servicio	1			
Dar baja servicio	1			
GESTIONAR FORMULARIOS DE HISTORIAL				
Listar formularios	1			
Registrar formulario	1			
Actualizar formulario	1			
Dar baja formulario	1			
GESTIONAR TRATAMIENTOS				
Listar tratamiento	1			
Registrar tratamiento	1			
Actualizar tratamiento	1			
Dar baja tratamiento	1			

GESTIONAR MATERIAS - ESPECIALIDADES				
Listar materias - especialidades	1			
Registrar materia - especialidad	1			
Actualizar materia - especialidad	1			
Dar baja materia - especialidad	1			
GESTIONAR INSUMOS				
Listar insumos			1	
Registrar insumo			1	
Actualizar insumo			1	
Dar baja insumo			1	
GESTIONAR INGRESO DE INSUMO				
Listar ingresos de insumo			1	
Registrar ingreso de insumo			1	
Actualizar ingreso de insumo			1	
Dar baja ingreso de insumo			1	
GESTIONAR SALIDA DE INSUMO				
Listar salida de insumo			1	
Registrar salida de insumo			1	
Actualizar salida de insumo			1	
Dar baja salida de insumo			1	
GESTIONAR PEDIDOS Y RECIBOS				
Listar pedidos - recibos		1		
Registrar recibos		1		
Dar baja recibo		1		
GESTIONAR PACIENTES				
Listar pacientes		1		
Actualizar paciente		1		
Dar baja paciente		1		
GESTIONAR DOCENTES				
Listar docentes		1		

Registrar docente		1		
Actualizar docente		1		
Dar baja docente		1		
GESTIONAR ESTUDIANTES				
Listar estudiantes		1		
Registrar estudiantes		1		
Actualizar estudiantes		1		
Dar baja estudiantes		1		

ANEXO F REPORTE DE EVALUACION DE SEGURIDAD

El sistema fue evaluado por la herramienta de pruebas de seguridad Zed Attack Proxy versión 2.9, el cual realiza pruebas automáticas de seguridad a todas las rutas disponibles de la aplicación. Luego de haber realizado el análisis, la herramienta emite un reporte considerando alertas de nivel: informativo, bajo, medio y alto. A continuación se presenta el reporte de la herramienta mencionada.

ZAP Scanning Report

Summary of Alerts

Risk Level	Number of Alerts
High	0
Medium	0
Low	6
Informational	3

Alert Detail

Low (Medium)	Cookie Without SameSite Attribute
Description	A cookie has been set without the SameSite attribute, which means that the cookie can be sent as a result of a 'cross-site' request. The SameSite attribute is an effective counter measure to cross-site request forgery, cross-site script inclusion, and timing attacks.
URL	http://localhost/odo
Method	GET
Parameter	ci_session
Evidence	Set-Cookie: ci_session
Instances	3

Solution	Ensure that the SameSite attribute is set to either 'lax' or ideally 'strict' for all cookies.
Reference	https://tools.ietf.org/html/draft-ietf-httpbis-cookie-same-site
CWE Id	16
WASC Id	13
Source ID	3
Low (Medium)	No se encuentra encabezado X-Content-Type-Options Header
Description	El encabezado Anti-MIME-Sniffing X-Content-Type-Options no estaba configurado para 'nosniff'. Esto permite versiones antiguas de Internet Explorers y Chrome ejecutar MIME-sniffing en el cuerpo de la respuesta, causando potencialmente que el cuerpo de respuesta sea interpretado y desarrollado como un tipo de contenido diferente que el tipo de contenido declarado. Estos (principios de 2014) y versiones antiguas de Firefox preferiblemente usarán el tipo de contenido declarado (si hay uno establecido), antes que ejecutar el MIME-Sniffing.
URL	http://localhost/odo/css/
Method	GET
Parameter	X-Content-Type-Options
URL	http://localhost/odo/usuarios/login
Method	GET
Parameter	X-Content-Type-Options
URL	http://localhost/odo/js/
Method	GET
Parameter	X-Content-Type-Options
Instances	15

Solution	<p>Asegúrese que el servidor de la aplicación/web establezca el encabezado Content-Type apropiadamente, y que esté establecido el encabezado X-Content-Type-Options en 'nosniff' para todas las páginas web.</p> <p>Si es posible, asegúrese que el último usuario usa un navegador web compatible con los estándares y moderno que no ejecute MIME-sniffing en absoluto, o que pueda ser dirigida por el servidor de la aplicación/web para no ejecutar MIME-sniffing.</p>
Other information	Este inconveniente aún aplica para páginas de error (401, 403, 500, etc) ya que esas páginas a menudo todavía están afectadas por problemas de inyección, en cuyos casos aún hay preocupación de buscadores rastreando páginas fuera de su tipo de contenido verídico. En límite 'alto' este escáner no alertará sobre las respuestas de error al cliente o servidor.
Reference	http://msdn.Microsoft.com/en-us/library/le/gg622941%28v=vs.85%29.aspx https://www.owasp.org/index.php/List_of_useful_HTTP_headers
CWE Id	16
WASC Id	15
Source ID	3
Low (Medium)	Absence of Anti-CSRF Tokens
Description	<p>No Anti-CSRF tokens were found in a HTML submission form.</p> <p>Una solicitud falsa entre sitios en un ataque que compromete y obliga a una víctima a enviar su solicitud HTTP a un destino objetivo sin su conocimiento o intención para poder realizar una acción como víctima. La causa oculta es la funcionalidad de la aplicación utilizando acciones de URL/formulario que pueden ser adivinados de forma repetible. La naturaleza del ataque es que CSRG explota la confianza que un sitio web proporciona a un usuario. Por el contrario, las cadenas de comandos de los sitios cruzados (XSS) explotan la confianza que un usuario proporciona en un sitio web. Al igual que XSS, los ataques CSRG no son de forma necesaria de sitios cruzados, pero hay la posibilidad de que si pueden serlo. La falsificación de las solicitudes ente los sitios también se conoce como CSRF, XSRG, ataques con un solo clic, montaje de sesión, diputado confundido y navegación en alta mar.</p> <p>Los ataques de CSRG son muy efectivos en varias situaciones, que incluyen:</p> <ul style="list-style-type: none"> *La víctima tiene una sesión activa en el sitio de destino. *La víctima se autoriza por medio de la autenticación HTTP en el sitio de destino.

	<p>*La víctima se encuentra en la misma red local que el sitio de destino.</p> <p>CSRF se ha utilizado especialmente para poder realizar una acción contra un sitio objetivo utilizando los privilegios de la víctima, pero se han revelado técnicas recientes para difundir información al obtener el acceso a la respuesta. El riesgo de divulgación de información aumenta de forma drástica cuando el sitio de destino se encuentra vulnerable a XSS, porque XSS se puede utilizar como una plataforma para CSRF, lo que le permite al atacante que opere desde adentro de los límites de la misma política de origen.</p>
URL	http://localhost/odo
Method	GET
Evidence	<code><form action="http://localhost/odo/usuarios/validar_cuenta" id="form-usuario" method="post" accept-charset="utf-8"></code>
Evidence	<code><form action="http://localhost/odo/usuarios/validar_cuenta" id="form-usuario" method="post" accept-charset="utf-8"></code>
Instances	4
Solution	<p>Frase: Arquitectura y Diseño</p> <p>Utilice una biblioteca o marco comprobado que no acepte que ocurra esta debilidad o que proporcione construcciones que permitan que esta debilidad sea más sencilla de evitar.</p> <p>Por ejemplo, utilice el paquete anti-CSRF como el CSRFGuard de OWASP.</p> <p>Fase: Implementación</p> <p>Asegúrese de que su aplicación esté libre de fallas de secuencias de comandos entre sitios, ya que la mayoría de las defensas de CSRF pueden detenerse por alto por medio del uso de secuencias de comandos manejadas por el atacante.</p> <p>Fase: Arquitectura y Diseño</p> <p>Origina un nonce único para cada uno de los formularios, coloque el nonce en el formulario y confirme la independencia al obtener el formulario. Asegúrese de que el nonce no sea predecible (CWE-330).</p> <p>Usted tiene que tener en cuenta que esto puede pasar desapercibido utilizando XSS.</p>

	<p>Identificar las operaciones que sean especialmente peligrosas. Cuando el usuario desarrolla una operación peligrosa, envíe una solicitud de confirmación de forma separada para poder garantizar que el usuario tenga la intención de desarrollar esa operación.</p> <p>Usted tiene que tener en cuenta que esto puede pasar desapercibido utilizando XSS.</p> <p>Utilice el control de gestión de la sesión de ESAPI.</p> <p>Este control introduce un elemento para CSRF.</p> <p>No utilice el método GET para ninguna de las solicitudes que puedan desencadenar un cambio de estado.</p> <p>Fase: Implementación</p> <p>Revise que la solicitud se creó en la página esperada. Esto podría quebrar la funcionalidad auténtica, ya que los usuarios o los representantes puede ser que hayan desactivado el envío de Referer por motivos de privacidad.</p>
Other information	No known Anti-CSRF token [anticsrf, CSRFToken, __RequestVerificationToken, csrfmiddlewaretoken, authenticity_token, OWASP_CSRFTOKEN, anoncsrf, csrf_token, _csrf, _csrfSecret] was found in the following HTML form: [Form 1: "hidden" "usr" "pwd"].
Reference	http://projects.webappsec.org/Cross-Site-Request-Forgery http://cwe.mitre.org/data/definitions/352.html
CWE Id	352
WASC Id	9
Source ID	3
Low (Medium)	Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)
Description	The web/application server is leaking information via one or more "X-Powered-By" HTTP response headers. Access to such information may facilitate attackers identifying other frameworks/components your web application is reliant upon and the vulnerabilities such components may be subject to.
URL	http://localhost/odo/usuarios/validar_cuenta
Method	POST

Evidence	X-Powered-By: PHP/7.4.7
Instances	8
Solution	Ensure that your web server, application server, load balancer, etc. is configured to suppress "X-Powered-By" headers.
Reference	http://blogs.msdn.com/b/varunm/archive/2013/04/23/remove-unwanted-http-response-headers.aspx http://www.troyhunt.com/2012/02/shhh-dont-let-your-response-headers.html
CWE Id	200
WASC Id	13
Source ID	3
Low (Medium)	Inclusión de archivos de origen JavaScript Cross-Domain
Description	Las páginas incluyen uno o mas archivos encriptados de un dominio de terceros.
URL	http://localhost/odo/
Method	GET
Parameter	https://www.google.com/recaptcha/api.js
Evidence	<script src="https://www.google.com/recaptcha/api.js" async defer></script>
Instances	4
Solution	Asegúrese que los archivos de la fuente JavaScript están descargados solo de sus fuentes confiables, y las fuentes no pueden ser controladas por los usuarios finales de la aplicación.
Reference	
CWE Id	829
WASC Id	15

Source ID	3
Low (Medium)	Divulgación de error de aplicación
Description	Esta página contiene un mensaje de error/advertencia que podría revelar información sensible como la ubicación del archivo que produjo la excepción no controlada. Esta información puede ser usada para lanzar futuros ataques contra la aplicación web. La alerta podría ser una falsa positiva si el mensaje de error es encontrado dentro de una página de documentación.
URL	http://localhost/odo/img/favicon.ico
Method	GET
Evidence	HTTP/1.1 500 Internal Server Error
Instances	1
Solution	Revisar el código de fuente de esta página. Implementación de páginas de error personalizadas. Considerar implementar un mecanismo para proveer una única referencia/identificación de error para el cliente (navegador) mientras insertando los detalles en el sitio del navegador y no exponiéndolos al usuario.
Reference	
CWE Id	200
WASC Id	13
Source ID	3
Informational (Low)	Timestamp Disclosure - Unix
Description	A timestamp was disclosed by the application/web server - Unix
URL	http://localhost/odo/fonts/Barlow-Regular.ttf
Method	GET
Evidence	54676332

URL	http://localhost/odo/fonts/Barlow-Regular.ttf
Method	GET
Evidence	76676654
URL	http://localhost/odo/fonts/Barlow-Regular.ttf
Method	GET
Evidence	554766554
Instances	25
Solution	Manually confirm that the timestamp data is not sensitive, and that the data cannot be aggregated to disclose exploitable patterns.
Other information	54676332, which evaluates to: 1971-09-25 15:52:12
Reference	http://projects.webappsec.org/w/page/13246936/Information%20Leakage
CWE Id	200
WASC Id	13
Source ID	3
Informational (Low)	Loosely Scoped Cookie
Description	<p>Cookies can be scoped by domain or path. This check is only concerned with domain scope. The domain scope applied to a cookie determines which domains can access it. For example, a cookie can be scoped strictly to a subdomain e.g. www.nottrusted.com, or loosely scoped to a parent domain e.g. nottrusted.com. In the latter case, any subdomain of nottrusted.com can access the cookie. Loosely scoped cookies are common in mega-applications like google.com and live.com. Cookies set from a subdomain like app.foo.bar are transmitted only to that domain by the browser. However, cookies scoped to a parent-level domain may be transmitted to the parent, or any subdomain of the parent.</p>
URL	http://localhost/odo

Method	GET
URL	http://localhost/odo/usuarios
Method	GET
Instances	6
Solution	Always scope cookies to a FQDN (Fully Qualified Domain Name).
Other information	The origin domain used for comparison was: localhost ci_session=sut09etf4ijenkc81o4852ne9bl8nm3l
Reference	https://tools.ietf.org/html/rfc6265#section-4.1 https://owasp.org/www-project-web-security-testing-guide/v41/4-Web_Application_Security_Testing/06-Session_Management_Testing/02-Testing_for_Cookies_Attributes.html http://code.google.com/p/browsersec/wiki/Part2#Same-origin_policy_for_cookies
CWE Id	565
WASC Id	15
Source ID	3
Informational (Low)	Information Disclosure - Suspicious Comments
Description	The response appears to contain suspicious comments which may help an attacker. Note: Matches made within script blocks or files are against the entire content not only comments.
URL	http://localhost/odo/js/bootstrap.min.js
Met	GET

ho d	
URL	http://localhost/odo/js/jquery34.min.js
M et ho d	GET
Instances	2
Solution	Remove all comments that return information that may help an attacker and fix any underlying problems they refer to.
Other information	<p>The following comment/snippet was identified via the pattern: \bFROM\b</p> <pre>!function(t,e){"object"==typeof exports&&"undefined"!=typeof module?e(exports,require("jquery"),require("popper.js")):"function"==typeof define&&define.amd?define(["exports","jquery","popper.js"],e):e((t=t self).bootstrap={},t.jQuery,t.Popper)}(this,function(t,g,u){"use strict";function i(t,e){for(var ...Script requires at least jQuery v1.9.1 but less than v4.0.0"})(t.Util=_,t.Alert=p,t.Button=P,t.Carousel=It,t.Collapse=bt,t.Dropdown=Jt,t.Modal=ve,t.Popover=sn,t.Scrollspy=Dn,t.Tab=Kn,t.Toast=ni,t.Tooltip=Be,Object.defineProperty(t,"__esModule",{value:!0}});</pre>
Reference	
CWE Id	200
WASC Id	13
Source ID	3

ANEXO F GLOSARIO

Insumo médico. Se refiere al elemento o conjunto de elementos que conforman un producto el cual es empleado para un diagnóstico, tratamiento o prevención de enfermedades.

Internet. Es el conjunto descentralizado de redes de comunicación interconectadas que utilizan la familia de protocolos TCP/IP, lo cual garantiza que las redes físicas heterogéneas que la componen constituyan una red lógica única de alcance mundial.

Web. Es un vocablo inglés que significa red o malla, y se utiliza en el ámbito tecnológico para nombrar una red informática y en general a Internet. También hace referencia a un conjunto de información que se encuentra en una dirección determinada de Internet.

Software. Conjunto de programas y rutinas que permiten a la computadora realizar determinadas tareas, representa el soporte lógico de un sistema informático.

Hardware. Es el conjunto de elementos físicos o materiales que constituyen una computadora, es decir, representa el soporte tangible de un sistema informático.

OWASP. Acronimo de Open Web Application Security Project, es un organismo sin animo de lucro que aglutina empresas, organizaciones educativas y particulares del mundo, con la finalidad de definir guías, estándares, métricas, recomendaciones legales, entre otros, en el ámbito de la seguridad informática.

ANEXO G
GUIA DE INSTALACIÓN DEL SISTEMA

UNIVERSIDAD PÚBLICA DE EL ALTO

Carrera de Odontología

CLINICA ODONTOLÓGICA



S I S T E M A

CAJA-INSUMOS

CLÍNICA ODONTOLÓGICA - UPEA

Versión 1.0

A stylized globe composed of a network of grey lines and dots, representing a global network or data structure, positioned in the lower right quadrant of the cover.

GUIA
DE
INSTALACION

2020

Sistema de emisión de recibos y control de insumos médicos

GUIA DE INSTALACION

Autor: Carlos Daniel Apaza Parisaca

Contenido

1. Requerimientos de instalación	133
1.1. Requerimientos de Hardware.....	133
1.2. Requerimientos de Software y conectividad	133
2.- Instalación del entorno LAMP	133
2.1. Descarga del paquete LAMP	133
2.2. Instalación de los servidores	133
3.- Instalación del sistema	135
3.1. Creación e importación de la base de datos	135
3.2. Configuración del proyecto para el primer arranque	136

1. Requerimientos de instalación

1.1. Requerimientos de Hardware

Los requerimientos mínimos de hardware para el servidor son los siguientes:

MEMORIA RAM	4GB
ALMACENAMIENTO	50GB

1.2. Requerimientos de Software y conectividad

Los requerimientos mínimos de software y conectividad para el servidor son los siguientes:

CONEXIÓN A INTERNET	10Mbps
IP – DOMINIO	IP dedicado o dominio o subdominio
TIPO DE SERVIDOR	1 Servidor Virtual
SISTEMA OPERATIVO	GNU/Linux de preferencia

2.- Instalación del entorno LAMP

LAMP es una infraestructura para provisión de servicios Web cuyos componentes principales son el servidor Web Apache, el gestor de bases de datos MySQL y el lenguaje de programación PHP. Es un entorno pre-configurado para el desarrollo u operación de un sistema orientado a la Web.

2.1. Descarga del paquete LAMP

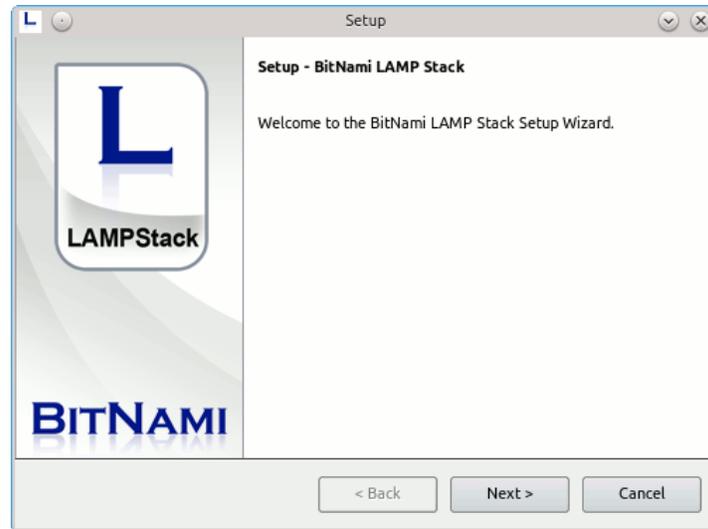
Ingresa a <https://bitnami.com/stack/lamp/installer>, seleccione el paquete de instalación LAMP y descárguelo.

En caso de que quiera instalar el paquete para un entorno Windows, puede descargar el instalador desde <https://bitnami.com/stack/wamp/installer>.

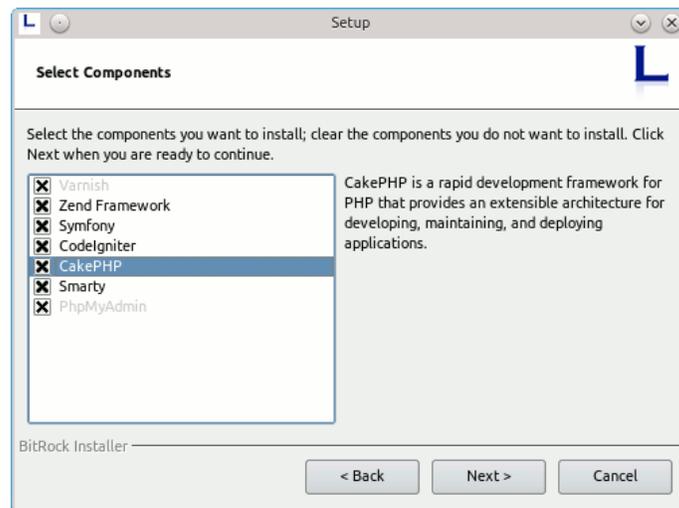
A noviembre de 2020, la última versión de LAMP disponible es 7.4, cuyos componentes son: Apache 2.4, MySQL 8.0, PHP 7.4 y phpMyAdmin 5.0.4.

2.2. Instalación de los servidores

La instalación del entorno LAMP es bastante sencilla. Solo debe arrancar y se abrirá el instalador como se muestra en la siguiente imagen.

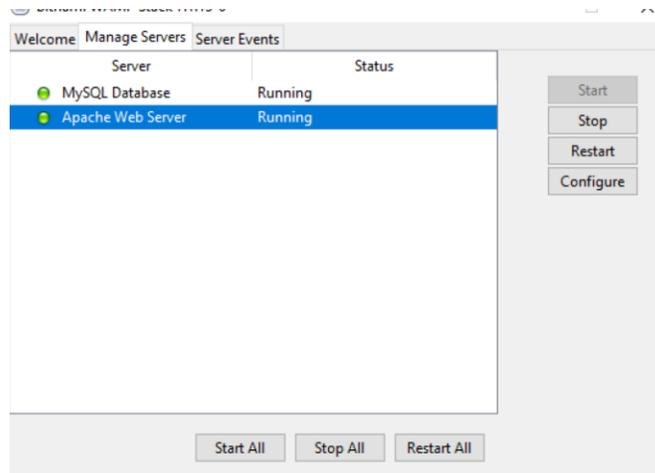


Luego debe seleccionar los componentes que desea instalar, puede dejar la configuración tal cual está y pulsar siguiente hasta que el asistente de instalación le solicite el password para el sistema de gestión de base de datos.



Una vez seleccionado los componentes y definida la contraseña, procederá la instalación hasta finalizar. Una vez finalizado debe abrir su navegador Web con la dirección <http://localhost>, el cual le mostrará una página de bienvenida al entorno LAMP.

Desde el menú del sistema puede acceder al gestor de servidores cuyo nombre es Bitnami Lampstack Manager Tool, el cual muestra un dialogo para arrancar o detener los servidores, como se muestra en la siguiente ilustración



3.- Instalación del sistema

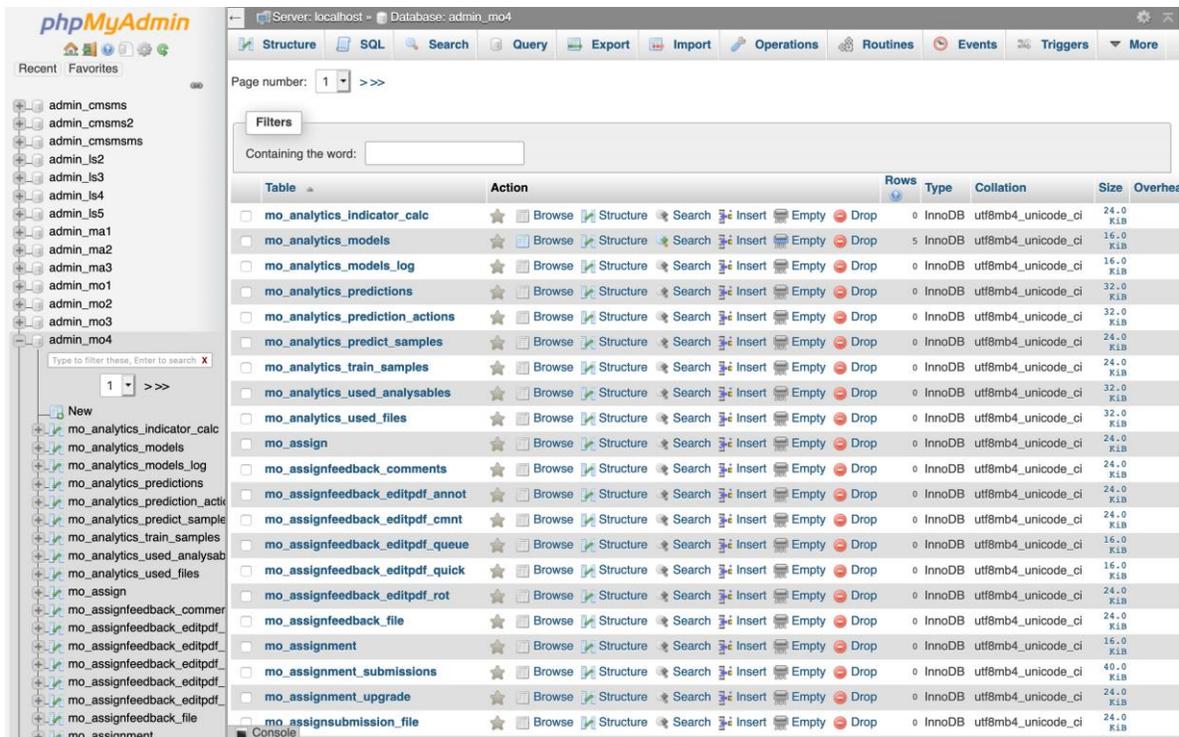
Para instalar el sistema, solamente debe copiar y descomprimir el archivo `odo.zip`, en la carpeta `../bitnami_X.Y/apache/htdocs/odo/`

El código del proyecto está organizado en las siguientes carpetas:

- `application`, contiene el código del proyecto.
- `css`, contiene el código de las hojas de estilo del proyecto.
- `fonts`, contiene los archivos de fuente del proyecto.
- `guias_usuario`, contiene las guías de usuario.
- `img`, contiene las imágenes e iconos utilizados en el proyecto.
- `js`, contiene los scripts en lenguaje Javascript del proyecto.
- `System`, contiene el código del núcleo de Codeigniter

3.1. Creación e importación de la base de datos

Para crear la base de datos e importar el esquema, diríjase a su navegador y abra phpMyAdmin accediendo a la ruta <http://localhost/phpmyadmin>. Debe acceder con el usuario `root` y la contraseña que estableció al instalar el entorno LAMP. Accederá a la vista principal del sistema como se muestra en la ilustración siguiente.



En la parte izquierda seleccione la opción NUEVO (NEW), y se abrirá una página con la lista de todas las bases de datos disponibles y un formulario donde debe especificar 2 campos, el nombre de la base de datos y el collation. Escriba un nombre para su base de datos y selecciones utf8-general-ci para el atributo collation, finalmente haga click en CREAR (CREATE) y se creará la base de datos.

Dado que el esquema está vacío, debe importar el backup o copia del esquema en formato SQL, para esto, diríjase a la pestaña superior de nombre Importar (Import) y seleccione el archivo SQL que contiene el esquema de la base de datos, no modifique ningún atributo mas y haga clic en continuar, el sistema procederá con la importación.

3.2. Configuración del proyecto y primer arranque

Una vez importada la base de datos y descomprimido el código del proyecto, debe editar los siguientes archivos de configuración **config.php** y **database.php** que están en la carpeta **application/config**.

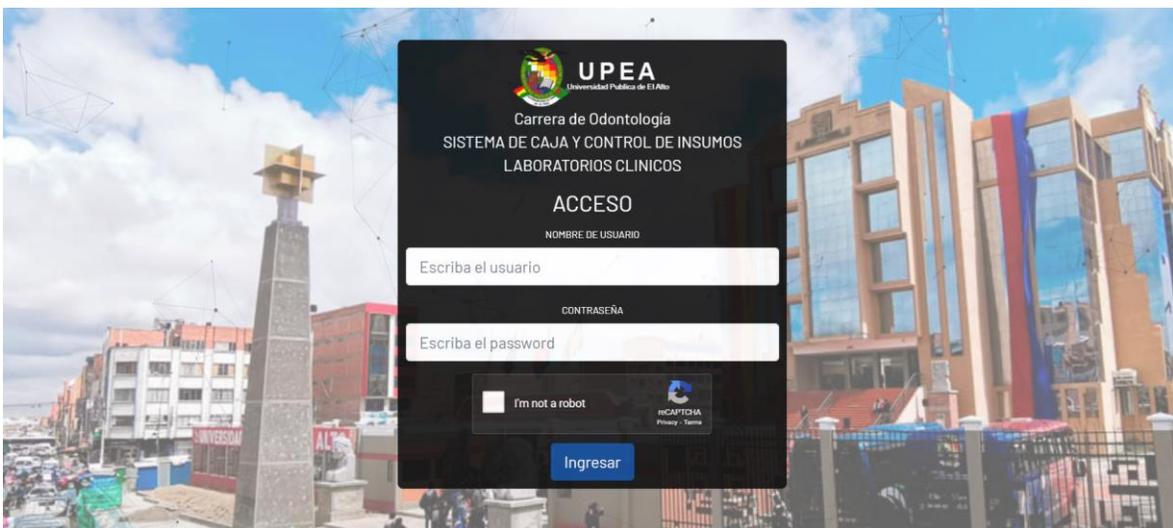
En el archivo **config.php**, debe modificar la URL base de acuerdo a su configuración local de servidor. Por ejemplo en caso de que el proyecto esté en ruta **htdocs/odo**, la variable será:

```
$config['base_url'] = 'http://localhost/odo';
```

En el archivo **database.php** debe modificar las conexiones a la base de datos estableciendo los valores correspondientes a su entorno, como el nombre que le puso a la base de datos y la contraseña. Por ejemplo, si la base de datos tiene por nombre **odonto** y la clave de acceso a la base de datos es **123**, entonces la configuración sería:

```
'hostname' => 'localhost',  
'username' => 'root',  
'password' => '123',  
'database' => 'odonto',  
'dbdriver' => 'mysqli',  
'port' => '3306',
```

Listo, ahora puede arrancar el sistema escribiendo en la barra de dirección del navegador <http://localhost/odo>, y el sistema estará en línea como se muestra a continuación.



ANEXO H
GUIA DE USUARIO

UNIVERSIDAD PÚBLICA DE EL ALTO

Carrera de Odontología

CLINICA ODONTOLÓGICA



S I S T E M A

CAJA-INSUMOS

CLÍNICA ODONTOLÓGICA - UPEA

Versión 1.0

A background graphic of a network diagram, consisting of a grid of grey dots connected by thin grey lines, forming a spherical shape. The dots are arranged in a pattern that suggests a globe or a complex network structure.

GUIA DE USUARIO

2020

Sistema de emisión de recibos y control de insumos médicos

GUIA DE USUARIO

Autor: Carlos Daniel Apaza Parisaca

Contenido

1. Requerimientos de la terminal de usuario	141
2. Acceso al sistema y página inicial	141
3.- Funcionalidades para el Encargado de Caja.....	142
3.1. Gestión de recibos	142
3.2. Gestión de estudiantes	145
3.3. Gestión de docentes	147
4. Funcionalidades para el Encargado de Insumos.....	149
4.1. Gestión de insumos	149
4.2. Gestión de ingreso de insumos.....	151
4.3. Gestión de salida de insumos	153
5. Funcionalidades para el Administrador de Sistema	154
6. Gestión de reportes.....	156

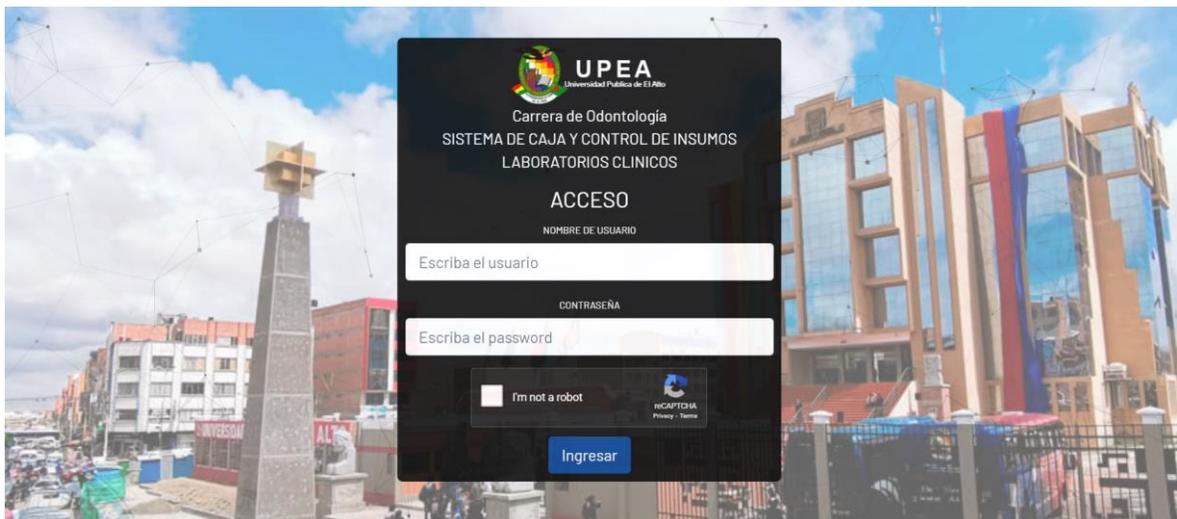
1. Requerimientos de la terminal de usuario

Los requerimientos necesarios para que un usuario pueda acceder al sistema son los siguientes:

PROCESADOR	Intel i3 o más reciente.
MEMORIA RAM	4GB
ALMACENAMIENTO	50GB
CONEXIÓN A INTERNET	2Mbps
SISTEMA OPERATIVO	Windows 7, 8 o 10.
NAVEGADOR WEB	Google Chrome v. 87+, Firefox v. 55+

2. Acceso al sistema y página inicial

Para acceder al sistema, escriba la URL del acceso al sistema. Si no la conoce, consulte con el encargado de la instalación del sistema en su institución. Una vez conectado con el sistema, se presentará el formulario de acceso. Debe escribir su nombre de usuario, contraseña y confirmar el código Captcha, finalmente hacer clic en ingresar. Si tiene problemas para ingresar, verifique si su usuario o password son correctos.



Una vez ingresado, el sistema muestra la página inicial de usuario, con los módulos disponibles (parte izquierda), un resumen de datos (parte derecha). Asimismo, tiene a su disposición el menú superior donde está la lista de módulos también. Una ilustración de la página inicial se muestra a continuación.



En la parte superior derecha tiene acceso las opciones de usuario, desde donde puede actualizar su contraseña, acceder a la guía de usuario o bien cerrar la sesión.

3.- Funcionalidades para el Encargado de Caja

3.1. Gestión de recibos

Una de las funciones principales del encargado de caja es la gestión de los recibos que se emiten por la venta de tratamientos o formularios de historial clínico.

3.1.1. Listar recibos

Al seleccionar el módulo de Gestión de recibos desde el menú superior o el menú de la pantalla de inicio el encargado de caja puede gestionar los recibos desde una lista como se muestra a continuación.

CAJA-INSUMOS INICIO MÓDULOS CUENTA DE USUARIO

GESTION DE PEDIDOS Y RECIBOS

LISTA DE RECIBOS NUEVO PEDIDO

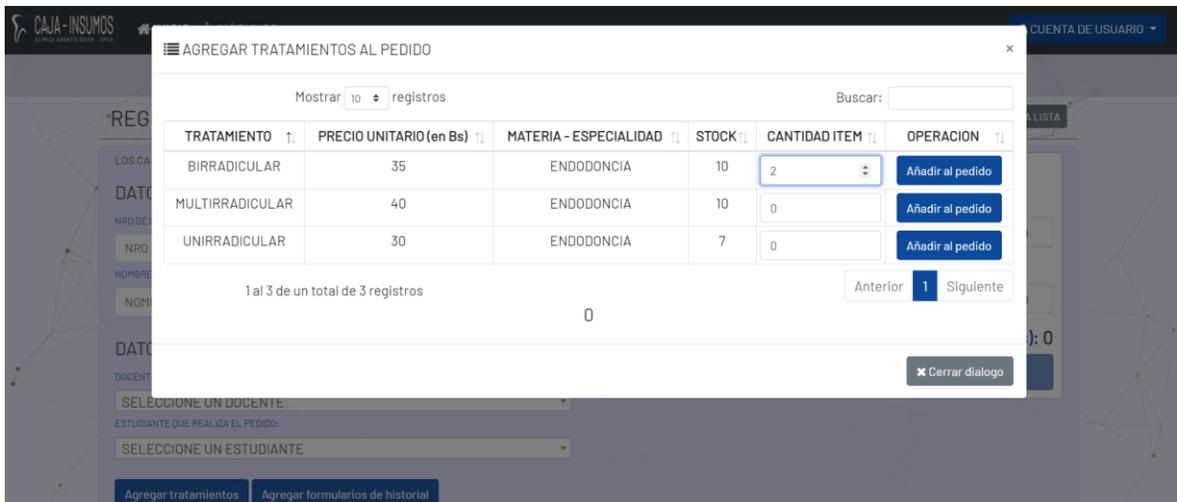
Mostrar 10 registros Buscar:

CODIGO RECIBO	ESTUDIANTE	PACIENTE	FECHA RECIBO	TOTAL RECIBO	ULTIMA ACTUALIZACION	OPERACION
RC-2-2020	MARIA LUISA APAZA CHOQUE	JUAN CARLOS POMIER ROCHA	2020-11-17	70	2020-11-17 12:57:21	OPCION
RC-3-2020	REYNILDA FLORES LEQUELEQUE	FABIOLA MEJIA MORALES	2020-11-17	105	2020-11-17 17:56:59	OPCION
RC-4-2020	REYNILDA FLORES LEQUELEQUE	GUALBERTO POMACUSI RIVERO	2020-11-17	44.5	2020-11-17 18:19:22	OPCION
RC-5-2020	DARWIN FELIX QUISPE CHIQUE	MARIO MAMANI MAMANI	2020-11-18	39.5	2020-11-17 20:14:15	OPCION
RC-6-2020	DARWIN FELIX QUISPE CHIQUE	SUSANA MERLO APAZA	2020-11-18	44.5	2020-11-18 09:25:28	OPCION
RC-7-2020	MIGUEL ALEJANDRO ORTEGA MORALES	HUGO AGUILAR HEREDIA	2020-11-19	35	2020-11-19 16:13:02	OPCION

En cada ítem de la lista tiene el botón de Opción que le permite acceder a dos funcionalidades, como ser: Imprimir el recibo o cancelar el recibo. En caso de que no exista ningún recibo, en la parte superior derecha de la página tiene la opción de crear un nuevo recibo.

3.1.2. Registrar nuevo recibo

En el caso de crear un recibo, haga clic en Nuevo recibo y se mostrará el formulario de registro de recibo, donde debe introducir los datos personales de un paciente, seleccionar el docente que autoriza la compra y el estudiante que está realizando la compra del tratamiento o formulario. Luego, se tienen dos opciones en la parte baja de la página, Agregar Tratamientos o Agregar Formularios de historial al detalle del recibo. En ambos casos debe seleccionar los ítems necesarios y establecer la cantidad de compra como se muestra en la siguiente ilustración. El sistema agregará el ítem y la cantidad definida una vez haga clic en Añadir al pedido. Además se calcula automáticamente los totales y subtotales del recibo.



Una vez agregado los tratamiento o formularios de historial al pedido, se muestra el total del recibo y el detalle de los ítems seleccionados. Finalmente debe hacer clic en Confirmar pedido y emitir recibo como se muestra en la siguiente ilustración.



El sistema le direccionará a la lista de recibos registrados. Como se indicó anteriormente, puede acceder a la opción de Imprimir el recibo o cancelar el recibo, como se muestra en las siguientes ilustraciones respectivamente.

CAJA-INSUMOS INICIO MÓDULOS CUENTA DE USUARIO

GESTION DE PEDIDOS Y RECIBOS

IMPRIMIR RECIBO IMPRIMIR VOLVER A LISTA

UNIVERSIDAD PUBLICA DE EL ALTO
CLINICA ODONTOLÓGICA

RECIBO

Paciente: FABIOLA MEJIA MORALES Estudiante: REYNILDA FLORES LEQUELEQUE
Docente que autoriza: RAMIRO LIMACHI PAZ Fecha: 2020-11-17

RC-3-2020

DETALLE

Item	Precio Unitario (en Bs)	Cantidad	Subtotal (en Bs)
BIRRADICULAR	35	3	105

TOTAL RECIBO Bs.- 105
ORIGINAL

CAJA-INSUMOS INICIO MÓDULOS CUENTA DE USUARIO

ELIMINAR ITEM

¿ESTA SEGURO DE ELIMINAR EL SIGUIENTE ITEM?

Detalle del ítem: RC-3-2020
Última actualización: 2020-11-17 17:56:59

No eliminar Eliminar ítem

LISTA DE RECIBOS

Mostrar 10 registros

CODIGO RECIBO	ESTUDIANTE	OPERACION
RC-2-2020	MARIA LUISA APAZA CHOQUE	JUAN CARLOS POMIER ROCHA
RC-3-2020	REYNILDA FLORES	FABIOLA MEJIA

3.2. Gestión de estudiantes

3.2.1. Listar de estudiantes

El encargado de caja puede acceder a la lista de estudiantes seleccionando Gestión de Estudiantes desde el menú superior o desde el panel inicial. Luego el sistema le muestra el listado de estudiantes registrados hasta el momento como se muestra a continuación.

CÉDULA DE IDENTIDAD	NOMBRE COMPLETO	GENERO	MATRICULA	ULTIMA ACTUALIZACION	OPERACION
6061701 LP	ABEL AMILCAR HUANCA MAMANI	< No establecido >	11004702	2020-11-16 19:47:37	OPCION
6767412 LP	ABEL NUÑEZ ULO	< No establecido >	200003610	2020-11-16 19:47:49	OPCION
7032377 LP	ABIGAIL ALIAGA VARGAS	< No establecido >	200012242	2020-11-16 19:48:15	OPCION
9971412 LP	ABIGAIL GOMEZ SANTA MARIA	< No establecido >	200031307	2020-11-16 19:49:59	OPCION

3.2.2. Registrar, actualizar y eliminar datos de estudiante

Para registrar un estudiante debe seleccionar la opción Nuevo Estudiante ubicado en la parte superior derecha de la lista de estudiantes. El sistema le presenta un formulario de registro como se muestra a continuación.

LOS CAMPOS MARCADOS CON ASTERISCO (*) SON OBLIGATORIOS.

NRO. DOCUMENTO IDENTIDAD: NÚMERO DE CI

EXPEDIDO EN: SELECCIONE UNA OPCION

NRO. MATRICULA UNIVERSITARIA: NÚMERO DE CI O MATRICULA

NOMBRES: NOMBRES DEL ESTUDIANTE

PRIMER APELLIDO: PRIMER APELLIDO

SEGUNDO APELLIDO: SEGUNDO APELLIDO

GENERO: SELECCIONE UNA OPCION

Una vez llenados los datos, debe pulsar en la opción de Guardar datos, en caso contrario, si desea volver a la lista de estudiantes, hacer clic en Volver a lista.

En caso de actualizar los datos de un estudiante, el formulario es similar, solo que se debe acceder desde la opción de cada ítem de la lista de estudiantes, seleccionando la opción Editar. El sistema mostrará el formulario como se ve a continuación.

CAJA-INSUMOS INICIO MÓDULOS CUENTA DE USUARIO

GESTION DE ESTUDIANTES

EDITAR ESTUDIANTE VOLVER A LISTA

LOS CAMPOS MARCADOS CON ASTERISCO (*) SON OBLIGATORIOS.

NRO. DOCUMENTO IDENTIDAD: 10003849 EXPEDIDO EN: La Paz NRO. MATRICULA UNIVERSITARIA: 13001682

NOMBRES: VIANCA LORENA PRIMER APELLIDO: CRUZ

SEGUNDO APELLIDO: BLANCO GENERO: FEMENINO

Guardar datos

En caso de requerir eliminar los datos de un estudiante, a partir de la lista de estudiantes, seleccionar la opción Eliminar de un ítem de la lista, y el sistema mostrará un dialogo de confirmación para dar de baja el registro, tal como se muestra en la siguiente ilustración.

CAJA-INSUMOS INICIO MÓDULOS CUENTA DE USUARIO

ELIMINAR ITEM

¿ESTA SEGURO DE ELIMINAR EL SIGUIENTE ITEM?

Detalle del ítem: ESPERANZA COAQUIRA TICONA

Última actualización: 2020-11-16 19:50:04

No eliminar Eliminar ítem

3.3. Gestión de docentes

3.3.1. Listar de docentes

El encargado de caja puede acceder a la lista de docentes seleccionando Gestión de docentes desde el menú superior o desde el panel inicial. Luego el sistema le muestra el listado de docentes registrados hasta el momento como se muestra a continuación.



3.3.2. Registrar, actualizar y eliminar datos de docentes

Para registrar un docente debe seleccionar la opción Nuevo Docente ubicado en la parte superior derecha de la lista de docentes. El sistema le presenta un formulario de registro como se muestra a continuación.



Una vez llenados los datos, debe pulsar en la opción de Guardar datos, en caso contrario, si desea volver a la lista de docentes, hacer clic en Volver a lista.

En caso de actualizar los datos de un docente, el formulario es similar, solo que se debe acceder desde la opción de cada ítem de la lista de docentes, seleccionando la opción Editar. El sistema mostrará el formulario como se ve a continuación.



En caso de requerir eliminar los datos de un docente, a partir de la lista de docentes, seleccionar la opción Eliminar de un ítem de la lista, y el sistema mostrará un dialogo de confirmación para dar de baja el registro, tal como se muestra en la siguiente ilustración.



4. Funcionalidades para el Encargado de Insumos

4.1. Gestión de insumos

4.1.1. Listar de insumos

El encargado de insumos puede acceder a la lista de insumos seleccionando Gestión de Insumos desde el menú superior o desde el panel inicial. Luego el sistema le muestra el listado de insumos registrados hasta el momento como se muestra a continuación.

INSUMO	MARCA	PROVEEDOR	UNIDAD MEDIDA	PRECIO UNITARIO	STOCK	ULTIMA ACTUALIZACION	OPERACION
PARACETAMOL	BAYER	ALBUS	MT	45	133	2020-11-20 08:59:00	OPCION
VENDAS	BAYER	ALBUS	MT	5	100	2020-11-15 14:07:08	OPCION

4.1.2. Registrar, actualizar y eliminar datos de insumos

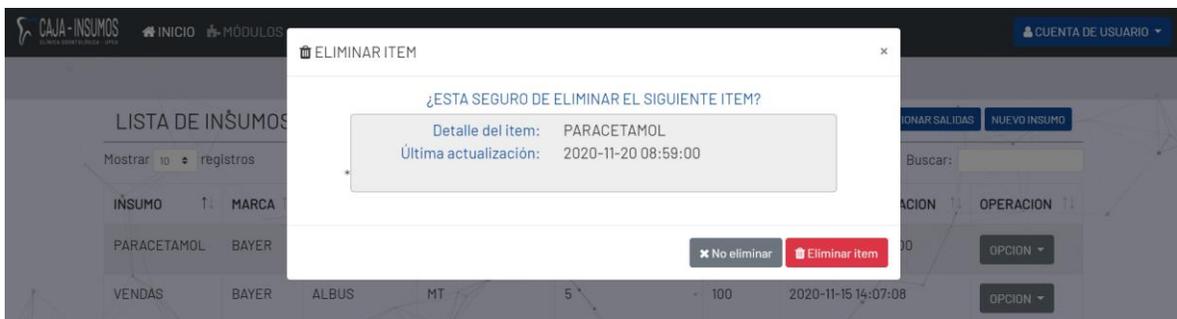
Para registrar un insumo debe seleccionar la opción Nuevo Insumo ubicado en la parte superior derecha de la lista de insumos. El sistema le presenta un formulario de registro como se muestra a continuación.

Una vez llenados los datos, debe pulsar en la opción de Guardar datos, en caso contrario, si desea volver a la lista de insumos, hacer clic en Volver a lista.

En caso de actualizar los datos de un insumo, el formulario es similar, solo que se debe acceder desde la opción de cada ítem de la lista de insumo, seleccionando la opción Editar. El sistema mostrará el formulario como se ve a continuación.



En caso de requerir eliminar los datos de un insumo, a partir de la lista de insumos, seleccionar la opción Eliminar de un ítem de la lista, y el sistema mostrará un dialogo de confirmación para dar de baja el registro, tal como se muestra en la siguiente ilustración.



4.2. Gestión de ingreso de insumos

4.2.1. Listar de ingresos insumos

El encargado de insumos puede acceder a la lista de ingresos insumos seleccionando la opción Gestión de Ingresos de un ítem de la lista de insumos. Luego el sistema le muestra el listado de ingresos de insumos registrados hasta el momento como se muestra a continuación.



4.2.2. Registrar, actualizar y eliminar ingresos de insumos

Para registrar un ingreso de insumo debe seleccionar la opción Nuevo Ingreso de Insumo ubicado en la parte superior derecha de la lista de ingresos de insumo. El sistema le presenta un formulario de registro como se muestra a continuación.



Una vez llenados los datos, debe pulsar en la opción de Guardar datos, en caso contrario, si desea volver a la lista de ingresos de insumos, hacer clic en Volver a lista.

En caso de requerir eliminar los datos de un ingreso de insumo, a partir de la lista de ingresos, seleccionar la opción Eliminar de un ítem de la lista, y el sistema mostrará un dialogo de confirmación para dar de baja el registro, tal como se muestra en la siguiente ilustración.



4.3. Gestión de salida de insumos

4.3.1. Listar de salidas insumos

El encargado de insumos puede acceder a la lista de salidas insumos seleccionando la opción Gestión de Salidas desde la parte superior derecha de la lista de insumos. Luego el sistema le muestra el listado de salidas de insumos registrados por recibos despachados hasta el momento, como se muestra a continuación. Se muestran en color amarillo los recibos por despachar, y los recibos despachados en color verde.

CODIGO RECIBO	ESTUDIANTE	PACIENTE	FECHA RECIBO	TOTAL RECIBO	DESPACHADO	ULTIMA ACTUALIZACION	OPERACION
RC-2-2020	MARIA LUISA APAZA CHOQUE	JUAN CARLOS POMIER ROCHA	2020-11-17	70	NO	2020-11-17 12:57:21	OPCION
RC-3-2020	REYNILDA FLORES LEQUELEQUE	FABIOLA MEJIA MORALES	2020-11-17	105	NO	2020-11-17 17:56:59	OPCION
RC-4-2020	REYNILDA FLORES LEQUELEQUE	GUALBERTO POMACUSI RIVERO	2020-11-17	44.5	NO	2020-11-17 18:19:22	OPCION

4.3.2. Despachar recibo

Para registrar una salida de insumo debe seleccionar la opción Despachar recibo ubicado en botón de opción de cada ítem de la lista. El sistema le presenta un formulario de registro con el resumen del recibo. Para despachar un recibo debe definir la salida de insumos haciendo clic en la opción Definir Salidas de Insumos.

Una vez ahí, el sistema presenta un formulario para despachar el recibo seleccionando ítems del stock de insumos, como se muestra en la siguiente ilustración. El encargado debe seleccionar el insumo y la cantidad de salida y añadir cada ítem a la salida, finalmente hacer clic en Guardar y finalizar para concluir.

NOMBRE INSUMO	STOCK	UNIDAD MEDIDA	CANTIDAD A DESPACHAR	OPERACION
PARACETAMOL	133	MT	0	Añadir a salida
VENDAS	100	MT	0	Añadir a salida

5. Funcionalidades para el Administrador de Sistema

El administrador del sistema tiene a su cargo la gestión de: usuarios, unidades organizacionales, cargos, roles, países, proveedores, marcas, unidades de medida, servicios, formularios de historial, tratamientos y materias – especialidades.

Para cada una de estos módulos, el administrador puede efectuar el listado, registro, actualización o eliminación de ítems.

5.1. Listar de items

El administrador de sistema puede acceder a la lista de items seleccionando Gestión de <<Nombre del módulo>> desde el menú superior o desde el panel inicial. Luego el sistema le muestra el listado de items registrados hasta el momento. La interfaz es similar para todos los casos como se muestra a continuación.



5.1.2. Registrar, actualizar y eliminar datos de ítems

Para registrar un ítem debe seleccionar la opción Nuevo <<Nombre del ítem>> ubicado en la parte superior derecha de la lista de ítems. El sistema le presenta un formulario de registro, similar al mostrado a continuación.



Una vez llenados los datos, debe pulsar en la opción de Guardar datos, en caso contrario, si desea volver a la lista de ítems, hacer clic en Volver a lista.

En caso de actualizar los datos de un ítem, el formulario es similar, solo que se debe acceder desde la opción de cada ítem de la lista de ítems, seleccionando la opción Editar. El sistema mostrará el formulario como se ve a continuación.



En caso de requerir eliminar los datos de un ítem, a partir de la lista de ítems, seleccionar la opción Eliminar de un ítem de la lista, y el sistema mostrará un dialogo de confirmación para dar de baja el registro, tal como se muestra en la siguiente ilustración.



6. Gestión de reportes

Para la gestión de reportes, el usuario tiene acceso a la lista general de reportes, donde el sistema tiene habilitado los reportes de acuerdo al usuario. Solamente debe seleccionar el reporte y definir los parámetros respectivos para poder visualizar e imprimir el reporte. La lista de reportes se muestra a continuación.

CAJA - INSUMOS INICIO MÓDULOS CUENTA DE USUARIO

GESTION DE REPORTES

LISTA DE REPORTES

ESTADOS FINANCIEROS

#	NOMBRE DEL REPORTE	OPERACION
1	INGRESO ECONOMICO GENERAL (POR DIA)	GENERAR REPORTE
2	INGRESO ECONOMICO GENERAL (POR RANGO DE FECHAS)	GENERAR REPORTE
3	INGRESO ECONOMICO POR VENTA DE TRATAMIENTOS (POR DIA)	GENERAR REPORTE
4	INGRESO ECONOMICO POR VENTA DE TRATAMIENTOS (POR RANGO DE FECHAS)	GENERAR REPORTE
5	INGRESO ECONOMICO POR VENTA DE FORMULARIOS DE HISTORIAL (POR DIA)	GENERAR REPORTE
6	INGRESO ECONOMICO POR VENTA DE FORMULARIOS DE HISTORIAL (POR RANGO DE FECHAS)	GENERAR REPORTE

Los reportes tienen la opción de impresión. Una vez generado el reporte puede imprimirlo, haciendo clic a la opción Imprimir correspondiente, como se muestra a continuación.

GESTION DE REPORTES

REPORTE TRATAMIENTOS MAS VENDIDOS

[IMPRIMIR REPORTE](#) [VOLVER A LISTA](#)

 UNIVERSIDAD PÚBLICA DE EL ALTO - UPEA
Carrera de Odontología
CLINICA ODONTOLÓGICA

REPORTE: REPORTE TRATAMIENTOS MAS VENDIDOS
 GENERADO CON DATOS: TOTAL A LA FECHA 2020-11-27
 FECHA Y HORA EMISION DE REPORTE: 2020-11-27, 01:11:14

TRATAMIENTO	CANTIDAD POR TRATAMIENTOS
BIRRADICULAR	8
MULTIRRADICULAR	2
TOTALES:	10

GRAFICA - TRATAMIENTOS MAS VENDIDOS

ANEXO I
DOCUMENTOS DE RESPALDO

Ciudad de El Alto, 23 de noviembre de 2020

Señor
Ing. David Carlos Mamani Quispe
DIRECTOR
CARRERA INGENIERÍA DE SISTEMAS – UPEA
Presente.-

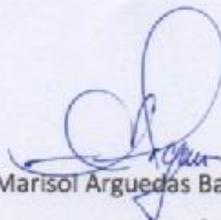
REF.: AVAL DE CONFORMIDAD

Distinguido ingeniero:

Tengo a bien dirigirme a su persona para comunicarle mi conformidad del trabajo final del Proyecto de Grado, titulado **“SISTEMA DE EMISIÓN DE RECIBOS, CONTROL EN DISTRIBUCIÓN DE INSUMOS MEDICOS EN LABORATORIOS CLINICOS CASO: CARRERA DE ODONTOLOGÍA”** elaborado por el universitario: **CARLOS DANIEL APAZA PARISACA** con cedula de identidad: **4892831 L.P.** y registro universitario: **6771**, para su defensa pública y evaluación correspondiente a la materia de Taller de Licenciatura II, de acuerdo al reglamento vigente de la Carrera de Ingeniería de Sistemas de la Universidad Pública de El Alto.

Sin otro particular, me despido de usted.

Atentamente.



Ing. Marisol Arguedas Balladares

TUTOR METODOLÓGICO

Ciudad de El Alto, 23 de noviembre de 2020

Señor
Ing. David Carlos Mamani Quispe
DIRECTOR
CARRERA INGENIERÍA DE SISTEMAS – UPEA
Presente.-

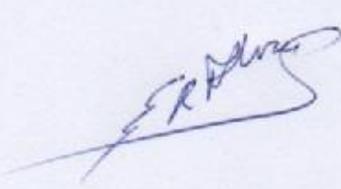
REF.: AVAL DE CONFORMIDAD

Distinguido ingeniero:

Tengo a bien dirigirme a su persona para comunicarle mi conformidad del trabajo final del Proyecto de Grado, titulado **"SISTEMA DE EMISIÓN DE RECIBOS, CONTROL EN DISTRIBUCIÓN DE INSUMOS MEDICOS EN LABORATORIOS CLINICOS CASO: CARRERA DE ODONTOLOGÍA"**, elaborado por el universitario: **CARLOS DANIEL APAZA PARISACA**, con cedula de identidad: **4892831 L.P.** y registro universitario: **6771**, para su defensa pública y evaluación correspondiente a la materia de Taller de Licenciatura II, de acuerdo al reglamento vigente de la Carrera de Ingeniería de Sistemas de la Universidad Pública de El Alto.

Sin otro particular, me despido de usted.

Atentamente.



Ing. Eduardo Rolando Flores Rodriguez

TUTOR ESPECIALISTA

Ciudad de El Alto, 23 de noviembre de 2020

Señor
Ing. David Carlos Mamani Quispe
DIRECTOR
CARRERA INGENIERÍA DE SISTEMAS – UPEA
Presente.-

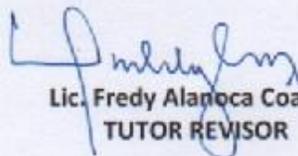
REF.: AVAL DE CONFORMIDAD

Distinguido ingeniero:

Tengo a bien dirigirme a su persona para comunicarle mi conformidad del trabajo final del Proyecto de Grado, titulado **“SISTEMA DE EMISIÓN DE RECIBOS, CONTROL EN DISTRIBUCIÓN DE INSUMOS MEDICOS EN LABORATORIOS CLINICOS CASO: CARRERA DE ODONTOLOGÍA”** elaborado por el universitario: **CARLOS DANIEL APAZA PARISACA** con cedula de identidad: **4892831 L.P.** y registro universitario: **6771**, para su defensa pública y evaluación correspondiente a la materia de Taller de Licenciatura II, de acuerdo al reglamento vigente de la Carrera de Ingeniería de Sistemas de la Universidad Pública de El Alto.

Sin otro particular, me despido de usted.

Atentamente.


Lic. Fredy Alanoca Coareti
TUTOR REVISOR



Universidad Pública de El Alto

Creada por Ley 2115 del 5 de Septiembre de 2000 y Autónoma por Ley 2556 del 12 de Noviembre de 2003

Carrera de Odontología

ODO/ 01/2020

CERTIFICADO

El suscrito Dirección de la Carrera de Odontología de la Universidad Pública de El Alto.

CERTIFICA:

Que, siendo auspiciantes del proyecto de grado **"SISTEMA DE EMISIÓN DE RECIBOS, CONTROL EN DISTRIBUCIÓN DE INSUMOS MEDICOS EN LABORATORIOS CLINICOS CASO: CARRERA DE ODONTOLOGÍA"**, elaborado por el universitario: **CARLOS DANIEL APAZA PARISACA**, con cedula de identidad: **4892831 L.P.** y registro universitario: **6771**, es grato informar que el proceso de implementación del sistema web ha superado satisfactoriamente, asimismo, las pruebas técnicas y los requerimientos de la institución, por lo que extendo el presente documento como constancia de aceptación y finalización del proyecto.

Autorizo al Sr. **CARLOS DANIEL APAZA PARISACA**, puede hacer uso de este documento para los fines pertinentes en la Universidad Pública de El Alto.

Atentamente.

El Alto, 23 de noviembre 2020

C.C.T./elvira
File Personal
C.c.: Archivo


Bernardo Choque Huanca
 EJECUTIVO
 CENTRO DE ESTUDIANTES ODONTOLOGIA

