



**FACULTAD DE INGENIERIA DE SISTEMAS
ESCUELA PROFESIONAL DE INGENIERIA DE SISTEMAS**

INFORME FINAL DE TESIS

**“DESARROLLO DE UN SISTEMA WEB PARA EL PROCESO DE VENTAS
EN LA BOTICA “EL GRAN PODER”, CARABAYLLO, LIMA 2020”**

PARA OPTAR EL TÍTULO PROFESIONAL DE:
Ingeniero de Sistemas.

AUTOR:
Br. Martín Diego Pinedo Dávila.

ASESOR:
Mg. Juan Carlos Lázaro Guillermo.

LÍNEA DE INVESTIGACIÓN:
Sistemas de Gestión de Información y Conocimiento

Sub línea:
Desarrollo de Software y Calidad de software.

UCAYALI - PERÚ 2020



Mg. David Alfonso Ponce López
Presidente



Mg. Adrián Marcelo Sifuentes Rosales
Secretario



Dr. Jaime Augusto Rojas Elescano
Vocal



Mg. Juan Carlos Lázaro Guillermo
Asesor

Dedicatoria

El presente informe final de tesis está dedicado en primer lugar a dios, que me ha brindado la fortaleza y sabiduría ante los momentos adversos y así poder culminar mi carrera profesional.

A mis amados padres Mayer y Moabita, por brindarme su apoyo incondicional en todo momento y sus consejos para hacer de mí una mejor persona y profesional.

A mis amadas hijas Daphne y Gabriela por sus palabras, confianza y apoyo incondicional, por su amor y brindarme el tiempo necesario para realizarme profesionalmente.

A mi amada esposa Dina por sus palabras de aliento, confianza y apoyo incondicional, por su amor y brindarme el tiempo necesario para realizarme profesionalmente.

Agradecimiento

Agradezco a Dios por protegerme durante todo mi camino y darme fuerzas para superar obstáculos y dificultades a lo largo de mi vida.

A mis padres Mayer y Moabita, que siempre me han demostrado su afecto y cariño y me han enseñado a no rendirme ante las adversidades de la vida y a seguir con pasión mis propósitos en la vida.

A mis amadas hijas Daphne y Gabriela que con su amor incondicional son mi gran motivación para seguir adelante y superarme cada día como persona, padre y profesional.

A mi amada esposa Dina, por su apoyo, sus consejos y palabras de aliento para seguir adelante con este proyecto de tesis.

Al Mg. Juan Carlos, asesor de tesis, por su valiosa guía y asesoramiento en la realización de esta tesis.

A todas las personas que me apoyaron en la realización de esta tesis.

“AÑO DEL BICENTENARIO DEL PERU: 200 AÑOS DE INDEPENDENCIA “

**COORDINACIÓN DE INVESTIGACIÓN DE LA UNIVERSIDAD
PRIVADA DE PUCALLPA
Constancia de Originalidad de trabajo de Investigación N° 40**

Pucallpa 31 de mayo del 2021

Yo, Dr. JAIME AUGUSTO ROJAS ELESCANO, informo a la decanatura y a quien corresponda que se presentó a mi despacho el informe de tesis titulado: “DESARROLLO DE UN SISTEMA WEB PARA EL PROCESO DE VENTAS EN LA BOTICA “EL GRAN PODER” CARABAYLLO, LIMA 2020” perteneciente al bachiller, MARTIN DIEGO PINEDO DAVILA.

Habiendo realizado la verificación de coincidencia con el Software Antiplagio PlagScan, los resultados de similitud fueron **28.2%**. El cual está en los parámetros aceptados por las normas de la Universidad Privada de Pucallpa, que es máximo el 30%, por consiguiente, esta Coordinación da su aprobación de conformidad de la aplicación de la prueba de similitud y se autoriza a los bachilleres a continuar con el trámite administrativo correspondiente.

Es todo por informar a su despacho señora Decana.

Atentamente,



Dr. Jaime Augusto Rojas Elescano
Coordinador de Investigación de la Facultad de CCyA

Constancia de Originalidad

Yo, MARTIN DIEGO PINEDO DAVILA, identificado con DNI N° 44081762; egresado de la Escuela Profesional de Ingeniería de Sistemas de la Facultad de Ingeniería de Sistemas, de la Universidad Privada de Pucallpa.

Declaro bajo juramento que:

Soy autor de la tesis titulada: "DESARROLLO DE UN SISTEMA WEB PARA EL PROCESO DE VENTAS EN LA BOTICA "EL GRAN PODER", CARABAYLLO, LIMA 2020"

- 1) La cual presento para optar el título profesional de Ingeniero de Sistemas.
- 2) He respetado las normas internacionales de citas y referencias para las fuentes consultadas. Por tanto, la tesis no ha sido plagiada ni total ni parcialmente.
- 3) La tesis no ha sido autoplagiada; es decir, no ha sido publicada ni presentada anteriormente para obtener algún grado académico previo o título profesional.
- 4) Los datos presentados en los resultados son reales, no han sido falseados, ni duplicados, ni copiados y por tanto los resultados que se presenten en la tesis se constituirán en aportes a la realidad investigada.

De identificarse fraude (datos falsos), plagio (información sin citar a autores), autoplagio (presentar como nuevo algún trabajo de investigación propio que ya ha sido publicado), piratería (uso ilegal de información ajena) o falsificación (representar falsamente las ideas de otros), asumo las consecuencias y sanciones que de mi acción se deriven, sometiéndome a la normatividad vigente de la Universidad Privada de Pucallpa.

Pucallpa, 28 de marzo del 2021


MARTIN DIEGO PINEDO DAVILA
DNI N° 44081762

Resumen

El presente trabajo de tesis se realizó en la Botica “El Gran Poder” E.I.R.L, es una empresa privada ubicada en el distrito de Carabaylo, Lima, su negocio es brindar servicio hacia el público en general, referente a la venta de medicamentos. Este es el principal proceso de la empresa, el cual presento problemas en el tiempo de atención de clientes, control de stock de medicamentos, demora en obtener un reporte de ventas, la emisión de comprobantes de pago y el control ingresos y egresos de caja diaria. Para solucionar este problema se analizó y desarrollo un sistema automatizado que satisface todas las necesidades del negocio, “IpharmaExpress”, teniendo como objetivo general: Determinar la mejora del desarrollo del sistema web en el proceso de venta de la Botica el Gran Poder, Carabaylo, Lima 2020.

Para la construcción del sistema “IpharmaExpress”, se identificaron a los usuarios potenciales, así como los intereses de información de estos y sus requerimientos de negocio que se tomaron como base para el diseño, modelado y la elaboración para el sistema, todo esto bajo el marco de referencia de la metodología RUP y seleccionando los atributos de calidad de software “Adecuación Funcional, Fiabilidad, Eficiencia en el Desempeño, Usabilidad, Mantenibilidad” para medir la calidad del sistema “Ipharmaexpress” basado en el estándar ISO/IEC 25000 SQuaRE.

Una vez desarrollado el sistema se concluyó lo siguiente: Una mejora en el tiempo de atención en un 50% según lo analizado en el diagrama de actividades TOBE del proceso de ventas.

Como resultado se realizó el análisis de calidad de software aplicando la norma ISO/IEC 25000 a los atributos de “Adecuación Funcional, Fiabilidad, Eficiencia en el Desempeño, Usabilidad, Mantenibilidad” enfocados en la calidad interna y externa del sistema web Ipharmaexpress obteniendo un 7,55 y 9,08 de puntaje respectivamente, haciendo un valor total de 8,31 sobre 10 lo que representa que el sistema tuvo un nivel de puntuación aceptable.

Abstract

This thesis work was carried out at the Botica "El Gran Poder" E.I.R.L, it is a private company located in the district of Carabayllo, Lima, its business is to provide service to the general public, referring to the sale of medicines. This is the main process of the company, which presented problems in customer service time, drug stock control, delay in obtaining a sales report, the issuance of payment vouchers and the control of income and daily cash expenses. . To solve this problem, an automated system that meets all business needs, "IpharmaExpress", was analyzed and developed, with the general objective of:

Determine the improvement of the development of the web system in the sale process of the Botica el Gran Poder, Carabayllo, Lima 2020.

For the construction of the "IpharmaExpress" system, potential users were identified, as well as their information interests and their business requirements that were taken as the basis for the design, modeling and development of the system, all under the framework of the RUP methodology and selecting the software quality attributes "Functional Adequacy, Reliability, Performance Efficiency, Usability, Maintainability" to measure the quality of the "Ipharmaexpress" system based on the ISO / IEC 25000 SQuaRE standard.

Once the system was developed, the following was concluded: A 50% improvement in service time as analyzed in the TOBE activities diagram of the sales process.

As a result, the software quality analysis was carried out applying the ISO / IEC 25000 standard to the attributes of "Functional Adequacy, Reliability, Performance Efficiency, Usability, Maintainability" focused on the internal and external quality of the Ipharmaexpress web system, obtaining a 7, 55 and 9.08 score respectively, making a total value of 8.31 out of 10 which represents that the system had an acceptable level of score.

Índice

Portada	i
Dedicatoria.....	iii
Agradecimiento.....	iv
Constancia de Originalidad	v
Resumen.....	vii
Abstract.....	viii
Introducción	xii
Índice de tablas.....	xiii
Índice de figuras	xiv
CAPÍTULO I: GENERALIDADES.....	1
1.1 Problemática	1
1.1.1 Problema General	3
1.1.2 Problema Especifico	3
1.2 Objetivo general.....	4
1.3 Objetivos específicos	4
1.4 Resultados esperados	4
1.5 Herramientas utilizadas	5
1.6 Métodos y procedimientos.....	7
1.6.1 Metodología de desarrollo del proyecto	7
1.7 Alcance	19
1.8 Limitaciones	19
1.8.1 Limitación espacial	19
1.8.2 Limitación temporal	19
1.8.3 Limitación teórica.....	20
1.9 Riesgos	20
1.10 Justificación y viabilidad del proyecto	20
1.10.1 Justificación	20
1.10.1.1 Justificación Teórica	20
1.10.2 Viabilidad del proyecto	21
CAPÍTULO II: MARCO DE REFERENCIA	23
2.1 Marco conceptual.....	23
2.1.1 Antecedentes del Problema	23
2.1.2 Bases teóricas	30
2.2 Marco regulatorio / legal	50
2.2.1 Ley de boticas y farmacias.....	50
2.2.2 Protocolo para efectuar la atención farmacéutica en pacientes con sospecha de coronavirus “covid-19”	52
2.2.3 Facturación Electrónica	53
2.2.4 Derecho de autor.....	54

2.2.5	Tasa Única de Procedimientos Administrativos UPP	55
CAPÍTULO III: PROCESOS DEL SISTEMA.....		56
3.1	Introducción.....	56
3.2	Recopilación y análisis de la información	57
3.2.1	Recopilación de procesos anteriores.....	57
3.3	Diagramas de los procesos del negocio	62
3.3.1	Diagrama macro de gestión de ventas	62
3.3.2	Diagrama macro de gestión de medicamentos	62
3.3.3.	Diagrama macro de gestión de operadores móviles.....	62
3.3.4	Diagrama macro de gestión de operadores bancarios.....	63
3.4	Conclusiones.....	63
CAPÍTULO IV: ANÁLISIS DE REQUISITOS DEL SISTEMA.....		65
4.1	Descripción del sistema.....	65
4.2	Reuniones principales	66
4.3	Objetivos del sistema.....	66
4.4	Catálogo de requisitos del sistema	67
4.4.1	Requisitos funcionales del sistema.....	67
4.5	Definición de actores	68
4.6	Diagrama de casos de uso	69
4.7	Casos de uso del sistema.....	70
4.8	Diagrama de actividades	78
4.9	Reglas de negocio	79
CAPÍTULO V: DISEÑO DEL SISTEMA		80
5.1	Programación.....	80
5.2	Interfaces graficas.....	110
CAPITULO VI: ARQUITECTURA		116
6.1	Alcance	116
6.2	Definición, acrónimos y abreviaturas	116
6.2.1	Definición.....	116
6.2.2	Acrónimos	117
6.2.3	Abreviaturas	117
6.3	Referencias	117
6.4	Visión general	118
6.5	Representación de la arquitectura	119
6.6	Metas y restricciones de la arquitectura.....	119
6.7	Vistas o planos.....	120
6.8	Implementación.....	121
6.9	Calidad	121
6.10	Diseño de base de datos	124

6.10.1 Script de tablas	124
6.11 Diagrama de clases del sistema.....	145
Conclusiones	146
Observaciones.....	147
Recomendaciones y trabajos futuros.....	147
REFERENCIAS BIBLIOGRÁFICAS.....	148
ANEXOS.....	150
Anexo 1: Matriz de consistencia	150
Anexo 2: Instrumentos de aplicación	151

Introducción

En el primer capítulo se describe el negocio y el proceso de ventas de la botica “El Gran Poder”, se plantea el problema y se establece el objetivo general y los objetivos específicos de la presente tesis, así como la justificación, los alcances, riesgos y los límites de esta.

El segundo capítulo se refiere al marco teórico, conceptual y marco legal en donde se tiene el conjunto de leyes, normas y reglamentos que le dan fundamento a nuestra investigación.

En el tercer capítulo se expone los procesos del sistema y se recopila la información necesaria para el análisis.

En el cuarto capítulo se analiza los requisitos del sistema, los cuales se obtienen a través de entrevistas con los principales actores, los cuales se modelan, clasifican y se toman como base para elaborar los diagramas de software con UML 2.1, necesarios según el marco de la metodología RUP.

En el quinto capítulo se presenta fragmentos de código de la aplicación programados en PHP de las clases más representativas y se muestran las interfaces gráficas construidas del sistema. En el sexto y último capítulo se presenta los planos de arquitectura, los diseños usados para la construcción de la base de datos y el diagrama lógico de clases del sistema, así como también las conclusiones y recomendaciones.

Índice de tablas

Tabla 1. Herramientas TIC utilizadas	6
Tabla 2. Fases de ejecución del proyecto.....	7
Tabla 3. Porcentaje referencial al nivel de importancia	10
Tabla 4. Características de calidad interna seleccionadas.	11
Tabla 5. Características de calidad externa seleccionadas.	12
Tabla 6. Subcaracterísticas de calidad seleccionadas.	14
Tabla 7. Subcaracterísticas de calidad externa seleccionados.....	16
Tabla 8. Características de calidad interna seleccionadas ponderadas.....	17
Tabla 9. Características de calidad externa seleccionadas ponderadas.....	18
Tabla 10. Niveles de puntuación aplicados a la medición de la calidad de software.	18
Tabla 11. Descripción de columnas de las matrices de calidad.....	19
Tabla 12. Riesgos del proyecto.....	20
Tabla 13. Métricas para la calidad Interna/Externa	47
Tabla 14. Tabla de normas de la ley de farmacias y boticas.	52
Tabla 15. Anexos normas legales - Factura Electrónica	54
Tabla 16. Requerimiento actor AC1.	60
Tabla 17. Requerimiento actor AC2.....	60
Tabla 18. Requerimiento actor AC3.....	61
Tabla 19. Requerimiento Actor AC4	61
Tabla 20. Actores del sistema	68
Tabla 21. Documentos de análisis realizados.....	118
Tabla 22. Resultado final del análisis de calidad de software	122
Tabla 23. Resumen de matriz de calidad.....	123
Tabla 26. Métricas para la característica de calidad de Educación funcional	152
Tabla 27. Métricas para la característica de calidad Fiabilidad	157
Tabla 28. Métricas para la característica de calidad Eficiencia en el desempeño	161
Tabla 29. Métricas para la característica de calidad Usabilidad	167
Tabla 30. Métricas para la característica de calidad Mantenibilidad.....	173
Tabla 31. Matriz para evaluación de calidad Interna.	184
Tabla 32. Matriz para la evaluación de calidad Externa.....	195

Índice de figuras

Figura 1. Ubicación del establecimiento comercial.....	1
Figura 2. Programa Excel Control de Botica Actual	2
Figura 3. Estructura genérica usada por el modelo de calidad.....	8
Figura 4. Indicadores de calidad del ISO-SQUARE 25010	8
Figura 5. Indicadores de calidad en uso.....	9
Figura 6. Ciclo de desarrollo bajo el paradigma POO.....	31
Figura 7. Arquitectura web en PHP	32
Figura 8. Ciclo de vida de MVC	33
Figura 9. Esfuerzos en actividades según fase del proyecto.....	40
Figura 10. División del ISO/IEC 25000 SQUARE	45
Figura 11. Modelo de referencia para la evaluación de calidad del producto de software.....	48
Figura 12. Modelo de referencia para la evaluación de calidad del producto software ..	50
Figura 13. Registro de ventas excel	56
Figura 14. Reporte de registro de ventas por vendedor.....	57
Figura 15. Reporte de productos.....	58
Figura 16. Gestión de ventas.....	62
Figura 17. Gestión de medicamentos	62
Figura 18. Gestión de operador móvil	63
Figura 19. Gestión de operador bancario.....	63
Figura 20. Catálogo de requisitos del sistema	67
Figura 21. Diagrama de caso de uso del sistema proceso de ventas.....	69
Figura 22. Caso de uso del sistema reportes.....	70
Figura 23. Caso de uso Gestionar Caja	70
Figura 24. Caso de uso Gestionar Configuraciones.....	71
Figura 25. Caso de uso Gestionar Clientes	72
Figura 26. Caso de uso Documentos Electrónicos.....	72
Figura 27. Caso de uso Gestionar Medicamentos.....	73
Figura 28. Casos de uso Gestionar Usuarios	73
Figura 29. Caso de uso Gestionar Venta	74
Figura 30. Diagrama de objetos - Cliente.....	74
Figura 31. Diagrama de objetos - Configuración.....	75
Figura 32. Diagrama de objetos - Usuario	75
Figura 33. Diagrama de objetos - Medicamento	76
Figura 34. Diagrama de objetos - Documento Electrónico.....	76
Figura 35. Diagrama de Matrix de Trazabilidad Caso de uso vs Requerimiento.....	77
Figura 36. Diagrama de dominio	78
Figura 37. Diagrama de Actividades - ASIS del Proceso de Ventas	78
Figura 38. Diagrama de Actividades - TOBE del Proceso de Ventas.....	79
Figura 39. Diagrama de reglas de negocio	79
Figura 40. Ventana de bienvenida al sistema	110
Figura 41. Ventana de logeo al sistema.....	110
Figura 42. Dashboard del sistema.....	111
Figura 43. Ventana listado de ventas.....	111
Figura 44. Realizar Venta.....	112
Figura 45. Búsqueda de medicamentos.....	112
Figura 46. Calculo automático de los montos de venta.....	113
Figura 47. Pago de venta	113
Figura 48. Factura de venta generada.....	114
Figura 49. Mantenimiento de productos	114
Figura 50. Configurar información de empresa	115
Figura 51. Ventana de Documentos Electrónicos.....	115

Figura 52. Arquitectura general de la solución.	119
Figura 53. Diagrama de los componente de arquitectura	120
Figura 54. Diagrama de componente de implementación	121
Figura 55. Resultado del porcentaje de calidad	122
Figura 56. Diagrama de base de datos.....	124
Figura 57. Diagrama de clases	145
Figura 58. Diagrama de despliegue de la aplicación	145

CAPÍTULO I

GENERALIDADES

1.1 Problemática

La presente investigación se realiza en la Botica “El gran poder” E.I.R.L, es una empresa privada ubicada en el distrito de Carabayllo, su negocio es brindar servicio hacia el público en general, referente a la venta de medicamentos.

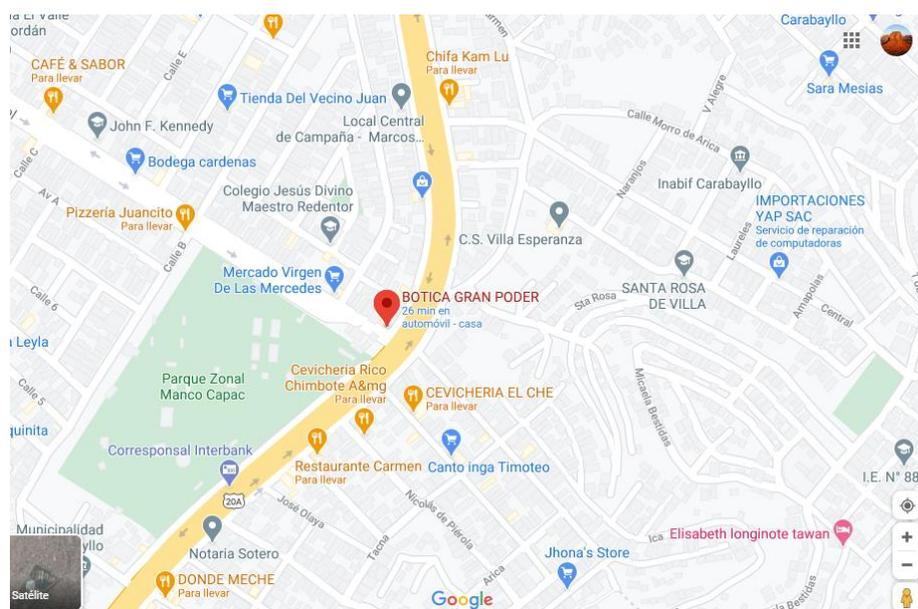


Figura 1. Ubicación del establecimiento comercial

Fuente: Google Maps

El proceso de ventas empieza cuando un cliente ingresa a la botica y se acerca a un vendedor, que lo recibe con un saludo e identifica las

necesidades del cliente, el 80% de los clientes consultan los medicamentos que requieren con su receta médica, con esta información el químico o técnico farmacéutico, revisa un Excel donde están registrados los medicamentos usando la función de búsqueda, de no encontrarse el medicamento, el vendedor sugiere un medicamento sustituto con igual formula, se verifica el stock que es controlado de forma manual y se informa del precio, todo este proceso se repite por cada medicamento hasta completar la venta, tomando un tiempo promedio en la atención de entre 6 a 12 minutos aproximadamente, el otro 20% de clientes realizan un pedido específico, sobre medicamentos que no requieren de una receta médica, de los cuales los vendedores tienen conocimiento, pero también están indicados en una columna del Excel, tomando un tiempo promedio de atención de 2 a 4 minutos.



Figura 2. Programa Excel Control de Botica Actual

Fuente: Boticas el Gran Poder

El problema principal o dolor de la empresa, empieza a surgir con el aumento de clientes en este último año, debido al contexto de la pandemia producido por el virus COVID19 aumentando la demanda de medicamentos, esta demanda no puede ser atendida con eficiencia, debido a que no cuentan con un sistema automatizado que permita agilizar este proceso, produciendo colas y demora para la atención, ocasionando molestias en los clientes, para agravar la situación algunos clientes solicitan factura, que se realiza de forma manual aumentando el tiempo de la atención, vale mencionar que ahora SUNAT exige la facturación electrónica, asimismo el control de sus medicamentos no es el adecuado, no se permite ver el stock en tiempo real y su disminución es ingresada por el vendedor, pudiendo ocasionar desfases en la cantidad y es más propenso a errores. El otro dolor de la empresa es en el cuadro de caja de venta diarias, como el proceso de ingresar o anotar la cantidad de las ventas es manual, hay ocasiones en que los montos vendidos no cuadran con exactitud.

Conociendo estos dolores presentados en el actual modelo de negocio se propone como solución, el desarrollo de un sistema web al cual nombraremos como “**lpharmaexpress**”, que permitirá gestionar el proceso de ventas, emitir tickets, boletas y facturas electrónicas, controlar el stock en tiempo real y control de ingresos y egresos de caja, con la finalidad de mejorar el proceso de venta.

1.1.1 Problema General

¿En qué medida del desarrollo del sistema web mejorara el proceso de venta de la Botica el Gran Poder, Carabayllo, Lima 2020?

1.1.2 Problema Especifico

- ¿En qué medida el estándar de, **adecuación funcional** permite mejorar el proceso de ventas de la Botica el Gran Poder Carabayllo, Lima 2020?
- ¿En qué medida el estándar de, **fiabilidad** permite mejorar el proceso de ventas de la Botica el Gran Poder Carabayllo, Lima 2020?
- ¿En qué medida el estándar de, **eficiencia de desempeño** permite mejorar el proceso de ventas de la Botica el Gran Poder Carabayllo, Lima 2020?
- ¿En qué medida el estándar de, **usabilidad** permite mejorar el proceso de ventas de la Botica el Gran Poder Carabayllo, Lima 2020?
- ¿En qué medida el estándar de, **mantenibilidad** permite mejorar el proceso de ventas de la Botica el Gran Poder Carabayllo, Lima 2020?

1.2 Objetivo general

Determinar la mejora del desarrollo del sistema web en el proceso de venta de la Botica el Gran Poder, Carabayllo, Lima 2020.

1.3 Objetivos específicos

- Determinar la mejora con **adecuación funcional** el proceso de ventas de la Botica el Gran Poder Carabayllo, Lima 2020.
- Determinar la mejora con **fiabilidad** el proceso de ventas de la Botica el Gran Poder Carabayllo, Lima 2020.
- Determinar la mejora con **eficiencia de desempeño** el proceso de ventas de la Botica el Gran Poder Carabayllo, Lima 2020.
- Determinar la mejora con **usabilidad** el proceso de ventas de la Botica el Gran Poder Carabayllo, Lima 2020.
- Determinar la mejora con **mantenibilidad** el proceso de ventas de la Botica el Gran Poder Carabayllo, Lima 2020.

1.4 Resultados esperados

Se obtuvo como resultado del desarrollo del sistema web “**lpharmaExpress**” los siguientes puntos:

- Mejorar significativamente el proceso de ventas de la Botica El Gran Poder, reduciendo el tiempo para la atención de los clientes.
- Mostrar el stock de las ventas en tiempo real, mediante alertas de fácil visibilidad para los usuarios finales.
- Emitir de forma automática tiques, boletas y facturas, que serán

comunicadas a SUNAT mediante su api de interconexión 2.1 y que usa la tecnología web servicios SOAP.

- Realizar un arqueo de caja optimo y eficiente, con las ventas registradas del día, luego se usaría esta información para reportaría y analítica de datos.
- La aplicación del estándar ISO/IEC 25000 SQuaRE, permitió medir el nivel de calidad de software del sistema “Ipharmaexpress” obtenido un puntaje de 8,31 sobre 10 lo que representa una puntuación aceptable.
- Mediante la aplicación de la Metodología RUP, se cumplió con las fases necesarias para el desarrollo de software, para la construcción del sistema web, obteniendo un producto de calidad.

1.5 Herramientas utilizadas

Las herramientas que se utilizaran para el desarrollo web del sistema “**Ipharmaexpress**” para la Botica el Gran Poder, Carabayllo, Lima 2020 son del tipo TIC. Las TIC son un conjunto de procesos y productos derivados de las nuevas herramientas (hardware y software), soportes y canales de comunicación, relacionados con el almacenamiento, procesamiento y la transmisión digitalizada de la información. Ochoa y Cordero (2002)

COMPONENTES	HERRAMIENTAS		CANTIDAD	MARCA/EMPRESA	LICENCIA
Tecnología de información y comunicación	Hardware	Laptop i7 3.1 - 16 GB RAM	1	TOSHIBA	No aplica
		Impresora BN/Color	1	BROTHER	No aplica
		Servidor Apache	1	Apache Foundation	Licencia Apache 2.0
	Software	Visual Studio Code	1	Microsoft Foundation	Free/Open Source
		XAMPP	1	Apache Friends	GNU General Public License
		Maria DB	1	MariaDB Corporation	GNU General Public License
		PHP	1	The PHP Group	Licencia PHP
		JavaScript	1	Fundación Mozilla	GNU/MIT
		Jquery	1	Fundación Mozilla	GNU/MIT
		Bootstrap	1	Twitter	MIT
		CSS3	1	W3C	Free/Open Source
		SaSS	1	Hampton Catlin	MIT
		HTML5	1	W3C	Free/Open Source
		GIT	1	Linux Foundation	GNU General Public License
		draw.io	1	Apache	Licencia Apache 2.0
		Bizagi Modeler	1	Bizagi Studio	Freemium Bizagi
		Notepad++	1	Don Ho	GNU General Public License
		Enterprise Architec	1	Sparx Systems	Sparx Systems
		Apache Server	1	Apache	Licencia Apache 2.0
		Codeigniter Framework	1	EllisLab	GNU/MIT
Microsoft Windows 10	1	Microsoft	License Microsoft Corporation		
Tecnología de redes y telecomunicaciones	Línea dedicada de internet CLARO (Mínimo 100 MB)				
Requerimiento y Evaluación de calidad de software	ISO/IEC 25000 SQuaRE	SONARQUBE	1	Manuel Pichler	BSD-3-Clause
		XDEBUG y WEBGRIND	1	The PHP Group	Licencia PHP
		Herramienta de Análisis Estático: Modelo SQuaRe	1	SonarSource	GNU General Public License

Tabla 1. Herramientas TIC utilizadas
Fuente: Elaboración propia

1.6 Métodos y procedimientos

1.6.1 Metodología de desarrollo del proyecto

La metodología de software que se usará para el desarrollo del sistema web “**lpharmarexpress**”, para la Botica el Gran Poder, estará basado en la metodología RUP, donde se aplicara las fases de desarrollo, como se detallan a continuación:

FASES DE EJECUCION	FLUJO DE TRABAJO	COMPONENTES
Inicio	Modelo del negocio	Plan de desarrollo del software
		Modelo de Proceso del negocio
		Modelo de Use Case del negocio
Elaboración	Requerimientos	Modelo de casos de uso del sistema
		Especificación de casos de uso
	Análisis y diseño	Prototipo Arquitectónico
		Diagrama de clases
		Diagrama de base de datos
		Diagrama de secuencia
Construcción	Implementación	Diagrama de componentes
		Diagrama de despliegue
		Prototipo del software Final
Transición	Prueba	Prueba por use case

Tabla 2. Fases de ejecución del proyecto

Fuente: Elaboración propia.

1.6.2 Metodología y procedimiento para la evaluación

El procedimiento para la etapa de evaluación se basará en el estándar ISO/IEC 25000 SQuaRe (Requerimiento y Evaluación de Calidad de Software), según esta familia de normas que brindan un marco de trabajo referencial en común, se encuentra compuesta por cinco divisiones que son las siguientes:

ISO/IEC 2500n: División para la gestión de la calidad.

ISO/IEC 2501n: División para el modelo de la calidad.

ISO/IEC 2502n: División para la medición de la calidad.

ISO/IEC 2503n: División para los requisitos de calidad.

ISO/IEC 2504n: División para la evaluación de calidad.

Dentro de este ISO, se especifica el modelo de calidad genérico que clasifica la calidad del producto se divide en características, que se pueden dividir en subcaracterísticas y atributos de calidad. Esta característica incluye dos partes:

- Modelos de calidad internos y externos de productos de software.
- Modelo de calidad de uso de productos de software:

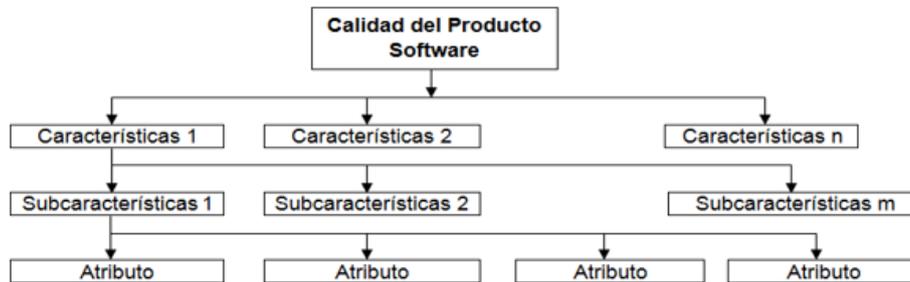


Figura 3. Estructura genérica usada por el modelo de calidad.
Fuente: Balseca, E.

El modelo de calidad para el producto de software interna y externa define ocho características de un producto de software, que de acuerdo con el modelo genérico (Véase Figura3.) están subdivididos en subcaracterísticas y cada una de estas en atributos, los cuales pueden aplicar a ser medidos con métricas internas o externas. A continuación, se muestran las características y subcaracterísticas del modelo de calidad de producto de software interna y externa en la siguiente figura:



Figura 4. Indicadores de calidad del ISO-SQUARE 25010
Fuente: <https://iso25000.com/index.php/normas-iso-25000/iso-25010>

Asimismo, el modelo para la calidad en uso define cinco características de un producto de software que de acuerdo con el modelo genérico (Véase Figura3.) están subdivididos en subcaracterísticas y cada una de estas en atributos, los cuales pueden aplicar a ser medidos con métricas de calidad en uso. A

continuación, se muestran en la siguiente figura las características y subcaracterísticas del modelo de calidad en uso:

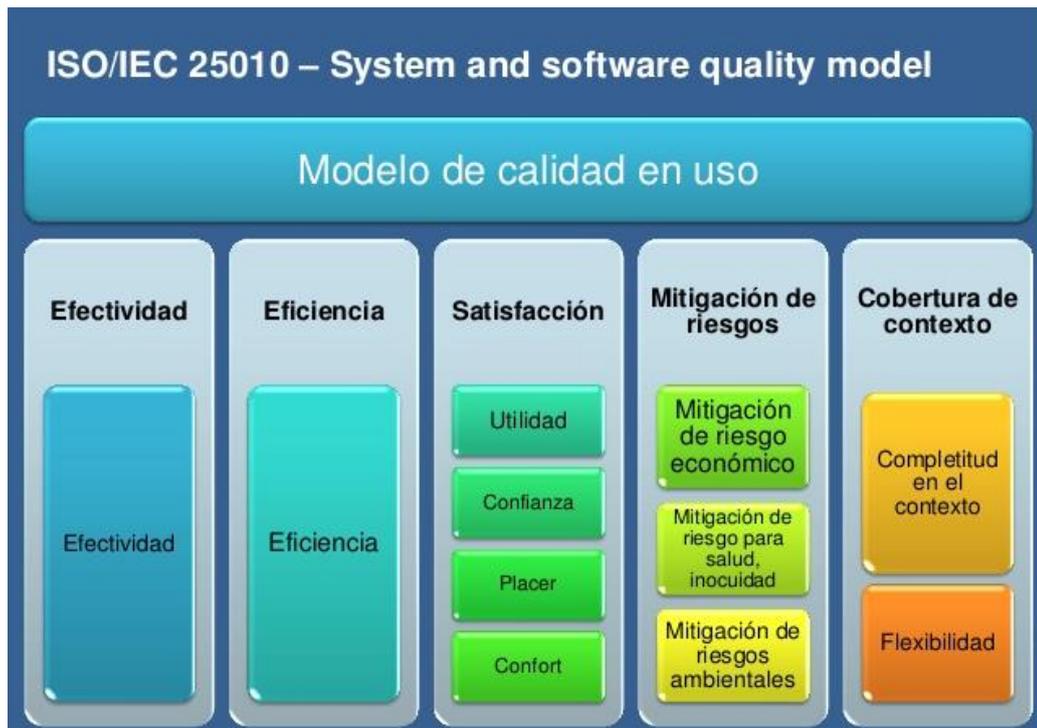


Figura 5. Indicadores de calidad en uso.

Fuente: Martínez, R.

Referente a la Calidad en Uso para este proyecto no se revisarán sus características, por el alcance de este, que se limita a la construcción del software.

Las normas ISO/IEC 25023 e ISO/IEC 25022 proporcionan un conjunto de indicadores de calidad internos, externos y en uso, que se utilizan junto con el modelo de calidad ISO/IEC 25010. En sus respectivos estándares, pueden modificarlos, e incluso utilizar métricas que no están definidas en el estándar, siempre que se especifique la relación entre el estándar métrico y el modelo de calidad ISO/IEC 25010 o el modelo de calidad que sustituirá al estándar. está especificado. Uno especificado en el estándar. El usuario debe seleccionar las características y subcaracterísticas de la calidad a evaluar, determinar los indicadores más apropiados y relevantes e interpretar los resultados de la medición de manera objetiva. Los usuarios pueden determinar la calidad de los productos de software de acuerdo con el proceso de evaluación de la calidad del producto definido en la norma ISO/IEC 2504n, que proporciona un método para evaluar la calidad de los productos o sistemas de software.

1.6.2.1 Selección de las características internas y externas de calidad a evaluar

Las características de calidad propuestas en la norma son aplicables a todos los sistemas software, pero dependiendo del tipo de sistema a evaluar, la importancia de estas características es mayor o menor, por lo que estas características de calidad se aplicarán a otros sistemas software. Los productos de software se definirán por tipo de producto, que en el caso del proyecto es un Sistema Web destinado a la venta de medicamentos que será desplegado en un ambiente Local. Para seleccionar las características a evaluar en el proyecto, se presenta el siguiente cuadro para medir el nivel de importancia y ponderación para la posterior interpretación de resultados:

Nivel de importancia	Simbología	Porcentaje referencial del nivel de la importancia	Significado
Alto	A	70% - 100%	La importancia de las características y subfunciones es muy alta, por lo que se medirán.
Medio	M	25% - 69%	Las características y subcaracterísticas no están relacionadas, pero pueden estar o No medible según los estándares del evaluador
Bajo	B	1% - 24%	Las funciones y subcaracterísticas no están relacionadas, ni estarán medidas.
No Aplica	NA	0%	Este valor dará características y subcaracterísticas que no se pueden medir en función de diferentes factores.

Tabla 3. Porcentaje referencial al nivel de importancia

Fuente: Balseca (2014).

De acuerdo con lo definido en el cuadro anterior se procederá a darle los pesos necesarios a las características a ser evaluadas tanto internas como externas:

CARACTERÍSTICAS DE CALIDAD INTERNA		
Características	Nivel de importancia	Motivo de selección
Adecuación funcional	M	Tiene un valor de importancia M porque es necesario evaluar si la función de codificación cumple los requisitos especificados para el sistema.
Fiabilidad	M	Tiene un valor de importancia M se usa para la calificación porque es necesario evaluar si la función de codificación cumple con los requisitos del sistema cuando están sujetos a ciertas condiciones y un cierto período de tiempo
Eficiencia en el desempeño	M	Tiene valor de importancia M se utiliza para la clasificación porque a nivel de código, los recursos que se utilizarán deben considerarse para evaluar el rendimiento del sistema.
Usabilidad	M	Tiene un valor de importancia M porque es necesario evaluar que cualquier programador puede comprender, aprender y utilizar el código.
Seguridad	B	Se califica con valor de importancia B porque el requerimiento de negocio indica que el producto funcionara en ambiente de red local.
Compatibilidad	B	Se califica con valor de importancia B porque el requerimiento de negocio indica que el producto solo debe funcionar con los navegadores actuales.
Mantenibilidad	A	La razón para usar la clasificación del valor de importancia A es porque es muy necesario que cualquier programador modifique o actualice el código de acuerdo con los requisitos de corrección.
Portabilidad	NA	Dado que no es adecuado para evaluar productos tipo web, se evalúa con un valor importante NA .

Tabla 4. Características de calidad interna seleccionadas.

Fuente: Balseca (2014).

CARACTERÍSTICAS DE CALIDAD EXTERNA		
Características	Nivel de importancia	Motivo de selección
Adecuación funcional	A	La razón por la que se usa el valor de importancia A porque es muy necesario evaluar que el sistema presente todas las funcionalidades especificadas para su uso.
Fiabilidad	M	La razón por la que se usa el valor de importancia M porque es necesario evaluar que el sistema realice todas las funciones especificadas cuando es usado bajo ciertas condiciones y periodos de tiempos.
Eficiencia en el desempeño	A	La razón por la que se usa el valor de importancia A para la calificación es porque es necesario considerar los recursos que se usarán para evaluar el desempeño del sistema.
Usabilidad	M	La razón por la que se usa el valor de importancia M , porque es necesario evaluar la inteligibilidad, comodidad y facilidad de uso del sistema
Seguridad	B	Se califica con valor de importancia B porque el requerimiento de negocio indica que el producto funcionara en ambiente de red local.
Compatibilidad	B	Se califica con valor de importancia B porque el requerimiento de negocio indica que el producto solo debe funcionar con los navegadores actuales.
Mantenibilidad	M	La razón por la que se usa el valor de importancia M para la calificación es porque es necesario evaluar si el sistema está funcionando normalmente para el usuario después de la actualización o modificación.
Portabilidad	NA	Dado que no es adecuado para evaluar productos tipo web, se evalúa con un valor importante NA .

Tabla 5. Características de calidad externa seleccionadas.

Fuente: Balseca (2014).

1.6.2.2 Selección de la subcaracterísticas internas y externas de calidad a evaluar

De acuerdo con lo definido en el cuadro anterior se procederá a darle los pesos necesarios a las subcaracterísticas a ser evaluadas tanto internas como externas:

SUBCARACTERÍSTICAS Y ATRIBUTOS DE CALIDAD INTERNA			
Características	Subcaracterísticas	Nivel de importancia	Motivo de selección
Adecuación funcional	Compleitud funcional	A	El valor de importancia A se utiliza para la clasificación porque es muy necesario evaluar si la función de codificación cubre todas las tareas determinadas por el usuario.
	Exactitud funcional	B	Dado que no es necesario evaluar la precisión funcional a nivel de código, se puntúa utilizando el valor de importancia B , por lo que se evaluará desde calidad externa.
Fiabilidad	Madurez	B	El valor de importancia B se utiliza para la calificación porque no es necesario evaluar la madurez a nivel de código, por lo que se evaluará desde calidad externa.
	Disponibilidad	B	El valor de importancia B se utiliza para la calificación porque no es necesario evaluar la usabilidad a nivel de código, por lo que se evaluará desde una perspectiva externa.
	Tolerancia a fallos	M	El valor de importancia M se usa para la calificación porque es necesario evaluar si ciertas funciones de codificación permiten que el sistema funcione en caso de falla.
	Recuperabilidad	B	Dado que no hay necesidad de evaluación, su nivel de importancia es B .
Eficiencia en el desempeño	Comportamiento Temporal	B	Como no es necesario evaluar la madurez a nivel de código, se evalúa utilizando el valor de importancia B , por lo que se evaluará desde calidad externa.
	Utilización de recursos	M	Dado que debe evaluarse que la cantidad necesaria de código se utiliza correctamente, se puntúa utilizando el valor de importancia M .
	Capacidad	B	Dado que no hay necesidad de evaluación, su nivel de importancia es B .
	Capacidad de reconocer su adecuación	M	La razón por la que se usa el valor de importancia M para la clasificación es porque es necesario evaluar si algún programador puede comprender la función de codificación.

Facilidad de uso	Capacidad de ser entendido	M	Está limitado por el valor de importancia M , porque es necesario evaluar si hay funciones en el código que son obvias para el programador.
	Operatividad	M	La razón por la que se usa el valor importante M para la calificación es porque es necesario evaluar si hay una función para que el sistema pueda operarse fácilmente.
	Protección frente a errores de usuarios	M	Está limitado por el valor de importancia M , porque es necesario evaluar si existen elementos de entrada que hayan sido validados en el código fuente.
	Estética de la interfaz de usuario	B	Dado que no es necesario evaluar la estética de la interfaz de usuario a nivel de código, se califica utilizando el valor de importancia B , por lo que se evaluará desde una perspectiva externa.
	Accesibilidad técnica	B	Dado que no hay necesidad de evaluación, su nivel de importancia es B .
	Autenticidad	A	Utiliza el valor de importancia A para puntuar, porque es muy necesario evaluar si existe un método de autenticación en el código fuente.
Mantenibilidad	Modularidad	A	Está limitado por el valor de importancia A , porque es necesario evaluar si tiene un impacto en otras funciones cuando se modifica el código.
	Reusabilidad	A	El valor de importancia A se utiliza para la identificación, porque es muy necesario evaluar si el código se puede reutilizar.
	Capacidad de ser analizado	M	La razón por la cual se utiliza el valor de importancia M para la calificación es porque es necesario evaluar si existe un registro de fallas que ocurrieron en respuesta a su análisis correspondiente.
	Capacidad de modificación	A	Está limitado por el valor de importancia A , porque es muy necesario evaluar si es posible modificar el código sin afectar la función del sistema.
	Capacidad de ser probado	B	Dado que no hay necesidad de evaluación, su nivel de importancia es B .

Tabla 6. Subcaracterísticas de calidad seleccionadas.

Fuente Balseca (2014).

SUBCARACTERÍSTICAS Y ATRIBUTOS DE CALIDAD EXTERNA			
Características	Subcaracterísticas	Nivel de importancia	Motivo de selección
Adecuación funcional	Compleitud funcional	A	Se Utiliza el valor de importancia A para clasificarlo, porque es necesario evaluar si el sistema proporciona todas las funciones necesarias para el usuario.
	Exactitud funcional	A	La importancia de este nivel es A , porque es necesario evaluar si el sistema proporciona el resultado correcto.
Fiabilidad	Madurez	A	La importancia de este nivel es A , porque es necesario evaluar si el sistema proporciona los resultados correctos
	Disponibilidad	M	El valor de importancia M se utiliza para la calificación porque es necesario evaluar si el sistema es operativo y utilizable.
	Tolerancia a fallos	M	El valor de importancia M se usa para la calificación porque es necesario evaluar si el sistema puede operar cuando ocurre una falla.
	Recuperabilidad	B	Dado que no necesita ser evaluado, se califica con el valor de importancia B .
Eficiencia en el desempeño	Comportamiento Temporal	M	Su puntuación es el valor de importancia M , porque es necesario evaluar si el sistema proporciona un tiempo de respuesta adecuado.
	Utilización de recursos	M	Su puntuación es el valor de importancia M , porque es necesario evaluar si el sistema utiliza los recursos adecuados en tiempo de ejecución.
	Capacidad	B	Dado que no hay necesidad de evaluación, se designa como el valor de importancia B .
Facilidad de uso	Capacidad de reconocer su adecuación	M	La razón por la que se usa el valor de importancia M para la clasificación es porque es necesario evaluar si algún programador puede comprender la función de codificación.
	Capacidad de ser entendido	A	El valor de importancia A se usa para calificar porque es muy necesario evaluar si el usuario puede entender el sistema cuando se usa.
	Operatividad	M	La razón por la que se usa el valor de importancia M para la calificación, porque es necesario evaluar si el usuario puede operar fácilmente el sistema.

	Protección frente a errores de usuarios	B	Su nivel de importancia es B , porque no necesita ser evaluado desde el exterior, por lo que se evaluará desde una perspectiva interna.
	Estética de la interfaz de usuario	M	La razón por la que se usa el valor de importancia M para la calificación es porque es necesario evaluar si la interfaz del sistema satisface y agrada a los usuarios.
	Accesibilidad técnica	B	Dado que no necesita ser evaluado, se califica con el valor de importancia B .
Mantenibilidad	Modularidad	NA	Su valor de importancia es NA porque no se aplica al exterior, pero se aplica al nivel de código.
	Reusabilidad	NA	Su valor de importancia es NA porque no se aplica al exterior, pero se aplica al nivel de código.
	Capacidad de ser analizado	M	La razón por la que se usa el valor de importancia M para la calificación es porque es necesario evaluar si el análisis de impacto después de la modificación es factible.
	Capacidad de modificación	M	Su valor de importancia es M , porque es necesario evaluar si el sistema se puede modificar sin afectar la función del sistema.
	Capacidad de ser probado	B	Dado que no hay necesidad de evaluación, su nivel de importancia es B .

Tabla 7. Subcaracterísticas de calidad externa seleccionados.

Fuente: Balseca (2014).

1.6.2.3 Ponderación de las características internas y externas de calidad a evaluar

De acuerdo con el análisis realizado en las tablas 4 y 5. Se tiene a modo de resumen las características de software interno y externo seleccionadas de acuerdo con el modelo de calidad de software del ISO 25010, cada uno con su respectivo porcentaje de ponderación que representaran en su totalidad el 100% que se requiere evaluar, en los siguientes cuadros:

CARACTERÍSTICAS DE CALIDAD INTERNA			
Características	Nivel de importancia	Ponderación	Motivo de ponderación
Adecuación funcional	M	15%	Su valor de peso es del 25%, porque es necesario evaluar si la función de codificación cumple con los requisitos especificados por el sistema.
Fiabilidad	M	10%	Su valor de peso es del 10% es porque es necesario evaluar si la función de codificación cumple con los requisitos de la regulación del sistema dentro de ciertas condiciones y un cierto período de tiempo.
Eficiencia en el desempeño	M	20%	Su valor de peso es del 25%, porque a nivel de código, se deben considerar los recursos a utilizar para evaluar el desempeño del sistema.
Usabilidad	M	20%	Su valor de peso es del 25% porque es necesario evaluar que cualquier programador puede comprender, aprender y utilizar el código.
Mantenibilidad	A	35%	Su valor de peso es del 25%, porque cualquier programador debe modificar o actualizar el código según la necesidad de corrección.

Tabla 8. Características de calidad interna seleccionadas ponderadas.

Fuente: Balseca (2014).

CARACTERÍSTICAS DE CALIDAD EXTERNA			
Características	Nivel de importancia	Ponderación	Motivo de ponderación
Adecuación funcional	A	30%	Su valor de peso es del 20%, porque es muy necesario evaluar si el sistema tiene todas las funciones especificadas para su uso.
Fiabilidad	M	15%	Su valor de peso es del 15%, porque es necesario evaluar si el sistema realiza todas las funciones especificadas cuando se utiliza el sistema en determinadas condiciones y períodos de tiempo.
Eficiencia en el desempeño	M	20%	Su valor de peso es del 30%, porque es necesario considerar los recursos a utilizar para evaluar el desempeño del sistema.
Facilidad de uso	M	15%	Su valor de peso es del 15% porque es necesario evaluar la comprensibilidad, comodidad y facilidad de uso del sistema.
Mantenibilidad	M	20%	Su valor de peso es del 20%, porque es necesario evaluar si el sistema está funcionando con normalidad para el usuario después de la actualización o modificación.

Tabla 9. Características de calidad externa seleccionadas ponderadas.

Fuente: Balseca (2014).

1.6.3 Procedimiento para la interpretación de los resultados de la evaluación de calidad.

Para interpretar los resultados de la evaluación se establece la escala de aceptación de calidad que deben cumplir las todas las subcaracterísticas por característica de calidad, que se presenta en el siguiente cuadro:

Escala de medición	Niveles de puntuación	Grado de satisfacción
8.75 – 10	Cumple con los requisitos	Muy satisfactorio
5 – 8.74	Aceptable	Satisfactorio
2.75 – 4.9	Mínimamente aceptable	Insatisfactorio
0 – 2.74	Inaceptable	

Tabla 10. Niveles de puntuación aplicados a la medición de la calidad de software.

Fuente: Vivanco, A. Elaboración propia.

Para las fórmulas de las métricas nos apoyaremos de los cuadros donde

se especifica la fórmula que se debe usar para calcular cada atributo de subcaracterística (Ver Anexo 2.1), asimismo las matrices donde se realizarán los cálculos de ponderación de los atributos evaluados de calidad interno como externo (Ver Anexo 2.2), tendrán las siguientes columnas, que se detallan a continuación:

Columna	Descripción
Característica	Nombre de la característica de calidad.
Subcaracterística	Nombre de la subcaracterística de calidad.
Métrica	Atributo de calidad a ser medido.
Formula	Formula que calcula el cumplimiento de la métrica.
Valor deseado/Umbral	Valor que se desea obtener al calcular la métrica con la formula.
Valor obtenido	Valor que se obtiene luego de aplicar la formula.
Ponderación	Valor que se pondera en base a 10. (Ver tabla 10).
Valor Parcial Total	Valor que se pondera en base a todos los atributos medidos.
Nivel de Importancia	Nivel de importancia definido para la subcaracterística
Porcentaje de Importancia	Porcentaje asignado en base al nivel de importancia (Ver tabla 3)
Valor Final	Valor final que se asigna a la característica.
Calidad del Sistema	Valor final que se asigna a la calidad interna o externa.

Tabla 11. Descripción de columnas de las matrices de calidad.

Fuente: Balseca (2014).

1.7 Alcance

El proyecto de tesis tiene como alcance el desarrollo de un sistema web que será desplegado en entorno local y que permitirá automatizar y mejorar el proceso de ventas de la Botica El Gran Poder, con el fin de solucionar los distintos dolores que tiene la empresa en dicho proceso, asimismo el sistema tendrá el módulo de ventas, documentos electrónicos, mantenedores, administrador, reportes y notificaciones, el desarrollo se gestionara teniendo en cuenta el marco metodológico de RUP, la calidad del software será medida con el ISO/IEC 25000 SQuaRE.

1.8 Limitaciones

1.8.1 Limitación espacial

El proyecto de tesis estará limitada a la unidad de estudio que es la Botica el Gran Poder EIRL.

1.8.2 Limitación temporal

El presente proyecto se realizará durante el año 2020, tomando en cuenta la información proporcionada por el usuario en referencia al proceso de ventas.

1.8.3 Limitación teórica

Para el presente proyecto de tesis se está considerando los Lineamientos Metodológicos de desarrollo de software RUP, la normativa del ISO/IEC 25000 SQuaRE.

1.9 Riesgos

Para el proyecto se establecieron los riesgos potenciales para su realización en función a su nivel de consecuencia, en el área de recursos humanos:

PROBLEMA	ITEM	ASPECTOS	NIVELES DE PROBABILIDAD	NIVEL DE CONSECUENCIA	CONTINGENCIA
Pérdida de información de la base de datos.	Computadoras	1	Posible	Menor	Uso IaaS
	Laptops	1	Posible	Menor	IaaS
	Otros Equipos	1	Posible	Menor	IaaS
	Coordinador	1	Improbable	Menor	-
	Técnico	1	Improbable	Mayor	-
	Tiempo	2 meses	Improbable	Menor	-
Errores de interconexión con SUNAT, pérdida de prestigio	Computadoras	1	Posible	Menor	IaaS
	Laptops	1	Posible	Menor	IaaS
	Otros Equipos	1	Posible	Menor	IaaS
	Coordinador	1	Improbable	Menor	-
	Técnico	1	Improbable	Mayor	-
	Tiempo	2 meses	Posible	Menor	Revisión técnica

Tabla 12. Riesgos del proyecto

Fuente: Elaboración propia.

1.10 Justificación y viabilidad del proyecto

1.10.1 Justificación

1.10.1.1 Justificación Teórica

La ejecución del análisis del presente proyecto de tesis, admitirá definir y teorizar la metodología para el desarrollo del sistema, en este caso la Metodología RUP, con la cual se implementará un software de forma metódica, de acuerdo a los estándares de calidad de la ISO/IEC 25000 SQuaRE y las necesidades del negocio de la empresa Botica El Gran Poder,

para tal efecto, se ejecutarán las siguientes fases: Modelado de Negocio, Requisitos, Análisis y Diseño, Implementación y Pruebas del Sistema web de ventas “Ipharmaexpress”.

1.10.1.2 Justificación Práctica

El proyecto de tesis admitirá evaluar y medir que el software “Ipharmaexpress” cumpla con los indicadores de calidad de la ISO/IEC 25000 SQuaRE y cubra la necesidad del negocio Botica el Gran Poder. Lo que permitirá automatizar y mejorar el proceso de venta.

1.10.1.3 Justificación Metodológica

El presente proyecto sirve como base para investigaciones y como modelo replicable para futuras investigaciones enfocadas en la misma o similares problemáticas teniendo en cuenta que cada realidad es distinta, pero estos procesos se inician bajo condiciones similares. Asimismo, permitirán explicar la validez de la Metodología RUP en el proceso de desarrollo del sistema web “Ipharmaexpress”, demostrando su confiabilidad y validez para ser usado en proyectos similares.

1.10.2 Viabilidad del proyecto

1.10.2.1 Viabilidad técnica

El proyecto cumple con los lineamientos establecidos en el Reglamento de Titulaciones y Titulaciones de la Universidad Privada de Pucallpa y se refleja en el alcance investigativo de la profesión de ingeniería de sistemas. En este sentido, para el sistema se aplicará la norma ISO / IEC 25000 SQuaRE, que permitirá evaluar la calidad del software en base a las siguientes características seleccionadas del modelo de calidad de software departamental ISO / IEC 25010: **Eficiencia de Desempeño, Usabilidad y Mantenibilidad** del sistema de acuerdo con los criterios establecidos en los requerimientos funcionales y no funcionales del sistema “**Ipharmaexpress**”

1.10.2.2 Viabilidad financiera

El proyecto de tesis del Sistema Web “Ipharmaexpress”, será financiado en su totalidad por la empresa DSG PERU TECH EIRL, dedica a la producción de software para diferentes rubros de negocio con sede en la ciudad de Pucallpa.

CAPÍTULO II: MARCO DE REFERENCIA

2.1 Marco conceptual

2.1.1 Antecedentes del Problema

2.1.1.1 Antecedentes Internacionales

Según Arana (2014) en la tesis titulada *Desarrollo e implementación de un sistema de gestión de venta de repuestos automotrices en el almacén de auto repuestos eléctricos marcos en la parroquia posorja cantón guayaquil, provincia de las guayas*, “pregrado” de la Universidad ESTATAL PENÍNSULA DE SANTA ELENA - Ecuador. La ejecución del presente trabajo investigativo tiene como objetivo primordial automatizar y sistematizar, mediante una aplicación informática, todo el proceso de compra, venta y control de inventarios del Almacén de ventas “Auto Repuestos Eléctricos Marcos” ubicado en la Parroquia Posorja, Barrio 20 de Diciembre, para tal objetivo la investigación se fundamentó en la información general del establecimiento comercial, utilizando un lenguaje de programación en software libre como tendencia mundial en elaboración de sistemas para el sector público y privado. La metodología utilizada en el diseño del sistema se desarrolló en la estructura de red de cliente servidor, utilizando formularios HTML y lenguaje de programación PHP y Apache, la realización del sistema constituye un hecho de gran importancia y trascendencia tanto para los usuarios del software desarrollado en función de las facilidades que

generará el uso de este, así como también para la autora de este en función de la grandiosa experiencia, que enriqueció profundamente su formación profesional académica estudiantil. Las **conclusiones** son:

- A través del proceso de gestión de ventas automatizado, obtener volumen de ventas diario, actualización completa de inventario, registro, superar el proceso manual que hacía que los clientes perdieran tiempo en el proceso de facturación. Devolución de mercancías, mercancías que ingresan al almacén, facturas, registro de proveedores, listas de clientes, para brindar a los clientes del almacén servicios rápidos, verdaderos y eficientes.
- El sistema desarrollado brinda una gran ayuda para una mejor gestión de las organizaciones comerciales, especialmente en lo que se refiere a la gestión de inventarios, registro, clasificación y acceso a la información de forma ágil, rápida y real de las actividades comerciales.

Según Pacuanca, F. (2014) en su tesis titulada *Sistema web de venta de boletos para las empresas de transporte de la terminal de buses la paz, caso: soluciones de tecnología en ingeniería de sistemas y telecomunicaciones*, “pregrado” de la Universidad MAYOR DE SAN ANDRÉS - Bolivia. El presente proyecto tiene como principal objetivo desarrollar un sistema web para mejorar el servicio de venta de boletos para las empresas descritas anteriormente, de forma que los procesos sean más eficientes y mejores. Cabe mencionar que el presente trabajo se desarrolló utilizando la metodología ágil XP (Programación Extrema) basándose en sus fases, complementando la fase de diseño con WebML (Lenguaje del Modelado Web) siendo de gran ayuda al momento de diseñar los diferentes procesos y las interfaces de usuario mediante el uso de esquemas. Se

empleó Web-Site QEM (Método de Evaluación de Calidad) para evaluar y medir la calidad del producto que se basa en las normas de la ISO 9126 tomando en cuenta: funcionalidad, confiabilidad, usabilidad, eficiencia y mantenibilidad, que proporcionan métricas para medir la calidad del producto final. Finalmente, los objetivos planteados para el proyecto se alcanzaron de manera que se produjo un producto de calidad, que en su desempeño cumple con los requerimientos del cliente. Las **conclusiones** son:

- Se puede mejorar el registro del personal de la empresa, y tenemos los datos más relevantes sobre el nuevo personal que ingresa a la empresa.
- Se facilitó el proceso de venta de los boletos para los pasajeros, disminuyendo el tiempo y dando comodidad de realizarlo desde cualquier lugar mediante el sitio web.
- Se perfeccionó la manera de pagar por parte de los usuarios, dando la posibilidad de hacerlo con una tarjeta de débito.
- Se pudo consolidar la información sobre los viajes que dispone la empresa, los diferentes buses que el pasajero puede elegir con su respectivo asiento.
- Se logró reducir el tiempo de adquisición de los boletos para los usuarios, por la generación rápida en el sistema.
- Se pudo mejorar el registro de los buses con sus respectivas características más relevantes, para mejorar la asignación de rutas.

- Para que los usuarios ingresen al sistema con un determinado privilegio se implementó la asignación de roles y la autenticación del usuario mediante un nombre de usuario más se contraseña que se almacenaron al momento de su registro.

2.1.1.2 Antecedentes Nacionales.

Según Bendezú, C (2017). En la tesis titulada: *Sistema web para el proceso de ventas en La Botica Helifarma E.I.R.L*, Lima, “pregrado” de la Universidad CESAR VALLEJO – Lima. La tesis que se presenta describe y detalla la implementación de un Sistema Web para el proceso de ventas en la botica “Helifarma” E.I.R.L. La presente tesis se encuentra desarrollada por la metodología RUP, inclusive se utilizó el tipo de estudio aplicada y experimental con diseño preexperimental, ya que esta metodología facilita la reutilización de código, además brinda una forma disciplinada de asignar tareas y responsabilidades en la empresa, reduce riesgos del proyecto e integra desarrollo con mantenimiento. Las **conclusiones** son:

- La conclusión es que el sistema Web mejora el proceso de venta del farmacéutico "Helifarma" porque puede incrementar el porcentaje de incremento en las ventas y la productividad de las ventas, logrando así el objetivo de este estudio.
- La conclusión es que el sistema Web ha incrementado el porcentaje de crecimiento de las ventas en un 10,43%. Por tanto, se puede decir que el sistema Web ha incrementado el porcentaje de ventas durante el proceso de venta.
- La conclusión es que el sistema Web ha aumentado la productividad de las ventas en un 19,31%. Por tanto, se

puede decir que el sistema Web mejora la eficiencia comercial en el proceso de venta.

Según Gamboa J, Guillermo R (2017). En la tesis titulada: *Implementación de un sistema web para las ventas en la empresa one to one contact solutions*, “pre pregado” de la Universidad SAN IGNACIO DEL OYOLA - Lima. El presente proyecto aborda la implementación de un sistema web como respuesta a una necesidad de la empresa One To One Contact Solutions en el área de ventas, la cual está posicionada entre los mejores canales de ventas asociados a Scotiabank Perú. Haciendo uso de la Metodología Ágil (Scrum y XP), se pretende desarrollar un aplicativo web basado en lenguaje de programación PHP, con un motor de base de datos MYSQL y servidor Web Apache 2.2. Las **conclusiones** son:

- La metodología Agile SCRUM y XP se ha dado cuenta de que el proyecto se completó en el menor tiempo posible y cumplió con las necesidades de la empresa.
- La sinergia entre SCRUM y XP permite el desarrollo de un proyecto bien diseñado, apoyado en las herramientas proporcionadas por todos.
- El sistema web se ha enfocado en mantener en contacto a los consultores de ventas, es decir, no hay tiempo de inactividad, porque ahora se ha aplicado el modo de marcación, y el resultado se muestra en el resultado que el número promedio de llamadas por cliente es de unas 10 llamadas lo que aumenta el alcance de Contractibilidad y la posibilidad de posterior finalización de transacciones.
- Se espera que el sistema web genere beneficios los cuales se den a corto plazo con el aumento de ventas y el uso adecuado del saldo para llamadas y a largo plazo en contratos con más clientes (entidades financieras y bancarias).

- Se espera que el proyecto sirva como guía para futuros desarrollos en la empresa.
- El proyecto tomo alrededor de 1 mes y medio desde la concepción del problema hasta el resultado y conclusiones.

Según Valles, S. (2015). En la tesis titulada: *Diseño e implementación de un sistema de venta para la distribuidora Josymar trujillo*, “pregrado” de la Universidad UNIVERSIDAD CATÓLICA LOS ÁNGELES DE CHIMBOTE – Chimbote. La investigación tuvo un diseño no experimental y fue de tipo descriptiva y de corte transversal. La población fue delimitada en 20 trabajadores y la muestra fue seleccionada en su totalidad de su población; con lo que una vez que se aplicó el instrumento se obtuvieron los siguientes resultados: En lo que respecta a la dimensión: Si la empresa contaba con un sistema de ventas, se puede observar que el 100% de los trabajadores encuestados expresó que NO contaban con ningún sistema que permita llevar el control de sus ventas. En cuanto a la dimensión: Nivel de satisfacción con respecto al inventario de los productos del negocio, se puede observar que el 85% de los trabajadores encuestados expresaron que NO están satisfechos con el actual inventario. En lo que respecta a la dimensión: Que debería hacer la distribuidora para mejorar sus servicios; se interpreta que el 100% de los trabajadores encuestados determinaron que SI es necesario tener un sistema que permita mejorar sus servicios. Estos resultados coinciden con las hipótesis específicas y en consecuencia con la hipótesis general; por lo que estas hipótesis quedan demostradas y la investigación queda justificada en la necesidad de implementar un sistema de venta para la distribuidora Josymar investigada. Las **conclusiones** son:

- El alcance de este estudio es describir el proceso comercial de implementación del sistema de ventas DISTRIBUIDORA JOSYMAR Trujillo. 2015. Considere el proceso en estudio: determine una solución de automatización, adquiera y mantenga software de aplicación para facilitar la operación y el uso. Luego de procesar los resultados, se determinó que el mayor problema al que se enfrentaba la distribuidora Josymar era que no contaba con un sistema de ventas y no podía sistematizar todas sus actividades comerciales.
- Al comparar los resultados con Rivas M, el antecesor de Mendoza F (Mendoza F) en la Universidad Estatal de Milagro en Ecuador, logró el 80% de los requisitos en trabajos de investigación similares y necesita implementar un sistema de ventas que permita mejorar el inventario. El sistema se puede automatizar, brindar mejores servicios a los clientes, mejorar los procesos de venta y obtener mejores ingresos económicos. Esta demanda es consistente con los resultados de nuestra investigación, es decir, el 85% del inventario puede mejorar el inventario, para lo cual tenemos La razón para investigar es basado en el precedente. Al comparar los resultados con los antecesores de Johanna R de la Universidad Católica del Perú, descubrió problemas en su trabajo de investigación y logró implementar un sistema de información para la tienda de ropa, que logró controlar efectivamente los productos que vendo para brindar servicios a los clientes. .Mejora del control de almacén, proceso de ventas y control de inventario, que son consistentes con nuestras similitudes y supuestos de investigación, por lo que es razonable implementar el sistema en la empresa encuestada.

2.1.2 Bases teóricas

Para la ejecución del presente proyecto de tesis, se ostentan las bases teóricas que permitirá facultar el sustento idóneo para el Desarrollo de un sistema web para el proceso de ventas en la botica “El gran poder”, Carabaylo, Lima 2020.

2.1.2.1 Programación Orientada a Objetos

Para el desarrollo del software “Ipharmaexpress” se utilizará el paradigma de programación orientada a objetos (POO), lo que nos permite reutilizar el código fuente y definir los patrones de diseño idóneos para la implementación, según Moreno (2015) define lo siguiente:

La programación orientada a objetos es un paradigma de programación totalmente diferente al método clásico de programación, el cual utiliza objetos y su comportamiento para resolver problemas y generar programas y aplicaciones informáticas. Con la programación orientada a objetos (POO) se aumenta el modularidad de los programas y la reutilización de estos. Además, la POO se diferencia de la programación clásica porque utiliza técnicas nuevas como el polimorfismo, el encapsulamiento, la herencia, etc.

Generalmente, los lenguajes de última generación permiten la programación orientada a objetos, así como la programación clásica. Por esta razón, puede entenderse la POO como una evolución de la programación clásica (programación estructurada).



Figura 6. Ciclo de desarrollo bajo el paradigma POO

Fuente: Programación Orientada a Objetos – Moreno (2015)

2.1.2.2 PHP (Hypertext Preprocessor)

Para el desarrollo del software “Ipharmaexpress”, se utilizará el lenguaje de programación PHP, Aguirre [2020] define al lenguaje de la siguiente manera: PHP es un lenguaje de programación de uso general que se adapta especialmente al desarrollo web. Fue creado inicialmente por el programador danés-canadiense Rasmus Lerdorf en 1994. En la actualidad, la implementación de referencia de PHP es producida por The PHP Group. PHP originalmente significaba Personal Home Page (Página personal), pero ahora significa el inicialismo recursivo PHP: Hypertext Preprocessor. PHP es un lenguaje pensado para estar orientado a objetos y aunque esta no es la única forma en la que se puede programar desde el, es muy recomendable hacerlo desde este paradigma, porque reduce considerablemente el código y, entre otras ventajas, vuelve mucho más seguras las aplicaciones. Dijimos que para crear objetos es preciso conocer sus atributos y métodos y declararlos dentro de clases, las cuales son moldes para la creación de dichos objetos. Veamos un ejemplo para que te vayas acostumbrando a este paradigma. Como hay que crear objetos de la vida real, u objetos que existan, piensa en alguno simple, como un vehículo (aunque puedes

moldear cualquier cosa que imagines). Para comenzar necesitas una clase, que se declara con la palabra reservada class, seguida del nombre de la clase. Todo lo que declares entre las llaves de apertura y cierra será parte de la clase. Comienza declarando algunos atributos. Estos se guardan en variables, que se declaren en la clase, pero no se las inicializa ya que una clase no es un objeto, sino un molde para objetos, y si quieres usar una clase para crear varios objetos, cada uno será único y tendrá características similares, pero no iguales. Los nombres de los atributos deben ser descriptivos y que tengan sentido.



Figura 7. Arquitectura web en PHP

Fuente: Slideplayer

2.1.2.3 Modelo Vista Controlador (MVC)

Para el desarrollo del software “Ipharmaexpress”, se utilizará el patrón de diseño Modelo-Vista-Controlador (MVC), Según Diaz y Fernández [2012] definen lo siguiente: El patrón MVC surge con el objetivo de reducir el esfuerzo de programación, necesario en la implementación de sistemas múltiples y sincronizados de los mismos datos, a partir de estandarizar el diseño de las aplicaciones.

El patrón MVC es un paradigma que divide las partes que conforman una aplicación en el Modelo, las Vistas y los Controladores, permitiendo la implementación por separado de cada elemento, garantizando así la actualización y mantenimiento

del software de forma sencilla y en un reducido espacio de tiempo. A partir del uso de frameworks basados en el patrón MVC se puede lograr una mejor organización del trabajo y mayor especialización de los desarrolladores y diseñadores.

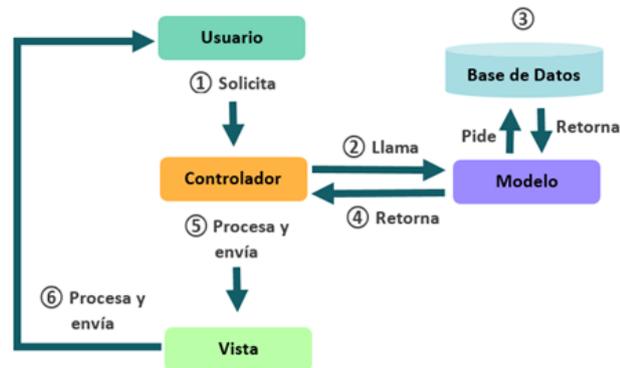


Figura 8. Ciclo de vida de MVC

Fuente: Gómez R.

2.1.2.4 Codeigniter Framework

Para el desarrollo del software “Ipharmaexpress”, se escribirá el código fuente con el framework Codeigniter el cual se base en el lenguaje de programación PHP e implementa el modelo MVC, teniendo una curva de aprendizaje corta en comparación de su principal competidor Laravel, la documentación oficial de Codeigniter define lo siguiente:

Codeigniter es un marco de desarrollo de aplicaciones, un conjunto de herramientas, para personas que crean sitios web utilizando PHP. Su objetivo es permitirle desarrollar proyectos mucho más rápido de lo que podría si estuviera escribiendo código desde cero, proporcionando un rico conjunto de bibliotecas para tareas comúnmente necesarias, así como una interfaz simple y una estructura lógica para acceder a estas bibliotecas.

Codeigniter le permite concentrarse creativamente en su proyecto minimizando la cantidad de código necesario para una tarea determinada.

Siempre que ha sido posible, CodeIgniter se ha mantenido lo más flexible posible, lo que le permite trabajar de la forma que desee, sin verse obligado a trabajar de una forma determinada. El marco puede tener partes centrales fácilmente extendidas o reemplazadas por completo para que el sistema funcione de la manera que lo necesita. En resumen, CodeIgniter es el marco maleable que intenta proporcionar las herramientas que necesita mientras se mantiene alejado e implementa el patrón de diseño MVC.

2.1.2.5 Desarrollo de Software

Según Noriega, R. (2017), manifiesta que el proceso de desarrollo de un software es una escritura utilizada para el desarrollo de un producto de software. Entre sus sinónimos están “ciclo de vida” y “proceso de software”. El desarrollo de un software es una actividad completamente diferente a todo lo que la industria construyó desde los tiempos de la revolución industrial. Por ello el desarrollo de software requiere prácticas especiales de gestión de proyecto.

2.1.2.6 Metodología de desarrollo RUP

Para el desarrollo del software “**lpharmaexpress**” se utilizará la metodología RUP, según Kruchten, P. (2004) define lo siguiente:

El Proceso Unificado de Rational, de manera similar a UML, es fruto de los aportes de un gran número de investigadores y empresas de desarrollo de programas. Entre los métodos más importantes que constituyen la base de RUP figuran los siguientes, que como puede verse, cubren diversos aspectos del ciclo de vida de desarrollo:

- Objectory: Método de desarrollo propuesto originalmente por Jacobson, caracterizado por ser un método orientado a objetos centrado alrededor de Casos de Uso.

- Rational Approach: Método de desarrollo resultante de la unificación de los conceptos desarrollados por Kruchten, Booch y Royce, entre los que se destacan los de proceso iterativo y desarrollo centrado en la arquitectura del programa.
- SQA Process: Método de pruebas.
- Requirements College: Guías para la gestión de requisitos.

RUP es un proceso de ingeniería de programación diseñado para garantizar software de alta calidad que se pueda producir, satisfacer las necesidades del cliente y producir de acuerdo con planes y presupuestos predecibles. Su característica más importante es:

- Es un proceso iterativo, basado en el refinamiento sucesivo del sistema.
- Es un proceso controlado, donde juegan un papel de primordial importancia la gestión de requisitos y el control de los cambios.
- Basado en la construcción de modelos visuales del sistema.
- Centrado en el desarrollo de la arquitectura, por lo que maneja el concepto de desarrollo basado en componentes.
- Conducido por los Casos de Uso.
- Soporta técnicas orientadas a objetos y en particular el uso de UML.
- Configurable.
- Fomenta el control de calidad.
- Soportado por herramientas.

Organización, No cabe duda de que en el desarrollo de una aplicación se sigue un proceso, y el proceso se irá realizando paso a paso hasta que se construya hasta que paulatinamente se entienda la función requerida y cómo realizar la función. Esto requiere la ejecución de un conjunto de actividades que se gestionan como un proyecto, es decir, con metas finales, plazos y

presupuestos. Como en cualquier proyecto, es importante contar con puntos de control intermedios denominados hitos a lo largo del proceso de ejecución, estos hitos se establecen al momento de preparar el plan de trabajo y se utilizan como balizas para verificar que el proyecto avanza con normalidad. En el modelo en cascada, el proceso de desarrollo avanza en secuencia a través de cinco actividades básicas: captura de requisitos, análisis, diseño, implementación y prueba. El modelo señala que cada actividad debe completarse antes de poder realizar la siguiente actividad, para estas actividades, ellas mismas pueden servir de referencia para el avance del proyecto en el tiempo y se denominan fases. Por tanto, un proyecto se planifica tomando como hitos la realización de diferentes etapas, en las que habitualmente se entregan uno o más productos relacionados con el desarrollo del sistema. RUP rompe la secuencia de las actividades básicas del modelo en cascada al proponer un desarrollo incremental e iterativo, en el que no es necesario agotar por completo una actividad para iniciar la siguiente. En cambio, se ha avanzado mediante la construcción de prototipos, cada uno de los cuales requiere parte de las actividades básicas para ser realizadas. Entonces, puede pensar en el desarrollo incremental como una serie de iteraciones, cada una de las cuales se lleva a cabo en un modelo en cascada. Esta estrategia significa que es imposible seguir utilizando la finalización de las actividades básicas para establecer hitos del proyecto, porque esto sucede al final de la ejecución del proyecto. Entonces es necesario establecer un nuevo estándar para definir los puntos de control del proyecto; el estándar determinado por el producto obtenido en sucesivas iteraciones. Por lo tanto, RUP organiza las actividades de desarrollo de acuerdo con dos estándares ortogonales. En el eje vertical describimos lo que llamamos actividades básicas, estas actividades se denominan componentes en términos de RUP, estos componentes determinan cómo mejorar la conceptualización y construcción del sistema. Corresponden a la estructura estática del proceso de desarrollo porque definen las operaciones que se deben realizar. En el eje horizontal se describen los criterios de planificación y control en el tiempo.

Corresponden a la dinámica del proceso de desarrollo y determinan cuándo deben ejecutarse las acciones definidas por el componente.

Organización por Componentes, Los componentes del proceso de desarrollo agrupan las actividades de acuerdo con su nivel y naturaleza, y determinan las operaciones que deben realizarse, quién debe realizarlas y cómo. Los siguientes términos describen cada componente:

- Artefacto (artifacts), representa cualquier tipo de información generada, modificada o utilizada en el desarrollo del sistema. Por ejemplo: en el componente "Análisis", la "Clase de análisis" se especifica en detalle.
- Trabajadores (workers), correspondiente al rol del componente de intervención (la misma persona puede desempeñar múltiples roles).
- Flujos de trabajo (workflows), Y las actividades deben ser realizadas por trabajadores para obtener artefactos de componentes.

Ciclo de vida, el ciclo de vida de RUP es la realización del desarrollo en espiral. Se ensambla ensamblando elementos en una secuencia semiordenada. El ciclo de vida organiza las tareas en fases e iteraciones. RUP divide el proceso en cuatro etapas, en esta etapa, según los diferentes proyectos, habrá múltiples iteraciones y más o menos enfoque en diferentes actividades. La primera iteración (en las fases de "inicio" y "especificar") se centró en comprender el problema y la tecnología, delinear el alcance del proyecto, eliminar los riesgos clave y establecer una línea de base arquitectónica. En la etapa inicial, la iteración se centrará más en los requisitos y las actividades de modelado empresarial.

En la fase de elaboración, desarrollo iterativo de arquitectura orientada a puntos de referencia, que incluye más flujo de trabajo de requisitos, modelo de negocio (optimización), análisis, diseño e implementación de la arquitectura orientada a la línea de base.

En la fase de construcción, La construcción del producto se completa a través de una serie de iteraciones. Para cada iteración, se seleccionarán algunos casos de uso, se mejorará su análisis y diseño, y se implementarán y probarán. Haz una pequeña cascada en cada ciclo. Realice iteraciones hasta que se complete la implementación de la nueva versión del producto.

En la fase de transición, el objetivo es garantizar que haya un producto listo para ser entregado a la comunidad de usuarios. Se puede ver en cada etapa que intervienen todas las disciplinas, pero dependiendo de la etapa, el trabajo dedicado a esa disciplina será diferente.

Hay dos tipos de componentes: componentes de procesos de ingeniería, que se refieren a actividades directamente relacionadas con la obtención de productos; soporte de componentes, se refieren a actividades de gestión de procesos. El proceso de ingeniería consta de siete componentes, a saber:

- **Modelado de la Organización.** Consiste en la identificación y documentación de la estructura y funcionamiento de la organización en la que se ejecutará la aplicación a desarrollar. Su propósito es permitir que los clientes y desarrolladores comprendan los problemas de la organización, identifiquen posibles mejoras y determinen el impacto que la aplicación a desarrollar tendrá en la organización.
- **Captura de Requisitos.** Su propósito es obtener la descripción de para qué sirve el sistema, y lograr un acuerdo entre el equipo de desarrollo y el cliente en este aspecto.
- **Análisis.** Este componente define la estructura (clases, paquetes, etc.) y el comportamiento del sistema. Su propósito es obtener una descripción de cómo funciona el sistema.

- **Diseño.** Si bien el Análisis se enfoca en establecer la función del sistema, el componente de Diseño se enfoca en hacer posible la función dada en la arquitectura física (computadora, red, etc.) y el entorno de implementación (sistema operativo, lenguaje de programación, etc.). Su propósito es obtener una descripción de cómo está construido el sistema.
- **Implementación.** Construcción del sistema obteniendo los archivos ejecutables, de configuración, librerías, etc.
- **Pruebas.** Se verifican los modelos, prototipos y demás artefactos ejecutables del sistema bajo desarrollo.
- **Puesta en Servicio.** En este componente se realizarán las actividades necesarias para la puesta en funcionamiento del producto en las instalaciones del cliente. Por su parte, contiene tres componentes de apoyo, a saber:
 - **Gestión de configuración y cambios.** Realiza un seguimiento de la evolución iterativa del sistema, registra la modificación de sus partes y la configuración que provocó la operación del prototipo.
 - **Gestión del proyecto.** Definir los detalles del proceso de desarrollo iterativo. Con este fin, proporciona un marco de razonamiento para la gestión de proyectos intensivos en planes, así como directrices prácticas para la planificación, la composición del equipo de trabajo, la ejecución y supervisión de proyectos y los estándares de gestión de riesgos.
 - **Entorno.** Su propósito es establecer la organización del entorno de desarrollo de programación (procesos y herramientas) que

requiere el equipo de desarrollo.

Organización en el tiempo, Establece la dinámica del proceso de desarrollo y define los criterios para planificar y controlar su ejecución en el tiempo. Se expresa en términos de ciclos, fases, iteraciones e hitos:

- **Ciclo:** Desarrollo de una nueva versión del producto. Corresponde al ámbito de un proyecto.
- **Fases:** Etapas en el desarrollo de una versión (proyecto). Cada fase termina en un Hito y puede estar dividida en Iteraciones. Las Fases y los Hitos asociados son:
- **Hito:** Punto de control, donde generalmente se revisan los resultados del proceso y se decide si se avanza a la siguiente Fase o Iteración.
- **Iteración:** Unidad de desarrollo del producto, en la que se obtienen o refinan uno o más artefactos del sistema.

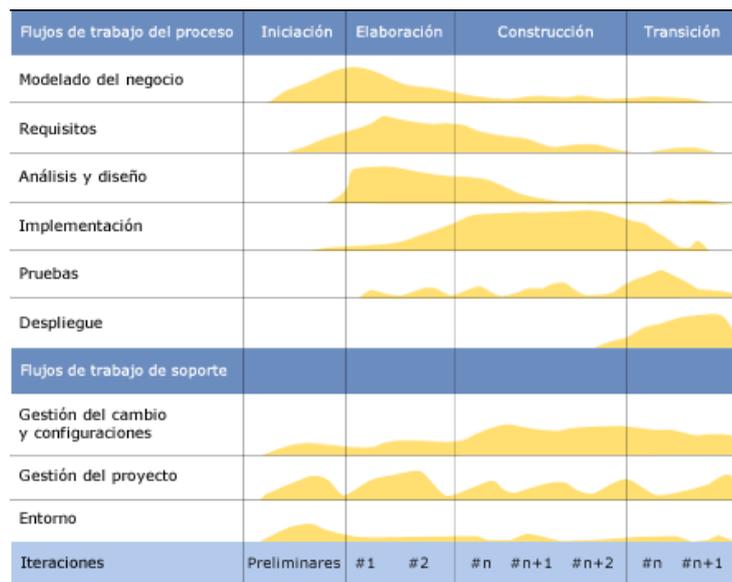


Figura 9. Esfuerzos en actividades según fase del proyecto.

Fuente: Wikipedia.com

2.1.2.7 UML (Lenguaje Unificado de Modelado)

El lenguaje de modelado unificado (UML) es un lenguaje de modelado visual que se utiliza para especificar, visualizar, construir y registrar los artefactos de los sistemas de software. Capturar decisiones y conocimiento sobre el sistema a construir. Se utiliza para comprender, diseñar, inspeccionar, configurar, mantener y controlar la información sobre dichos sistemas. Está diseñado para usarse con todos los métodos de desarrollo, etapas del ciclo de vida, dominios de aplicación y medios. El lenguaje de modelado tiene como objetivo unificar la experiencia pasada en tecnología de modelado e incorporar las mejores prácticas actuales en métodos estándar. UML incluye conceptos semánticos, símbolos y principios generales. Tiene partes estáticas, dinámicas, ambientales y organizativas. Está destinado a ser utilizado con herramientas de modelado visual interactivo con generadores de código y generadores de informes. La especificación UML no define un proceso estándar, pero está pensada para ser utilizada en el proceso de desarrollo iterativo. Diseñado para admitir la mayoría de los procesos de desarrollo orientados a objetos.

UML captura información sobre la estructura estática y el comportamiento dinámico del sistema. El sistema se modela como una colección de objetos discretos que interactúan para realizar un trabajo que en última instancia beneficia a los usuarios externos. La estructura estática define los tipos de objetos que son importantes para el sistema y su implementación, así como las relaciones entre los objetos. El comportamiento dinámico define la historia temporal de los objetos y la comunicación entre objetos para lograr sus objetivos. Puede comprender el modelado del sistema desde diferentes perspectivas (independientes pero relacionadas entre sí) para diferentes propósitos.

2.1.2.8 Norma ISO/IEC 25000 SQuaRE

De acuerdo con la norma ISO / IEC 25000 (2016) denominada SQuaRE (Requisitos y Evaluación de Calidad de Sistemas y Software), se trata de una serie de normas que tiene como objetivo crear un marco común para evaluar la calidad de los productos de software. La serie ISO / IEC 25000 es el resultado de la evolución de otras normas anteriores (especialmente la norma ISO / IEC 9126 que describe las peculiaridades de los modelos de calidad de productos de software) y la ISO / IEC 14598 que aborda el proceso de evaluación de productos de software. La serie de normas ISO / IEC 25000 consta de cinco partes.

ISO/IEC 2500n – División de Gestión de Calidad

Las normas que forman este apartado definen todos los modelos, términos y definiciones comunes referenciados por todas las otras normas de la familia 25000. Actualmente esta división se encuentra formada por:

- **ISO/IEC 25000 - *Guide to SQuaRE***: contiene el modelo de la arquitectura de SQuaRE, la terminología de la familia, un resumen de las partes, los usuarios previstos y las partes asociadas, así como los modelos de referencia.
- **ISO/IEC 25001 - *Planning and Management***: establece los requisitos y orientaciones para gestionar la evaluación y especificación de los requisitos del producto software.

ISO/IEC 2501n – División de Modelo de Calidad

Las normas de este apartado presentan modelos de calidad detallados incluyendo características para calidad interna, externa y en uso del producto software. Actualmente esta división se encuentra formada por:

- **ISO/IEC 25010 - *System and software quality models***: describe el modelo de calidad para el producto software y para la calidad en uso. Esta Norma presenta las

características y subcaracterísticas de calidad frente a las cuales evaluar el producto software.

- **ISO/IEC 25012 - *Data Quality model***: define un modelo general para la calidad de los datos, aplicable a aquellos datos que se encuentran almacenados de manera estructurada y forman parte de un Sistema de Información.

ISO/IEC 2502n – División de Medición de Calidad

Estas normas incluyen un modelo de referencia de la medición de la calidad del producto, definiciones de medidas de calidad (interna, externa y en uso) y guías prácticas para su aplicación. Actualmente esta división se encuentra formada por:

- **ISO/IEC 25020 - *Measurement reference model and guide***: Presenta una explicación introductoria y un modelo de referencia común a los elementos de medición de la calidad. También proporciona una guía para que los usuarios seleccionen o desarrollen y apliquen medidas propuestas por normas ISO.
- **ISO/IEC 25021 - *Quality measure elements***: Define y especifica un conjunto recomendado de métricas base y derivadas que puedan ser usadas a lo largo de todo el ciclo de vida del desarrollo software.
- **ISO/IEC 25022 - *Measurement of quality in use***: Define específicamente las métricas para realizar la medición de la calidad en uso del producto.
- **ISO/IEC 25023 - *Measurement of system and software product quality***: Define específicamente las métricas para realizar la medición de la calidad de productos y sistemas software.
- **ISO/IEC 25024 - *Measurement of data quality***: Define específicamente las métricas para realizar la medición de la calidad de datos.

ISO/IEC 2503n – División de Requisitos de Calidad

Las normas que forman este apartado ayudan a especificar requisitos de calidad que pueden ser utilizados en el proceso de

elicitación de requisitos de calidad del producto software a desarrollar o como entrada del proceso de evaluación. Para ello, este apartado se compone de:

- **ISO/IEC 25030 - *Quality requirements***: Provee de un conjunto de recomendaciones para realizar la especificación de los requisitos de calidad del producto software.

ISO/IEC 2504n – División de Evaluación de Calidad

Este apartado incluye normas que proporcionan requisitos, recomendaciones y guías para llevar a cabo el proceso de evaluación del producto software. Esta división se encuentra formada por:

- **ISO/IEC 25040 - *Evaluation reference model and guide***: Propone un modelo de referencia general para la evaluación, que considera las entradas al proceso de evaluación, las restricciones y los recursos necesarios para obtener las correspondientes salidas.
- **ISO/IEC 25041 - *Evaluation guide for developers, acquirers and independent evaluators***: Describe los requisitos y recomendaciones para la implementación práctica de la evaluación del producto software desde el punto de vista de los desarrolladores, de los adquirentes y de los evaluadores independientes.
- **ISO/IEC 25042 - *Evaluation modules***: Define lo que la Norma considera un módulo de evaluación y la documentación, estructura y contenido que se debe utilizar a la hora de definir uno de estos módulos.
- **ISO/IEC 25045 - *Evaluation module for recoverability***: Define un módulo para la evaluación de la subcaracterística Recuperabilidad (Recoverability).

La división de extensión de SQuaRE (ISO/IEC 25050 a ISO/IEC 25099) se reserva para normas o informes técnicos que aborden

dominios de aplicación específicos o que puedan ser utilizados para complementar otras normas de la familia SQuaRE.



Figura 10. División del ISO/IEC 25000 SQUARE

Fuente: <https://iso25000.com/index.php/normas-iso-25000>

2.1.2.8.1 Métricas de Calidad del Producto Software (Calidad Interna y Externa) - ISO/IEC 25023

Las métricas de calidad internas y externas evalúan las características del modelo de calidad del producto software definido en el punto anterior. La siguiente tabla describe las métricas de calidad internas y externas:

MÉTRICAS PARA LA CALIDAD INTERNA/EXTERNA		
Características	Subcaracterísticas	Métricas
Adecuación funcional	Complejidad funcional	➤ Complejidad de la implementación funcional.
	Exactitud funcional	➤ Exactitud. ➤ Precisión computacional.
Fiabilidad	Madurez	➤ Disipación del fallo. ➤ Suficiencia de las pruebas. ➤ Tiempo medio entre fallos.

	Disponibilidad	<ul style="list-style-type: none"> ➤ Tiempo de servicio. ➤ Tiempo medio de inactividad.
	Tolerancia a fallos	<ul style="list-style-type: none"> ➤ Prevención de fallas. ➤ Redundancia (componentes). ➤ Anulación de operación incorrecta.
	Recuperabilidad	<ul style="list-style-type: none"> ➤ Tiempo medio de recuperación.
Eficiencia en el desempeño	Comportamiento temporal	<ul style="list-style-type: none"> ➤ Tiempo de respuesta. ➤ Tiempo de espera. ➤ Rendimiento.
	Utilización de recursos	<ul style="list-style-type: none"> ➤ Líneas de código. ➤ Utilización de CPU. ➤ Utilización de la memoria. ➤ Utilización de los dispositivos de E/S.
	Capacidad	<ul style="list-style-type: none"> ➤ Número de peticiones online. ➤ Número de accesos simultáneos. ➤ Sistema de transmisión de ancho de banda.
Facilidad de uso	Capacidad de reconocer su adecuación	<ul style="list-style-type: none"> ➤ Integridad de descripción. ➤ Capacidad de demostración.
	Capacidad de ser entendido	<ul style="list-style-type: none"> ➤ Funciones evidentes. ➤ Efectividad de la documentación del usuario o ayuda del sistema.
	Operatividad	<ul style="list-style-type: none"> ➤ Recuperabilidad de error operacional. ➤ Claridad de mensajes. ➤ Consistencia operacional. ➤ Posibilidad de personalización.
	Protección contra errores del usuario	<ul style="list-style-type: none"> ➤ Verificación de entradas válidas. ➤ Prevención del uso incorrecto.
	Estética de la Interfaz del usuario	<ul style="list-style-type: none"> ➤ Personalización de la apariencia de la interfaz del usuario.
	Accesibilidad técnica	<ul style="list-style-type: none"> ➤ Accesibilidad física.
Seguridad	Confidencialidad	<ul style="list-style-type: none"> ➤ Capacidad de control de acceso. ➤ Encriptación de datos.

	Integridad	➤ Prevención de corrupción de datos.
	No repudio	➤ Utilización de firma digital.
	Responsabilidad	➤ Capacidad de auditoría de acceso.
	Autenticidad	➤ Métodos de autenticación.
	Co – Existencia	➤ Co – existencia disponible.
Compatibilidad	Interoperatividad	<ul style="list-style-type: none"> ➤ Conectividad con sistemas externos. ➤ Capacidad de intercambiar de datos.
	Modularidad	<ul style="list-style-type: none"> ➤ Capacidad de condensación. ➤ Acoplamiento de clases.
Mantenibilidad	Reusabilidad	➤ Ejecución de reusabilidad.
	Capacidad de ser analizado	<ul style="list-style-type: none"> ➤ Capacidad de pistas de auditoría. ➤ Diagnóstico de funciones suficientes.
	Capacidad de ser modificado	<ul style="list-style-type: none"> ➤ Complejidad ciclomática. ➤ Profundidad de herencia. ➤ Grado de localización de corrección de impacto. ➤ Complejidad de modificación. ➤ Índice de éxito de modificación.
	Capacidad de ser probado	<ul style="list-style-type: none"> ➤ Completitud funcional de funciones de pruebas. ➤ Capacidad de prueba autónoma. ➤ Capacidad de reinicio de pruebas.
	Adaptabilidad	<ul style="list-style-type: none"> ➤ Adaptabilidad en entorno hardware. ➤ Adaptabilidad en entorno de software. ➤ Adaptabilidad en entorno organizacional.
Portabilidad	Capacidad de ser Instalado	<ul style="list-style-type: none"> ➤ Eficiencia en el tiempo de instalación. ➤ Facilidad de instalación.
	Capacidad de ser Reemplazado	<ul style="list-style-type: none"> ➤ Consistencia en la función de soporte al usuario. ➤ Inclusividad funcional. ➤ Uso continuo de datos.

Tabla 13. Métricas para la calidad Interna/Externa

Fuente: Balseca (2014)

2.1.2.9 Modelo de Evaluación De Calidad Usando ISO/IEC 25040

Según Balseca (2014), la norma ISO / IEC 25040 proporciona un modelo de referencia y una descripción del proceso de evaluación de la calidad del producto de software y determina los requisitos de aplicación del proceso. Este proceso se puede utilizar para la evaluación de la calidad interna, externa y en uso.

2.1.2.9.1 Modelo de referencia para la evaluación de calidad del producto software

Según Balseca (2014), el modelo de referencia describe los insumos, resultados, limitaciones y recursos del proceso de evaluación, como

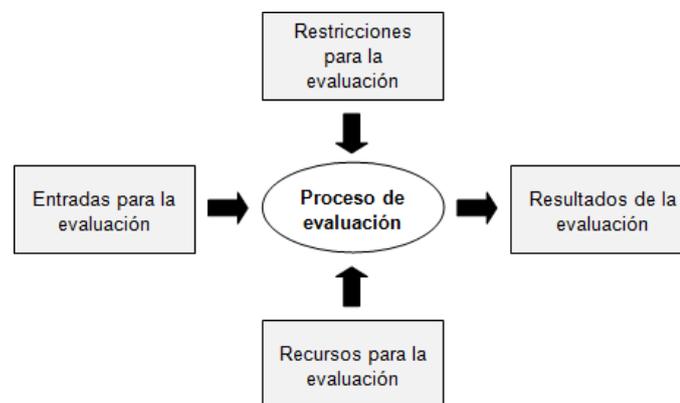


Figura 11. Modelo de referencia para la evaluación de calidad del producto de software.

Fuente: ISO/IEC 25040

Como ejemplo las entradas, salidas, recursos y restricciones pueden incluir lo siguiente:

1. Entradas

- Requisitos de evaluación de calidad del producto software.
- Especificación de requisitos de calidad del producto software.
- Producto software y productos intermedios a ser evaluados.

2. Salidas

- Reporte de evaluación.

- Plan de evaluación de calidad del producto software.
- Criterios de decisión definidos por las métricas de calidad.
- Criterios de decisión para la evaluación.
- Planificación de las actividades de evaluación.
- Métricas de calidad

3. Recursos

- Documentos SQuaRE: ISO/IEC 25001, 25010, 2502n, 25030 y 25040.
- Metodología y herramientas de medida apropiadas
- Recursos humanos y económicos para la evaluación.
- Sistema de información para la evaluación.
- Bases de dato para la evaluación.

4. Restricciones

Restricciones con respecto a:

- Recursos para la evaluación.
- Horarios para la evaluación.
- Costos para la evaluación.
- Entornos para la evaluación.
- Metodología y herramientas para la evaluación.
- Informes para la evaluación

2.1.2.9.2 Modelo de referencia para la evaluación de calidad del producto software

Describe el proceso general y detalla las actividades, tareas, su propósito, insumos, resultados e información complementaria para la evaluación de la calidad.

Para el proceso de evaluación de productos de software, se identifican dos roles: solicitante y evaluador.

- El primer rol puede estar representado por el desarrollador, proveedor, adquirente o usuario de software.
- El segundo rol es representado por el evaluador.

Se presentan los procesos generales con las respectivas tareas

y los resultados entregables que se deben obtener:

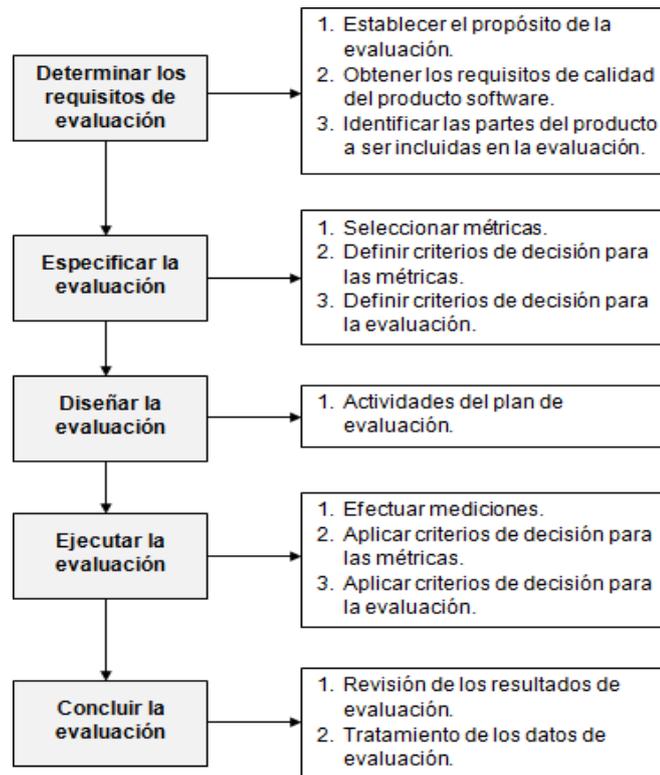


Figura 12. Modelo de referencia para la evaluación de calidad del producto software

Fuente: Balseca (2014)

La evaluación de la calidad puede ser realizada durante o después del proceso de desarrollo o adquisición, por la institución / empresa del desarrollo, adquisición o personal de evaluación independiente.

2.2 Marco regulatorio / legal

2.2.1 Ley de boticas y farmacias

Según La Ley de los Productos Farmacéuticos, Dispositivos Médicos y Productos Sanitarios N° 29459, publicada en noviembre de 2009, orienta la regulación de estos productos sustituyendo al Capítulo III de la Ley General de Salud N.º 26842. A través de esta Ley se ha modificado los aspectos más cuestionados en la Ley N° 26842, al establecer requisitos para solicitar el registro sanitario de productos farmacéuticos que son necesarios para garantizar su eficacia, seguridad y calidad, incluyendo también los plazos necesarios para su evaluación y que el costo de la tasa por registro sanitario sea en función de lo que implique otorgar dicho registro, lo cual incluye

también las actividades de control y vigilancia sanitaria. Así mismo, restablece la exigencia de la autorización sanitaria de funcionamiento, previa al inicio de las actividades, previa inspección para verificar el cumplimiento de los dispositivos legales vigentes. La Ley incorpora también tres capítulos específicos sobre acceso, uso racional de productos farmacéuticos, dispositivos médicos y productos sanitarios, así como un capítulo de investigación. A continuación, se muestra la lista de normas vinculadas a la atención de boticas y farmacias:

Documentos	Nombre del Documento
DOCUMENTOS NACIONALES	Ley General de Salud
	Buenas Prácticas de Prescripción
	Listado de Productos Farmacéuticos para venta sin receta médica en establecimientos farmacéuticos
	Listado de Productos Farmacéuticos para venta sin receta médica en establecimientos no farmacéuticos
	Listado de Productos y Servicios que no podrán ser ofrecidos en Farmacias y Boticas
	Listado de Productos Farmacéuticos y Afines para Botiquines
	Establecen requisitos y condiciones sanitarias mínimas que deben cumplir locales de farmacias y boticas
	Petitorio Nacional de Medicamentos Esenciales - 2005
	Medicamentos Esenciales y sus alternativas de marca (2006)
	Evaluación de la situación de los medicamentos en el Perú
Manual de Buenas Prácticas de Dispensación	
DOCUMENTOS INTERNACIONALES	Desarrollo de la Práctica de Farmacia, centrada en la atención al paciente
	Criterios éticos para la promoción de medicamentos de la organización mundial de la salud (OMS 1998)
	Normas para la calidad de los servicios farmacéuticos (FIP - OMS)
	El papel del Farmacéutico en la lucha contra la pandemia del VIH-SIDA
	Declaración conjunta por la Federación Farmacéutica Internacional y la Industria Mundial de la Automedicación Responsable
	Declaración de la FIP sobre Normas Profesionales - La Atención Farmacéutica
	Declaración de la FIP sobre estándares profesionales - Errores de medicación asociados a los medicamentos de prescripción
	Declaración de normas profesionales sobre recetas electrónicas
	FIP - Guías para el etiquetado de medicamentos prescritos
	Declaración de las normas profesionales de FIP - El papel del farmacéutico en la estimulación de la adherencia a los tratamientos a largo plazo
	Declaración de la FIP de estándares profesionales en la dispensación de medicamentos que afectan la capacidad de conducción
	Declaración de la FIP sobre estándares profesionales de códigos éticos para farmacéuticos
	Autocuidado - Incluida la automedicación responsable - El papel profesional del farmacéutico
	La autoridad del farmacéutico en la selección del producto: intercambio terapéutico y sustitución genérica
	La Buena Práctica en la donación de medicinas
Federación Farmacéutica Internacional - Declaración de principios Buenas Prácticas de Educación Farmacéutica	

Federación Farmacéutica Internacional - Informe sobre la política de control de resistencia a los antimicrobianos
Declaración de principios de la FIP - La función del farmacéutico y su responsabilidad en la educación de niños y adolescentes sobre el uso de medicamentos
Declaración de política de la FIP sobre el papel del farmacéutico en la promoción de un futuro sin tabaco
Declaración de FIP de política sobre medicamentos falsificados
Declaración de la FIP sobre la política de confidencialidad de información obtenida durante el ejercicio farmacéutico
Posicionamiento político de la FIP sobre el papel del farmacéutico en la prevención y tratamiento de las enfermedades crónicas

Tabla 14. Tabla de normas de la ley de farmacias y boticas.

Fuente: <http://www.digemid.minsa.gob.pe/Main.asp?seccion=813>

2.2.2 Protocolo para efectuar la atención farmacéutica en pacientes con sospecha de coronavirus “covid-19”

Según el Minsa, El coronavirus (CoV) es una amplia familia de virus que pueden causar una variedad de enfermedades, desde el resfriado común hasta enfermedades más graves como el coronavirus del síndrome respiratorio de Oriente Medio (MERS-CoV) y el coronavirus del síndrome respiratorio. -CoV). Un nuevo tipo de coronavirus es un nuevo tipo de coronavirus que nunca se ha encontrado en humanos. El coronavirus puede transmitirse de animales a humanos (transmisión animal). Según una extensa investigación, sabemos que el coronavirus del SARS se transmite de la civeta a los humanos, mientras que el coronavirus MERS se transmite de dromedarios a los humanos. Además, se sabe que otros coronavirus circulan entre animales y aún no han infectado a los humanos. Al igual que otros miembros de la familia del coronavirus que infecta a los humanos, el SARSCoV-2 puede causar diversas manifestaciones clínicas cubiertas por el término COVID-19, que incluyen fiebre alta y síntomas respiratorios, desde el resfriado común hasta la neumonía grave, acompañada de síndrome de dificultad respiratoria aguda, séptica shock, neumonía múltiple. -Fallo orgánico e incluso muerte. El consejo habitual para evitar la propagación de infecciones es una buena higiene de las manos y las vías respiratorias (cubrirse la boca y la nariz al toser y estornudar) y carne y huevos bien cocidos. Además, evite el contacto cercano con cualquier persona que tenga signos de enfermedad respiratoria (como toser o estornudar). Hasta la fecha, la mayoría de los casos de COVID-19 notificados (aproximadamente el 80%) son leves, y los casos más graves se presentan en personas de edad avanzada. No existe un tratamiento específico, pero algunos medicamentos

antivirales se están utilizando en ensayos clínicos y han demostrado cierta eficacia en estudios recientes. También se está produciendo una vacuna candidata contra el coronavirus, pero llevará alrededor de un año determinar su eficacia preventiva. Sí, existen muchos tratamientos para controlar sus síntomas, por lo que la atención médica puede mejorar su pronóstico. Ante el creciente número de nuevos casos de COVID-19 en nuestro país, es necesario preparar todas las puertas que puedan ingresar al sistema de salud. Por tanto, como profesionales sanitarios que desarrollan su actividad profesional desde las farmacias de primera línea (es decir, farmacias), los farmacéuticos de las distintas farmacias deben seguir las mismas recomendaciones que hace el Ministerio de Sanidad para los profesionales sanitarios, especialmente los incluidos en la atención primaria.

2.2.3 Facturación Electrónica

SUNAT (2020) señala que la Factura Electrónica: “Es el tipo de comprobante de pago denominado FACTURA, emitido a través del sistema de emisión electrónica desarrollado desde los sistemas del contribuyente. Mediante el Sistema de Emisión Electrónica desde los sistemas desarrollados por el contribuyente, se emiten también las Notas de Débito y Crédito vinculadas a la Factura Electrónica desde el Contribuyente”. Se detalla las siguientes resoluciones que iniciaron este sistema con sus anexos:

- Resolución de Superintendencia N° 097-2012/SUNAT (anexos al final de página).
- Resolución de Superintendencia N° 227-2012/SUNAT
- Resolución de Superintendencia N° 251-2012/SUNAT
- Resolución de Superintendencia N° 279-2012/SUNAT
- Resolución de Superintendencia N° 065-2013/SUNAT - Anexos
- Resolución de Superintendencia N° 288-2013/SUNAT
- Resolución de Superintendencia N° 374-2013/SUNAT - Anexo
- Resolución de Superintendencia N° 199-2014/SUNAT - Anexos
- Resolución de Superintendencia N° 300-2014/SUNAT - Anexos
- Resolución de Superintendencia N° 086-2015/SUNAT - Anexo
- Resolución de Superintendencia N° 137-2015/SUNAT
- Resolución de Superintendencia N° 185-2015/SUNAT

Anexos de la RS N° 097-2012/SUNAT (modificados por RS 065-2013/SUNAT, RS 199-2014/SUNAT y RS 300-2014/SUNAT)

Anexo 1	Contenido de la Factura Electrónica
Anexo 2	Contenido de la Boleta de Venta Electrónica
Anexo 3	Contenido de la Nota de Crédito Electrónica
Anexo 4	Contenido de la Nota de Débito Electrónica
Anexo 5	Resumen diario de boletas de venta y notas asociadas
Anexo 6	Consideraciones Técnicas
Anexo 7	Declaración Jurada
Anexo 8	Catálogo de Códigos
Anexo 9	Estándar UBL de los documentos electrónicos
Anexo 10	Contenido de la comunicación de baja
Anexo 11	Archivo - Resumen de Comprobantes Impresos

Tabla 15. Anexos normas legales - Factura Electrónica

Fuente: <https://orientacion.sunat.gob.pe>

2.2.4 Derecho de autor

Según INDECOPI (2020) sobre protección de derechos de autor de informes legales, es importante incluir en el contrato de servicio o en el contenido del propio informe, restricciones al uso de terceros no relacionados con el autor del informe, así como cláusulas de confidencialidad para el autor y el cliente, por favor Considere la información privilegiada que es esencial para redactar informes en algunos casos. De hecho, la "Ley de Propiedad Intelectual" (Decreto Legislativo No. 822) establece en su artículo 3 que la protección de los derechos de autor pertenece a todas las obras de arte en el campo de la literatura o el arte, independientemente de su género, expresión, valor o finalidad, por lo que no hay nada Puede evitar que los informes legales de los abogados se conviertan en objeto de protección por derechos de autor.

Sin embargo, como se mencionó anteriormente, hay que considerar que no todo lo creado por la propia creatividad es susceptible de protección por derechos de autor, porque existen características o requisitos básicos que todos deben cumplir. Las creaciones deben considerarse obras y, por lo tanto, deben considerarse derechos de autor. Lo que llamamos este requisito es "original". Aunque no está definido en nuestra ley de derechos de autor actual, el precedente de cumplimiento obligatorio emitido por la Oficina de Propiedad Intelectual ha enriquecido su contenido. INDECOPI, estableció la

siguiente:

"La originalidad de una obra debe entenderse como la expresión creativa y personalizada (o forma representativa) de la obra, independientemente del grado de creación e individualización. La obra debe expresar el carácter propio del autor y llevar la impronta de su personalidad. Arte, ciencia o literatura Las partes que ya forman parte del patrimonio cultural no serán consideradas como individuos, ni estarán comprometidos con la naturaleza de las cosas o simplemente expresados mediante la aplicación mecánica de ciertas normas legales. Se simplificará en técnicas simples o simples instrucciones. En forma de expresiones, estas técnicas o instrucciones simples solo requieren habilidades manuales para ejecutarse. Por lo tanto, no todas las obras producidas bajo los esfuerzos de sus creadores deben estar protegidas por derechos de autor. Asimismo, incluso si es cierto que una creación que carece de individualidad se copia textualmente, esta situación no la convierte en una obra. Como hemos visto, aunque el concepto de originalidad parece arbitrario y difícil de explicar en situaciones reales, el INDECOPÍ lo contextualiza y define cuidadosamente para que se adapte a cada caso concreto e ignore La subjetividad del estándar.

2.2.5 Tasa Única de Procedimientos Administrativos UPP

Actualmente, se cuenta con un TUPA aprobado 2018.

CAPÍTULO III: PROCESOS DEL SISTEMA

3.1 Introducción

Boticas el gran poder es una empresa que apertura sus operaciones comerciales en enero del 2020, con respecto al proceso de ventas, para llevar el respectivo control utiliza un sistema elaborado en Excel.

CANT.	DESCRIPCION	P. UNIT.	IMPORTE	EXISTENCIA	UNIT. PROM.	PRECIO
	AZADOR (BILL LEBOWHART MOD 349)	841.00	841.00	1		1
	PLANCHAS BLD	226.70	226.70	5		1
	ALACENA CARTONERA					1
	ALACENA COMPACTA					1
	ALACENA CRICETA					1
	AZADOR (BILL LEBOWHART MOD 349)					1
	BODINA RADSON (MOD PPSA) USB					1
	LICUADORA BNC (MOD BTR) 1.5L					1
	TOSTADOR DE PAN AMERICAN					1
	PLANCHAS BLD					1

SUB-TOTAL	107.98	EFECTIVO:	
IMPORTE	88.082	CAMERO:	-
I.V.A.	19.898	Impresión documento:	
TOTAL	225.89	Sistema de copias:	

CANTIDAD CON LETRA
DINERO DOSCIENTOS VEINTYSEIS PESOS 89/100 C/100

Figura 13. Registro de ventas excel
Fuente: Boticas el Gran Poder.

En este Excel se registran los pagos de los clientes, el stock de los medicamentos y su precio, no se cuenta con una conexión a SUNAT para realizar la comunicación directa de documentos electrónicos. Los datos no se

registran en una base de datos, por lo que ocurrió en varias ocasiones la duplicidad de información y mal control del stock de medicamentos.

3.2 Recopilación y análisis de la información

3.2.1 Recopilación de procesos anteriores

A continuación, se mostrará las figuras de los procesos anteriores detallados.

Reporte de ventas por vendedor

Botica El Gran Poder Fecha: 11.12.20								
#	Cliente	Vendedor	Comprobante	Pago	Estado	Número	Fecha	Total S/.
10460	PUBLICO EN GENERAL	MARCO ANTONIO ROMÁN VILCARROMERO	FACTURA	EFFECTIVO	PAGADO	00000008	2020-10-31 14:51:00	120.00
10459	PUBLICO EN GENERAL	MARCO ANTONIO ROMÁN VILCARROMERO	FACTURA	EFFECTIVO	PAGADO	00000007	2020-10-23 20:43:00	60.00
10458	PUBLICO EN GENERAL	MARCO ANTONIO ROMÁN VILCARROMERO	FACTURA	EFFECTIVO	PAGADO	00000006	2020-10-21 23:49:00	9.00
10457	PUBLICO EN GENERAL	MARCO ANTONIO ROMÁN VILCARROMERO	BOLETA	EFFECTIVO	PAGADO	00000008	2020-10-20 13:12:00	32.00
10456	PUBLICO EN GENERAL	MARCO ANTONIO ROMÁN VILCARROMERO	BOLETA	EFFECTIVO	PAGADO	00000007	2020-10-20 13:04:00	85.00
10455	PUBLICO EN GENERAL	MARCO ANTONIO ROMÁN VILCARROMERO	BOLETA	EFFECTIVO	PAGADO	00000006	2020-10-20 12:50:00	9.60
10454	PUBLICO EN GENERAL	MARCO ANTONIO ROMÁN VILCARROMERO	BOLETA	EFFECTIVO	PAGADO	00000005	2020-10-20 12:22:00	20.00
10453	PUBLICO EN GENERAL	MARCO ANTONIO ROMÁN VILCARROMERO	BOLETA	EFFECTIVO	PAGADO	00000004	2020-10-17 10:40:00	51.00
10452	PUBLICO EN GENERAL	MARCO ANTONIO ROMÁN VILCARROMERO	BOLETA	EFFECTIVO	PAGADO	00000003	2020-10-17 10:32:00	60.00
10451	PUBLICO EN GENERAL	MARCO ANTONIO ROMÁN VILCARROMERO	FACTURA	EFFECTIVO	PAGADO	00000005	2020-10-14 00:17:00	68.00
10450	PUBLICO EN GENERAL	MARCO ANTONIO ROMÁN VILCARROMERO	FACTURA	EFFECTIVO	PAGADO	00000004	2020-10-13 10:13:00	51.00
10449	PUBLICO EN GENERAL	MARCO ANTONIO ROMÁN VILCARROMERO	FACTURA	EFFECTIVO	PAGADO	00000003	2020-10-13 09:48:00	70.00
10448	PUBLICO EN GENERAL	MARCO ANTONIO ROMÁN VILCARROMERO	FACTURA	EFFECTIVO	PAGADO	00000002	2020-10-12 20:39:00	120.00
10447	PUBLICO EN GENERAL	MARCO ANTONIO ROMÁN VILCARROMERO	FACTURA	EFFECTIVO	PAGADO	00000001	2020-10-12 13:04:00	20.00
10445	PUBLICO EN GENERAL	MARCO ANTONIO ROMÁN VILCARROMERO	TICKET	EFFECTIVO	PAGADO	00000002	2020-10-12 12:20:00	3.20
10443	PUBLICO EN GENERAL	MARCO ANTONIO ROMÁN VILCARROMERO	BOLETA	EFFECTIVO	PAGADO	00000001	2020-10-05 11:49:00	2.00
							Total S/	780.8

Figura 14. Reporte de registro de ventas por vendedor

Fuente: Boticas el gran poder.

Reporte de productos

#	Categoría	Laboratorio	Nombre	Precio Caja	PVP1	PVP2	Precio Blister	Stock	F. Vencimiento
1	FARMACIA	3M PERU	ESPARADRAPO DE SEDA 1.8MT X 2.5CM (MEDICAL)	0	3	0	0	100	2024-12-01
2	FARMACIA	3M PERU	ESPARADRAPO DE SEDA 4.5MT X 1.25CM (MEDICAL)	0	2,9	0	0	100	2024-12-01
3	FARMACIA	3M PERU	ESPARADRAPO DE SEDA 4.5MT X 2.5CM MEDICAL	0	4,5	0	0	100	2024-12-01
4	FARMACIA	3M PERU	ESPARADRAPO DURAPOR 5CMX9.1M	0	15,5	0	0	100	2024-03-01
5	FARMACIA	3M PERU	ESPARADRAPO DURAPORE X 6 CORTES	0	16,33	0	0	100	2024-01-01
6	FARMACIA	3M PERU	ESPARADRAPO HIPOALERGICO 0.9X1.25CM (MEDICAL)	0	1,2	0	0	100	2024-12-01
7	FARMACIA	3M PERU	ESPARADRAPO HIPOALERGICO 4.5MT X 2.5 CM (MEDICAL)	0	3,7	0	0	100	2022-05-01
8	FARMACIA	3M PERU	ESPARADRAPO HIPOALERGICO 4.5MTX 1.25CM (MEDICAL)	0	2,5	0	0	100	2024-11-01
9	FARMACIA	3M PERU	ESPARADRAPO MICROPOROSO 1.25 CM X 1YD	0	1	0	0	100	2022-11-01
10	FARMACIA	3M PERU	ESPARADRAPO TELA ADHESIVA 1.25CMX5 YDS	0	3,89	0	0	100	2022-05-01
11	FARMACIA	3M PERU	ESPARADRAPO TELA ADHESIVA1.25 CM X 1Y	0	1,2	0	0	100	2022-01-01
12	FARMACIA	3M PERU	ESPARADRAPO TRANS. 4.5MT X 1.25CM (MEDICAL)	0	3	0	0	100	2024-12-01
13	FARMACIA	3M PERU	ESPARADRAPO TRANSPARENTE 1.25CM X 1 Y	0	1,45	0	0	100	2022-05-01
14	FARMACIA	3M PERU	ESPARADRAPO TRANSPARENTE 1.8MT X 2.5 CM (MEDICAL)	0	2,5	0	0	100	2024-12-01
15	FARMACIA	3M PERU	ESPARADRAPO TRANSPARENTE 4.5MT X 2.50CM (MEDICAL)	0	5,6	0	0	100	2024-12-01
16	FARMACIA	ABBOTT	CLUVAX X 3 OVULOS	0	12	0	0	100	2021-03-01
17	FARMACIA	ABBOTT	CREMA ADHESIVA Y-KELIN X 40GR - MENTA	0	37,5	0	0	100	2021-12-01
18	LACTEOS	ABBOTT	ENFAGROW PRE VAINILLA 1100PRO MENTAL	0	91,9	0	0	100	2021-11-01

Figura 15. Reporte de productos

Fuente: Boticas el gran poder.

3.2.2 Entrevista a los actores

• Gerente/Dueño (AC1)

Pta1.1 ¿Cómo se realiza el control de ventas de los medicamentos?

Respuesta. Actualmente se tiene un archivo Excel donde es como un sistema que lleva el control de las ventas de los medicamentos.

Pta1.2 ¿Cómo se obtiene los reportes para la toma de decisiones?

Respuesta. La reportaría funciona con una macro que está configurada en el Excel que tenemos como sistema. No se cuenta actualmente con una base de datos, los datos se van transfiriendo en otra hoja Excel aparte, para evitar la saturación y tener un histórico de esta información.

• Químico Farmacéutico (AC2)

Pta1.1 ¿Cómo se clasifican los medicamentos?

Respuesta. Los medicamentos se clasifican de acuerdo con el laboratorio que los fabrica y el lote.

Pta1.2 ¿Cómo se realiza el proceso de registro de medicamentos?

Respuesta. Los medicamentos se registran actualmente en el archivo Excel, de acuerdo con el número de lote y la empresa que los fabrica, además el

stock se suma de forma manual.

Pta1.3 ¿Cómo se controlan los lotes de los medicamentos?

Respuesta. Cuando un cliente solicita un medicamento se verifica el lote más antiguo, para evitar pérdidas.

Pta1.4 ¿Existe demora en la atención de los clientes?

Respuesta. Si, sobre todo los fines de semana. El sistema no es óptimo, para estos casos, porque solo lo puede ocupar una sola persona y no se puede usar en simultaneo.

- **Cajero (AC3)**

Pta1.1 ¿Existe demora en el proceso de venta?

Respuesta. Si existe demora porque a veces el cliente tiene una receta amplia de medicamentos y como el stock a veces no se actualiza y ya no hay unidades entonces se pierde tiempo buscando el medicamento solicitado en almacén, ocasionando a veces que el cliente este descontento por este error.

Pta1.2 ¿Existe demora en ubicar el medicamento en el sistema actual?

Respuesta. Si existe demora, por los errores en control de stock.

Pta1.3 ¿Existe duplicidad de información en el sistema actual?

Respuesta. Si en algunos casos, porque la información se introduce sin validaciones.

Pta1.4 ¿Existe demora en el despacho de documentos de venta?

Respuesta. Si porque se realizan de forma manual.

Pta1.5 ¿Se puede vender medicamentos en soles o dólares?

Respuesta. Si, pero no hay un control adecuado en la caja.

Pta1.6 ¿Existe errores al momento de controlar el stock?

Respuesta. Si, porque el stock se introduce de forma manual.

Pta1.7 ¿Existe demora al registrar clientes?

Respuesta. Actualmente no se cuenta con esa funcionalidad.

- **Contador (AC4)**

Pta1.1 ¿Existe demora en el proceso de contabilidad?

Respuesta. Si existe demora porque los documentos están en papel físico, no se procesan fácilmente.

3.2.3 Listado de requerimientos por actor

A continuación, se presenta el listado de requerimientos funcionales y no funcionales por relevancia de actores dentro del negocio:

FECHA	15/12/2020
FUNCION	Gerente/Dueño
ANALISTA	Martín Diego Pinedo Dávila
VERSION	1.0
REQUERIMIENTOS FUNCIONALES	
R 1.1. Registro de medicamentos.	
R 1.2. Registro de clientes persona naturales y jurídicas.	
R 1.3. Reporte de ventas.	
R 1.4. Reporte de medicamentos próximos a vencer.	
R 1.5. Reporte de caja clasificado por ventas al contado, crédito, soles y dólares.	
R 1.6. Listado de medicamentos con su stock.	
R 1.7. Listado de cajeros.	
R 1.8. El sistema debe estar interconectado con SUNAT para la emisión de documentos electrónicos.	
REQUERIMIENTOS NO FUNCIONALES	
R 1.1. El Software deberá tener una interfaz amigable.	
R 1.2. El Software deberá ser libre de pagos por licencias.	

Tabla 16. Requerimiento actor AC1.
Fuente: Elaboración propia.

FECHA	15/12/2020
FUNCION	Químico Farmacéutico
ANALISTA	Martín Diego Pinedo Dávila
VERSION	1.0
REQUERIMIENTOS FUNCIONALES	
R 1.1. Registrar los medicamentos debe permitir clasificar por laboratorio.	
R 1.2. La búsqueda de medicamentos debe permitir filtrar por nombre.	
R 1.3. Visualizar el lote de los medicamentos.	
REQUERIMIENTOS NO FUNCIONALES	
R 1.1. El Software deberá tener una interfaz amigable.	
R 1.2. El Software debe ser rápido.	

Tabla 17. Requerimiento actor AC2
Fuente: Elaboración propia.

FECHA	15/12/2020
FUNCION	Cajeros
ANALISTA	Martín Diego Pinedo Dávila
VERSION	1.0
REQUERIMIENTOS FUNCIONALES	
R 1.1. Búsqueda de medicamentos.	
R 1.2. Opción de venta al contado al crédito.	
R 1.3. Ingresar el tipo de cambio, cuando la venta es en dólares.	
R 1.4. Imprimir directamente en la impresora los documentos electrónicos.	
R 1.5. Visualizar mis ventas realizadas.	
R 1.6. Visualizar el stock de los medicamentos en tiempo real.	
R 1.7. Verificar en el sistema si la venta del medicamento requiere de receta médica.	
R 1.8. Realizar la baja de ventas.	
R 1.9. Visualizar las presentaciones de los medicamentos.	
REQUERIMIENTOS NO FUNCIONALES	
R 1.1. El Software deberá tener una interfaz amigable.	
R 1.2. El Software debe ser rápido.	

Tabla 18. Requerimiento actor AC3
Fuente: Elaboración propia

FECHA	15/12/2020
FUNCION	Contador
ANALISTA	Martín Diego Pinedo Dávila
VERSION	1.0
REQUERIMIENTOS FUNCIONALES	
R 1.1. Software debe estar conectado a SUNAT.	
R 1.2. Visualizar en el reporte de caja las entradas y salidas de dinero.	
R 1.3. Filtrar las ventas por vendedor y fecha.	
R 1.4. Visualizar las ventas dadas de baja.	
R 1.5. Visualizar las ventas en soles y dólares.	
REQUERIMIENTOS NO FUNCIONALES	
R 1.1. El Software deberá tener una interfaz amigable.	
R 1.2. El Software debe ser rápido.	

Tabla 19. Requerimiento Actor AC4
Fuente: Elaboración propia.

3.3 Diagramas de los procesos del negocio

3.3.1 Diagrama macro de gestión de ventas

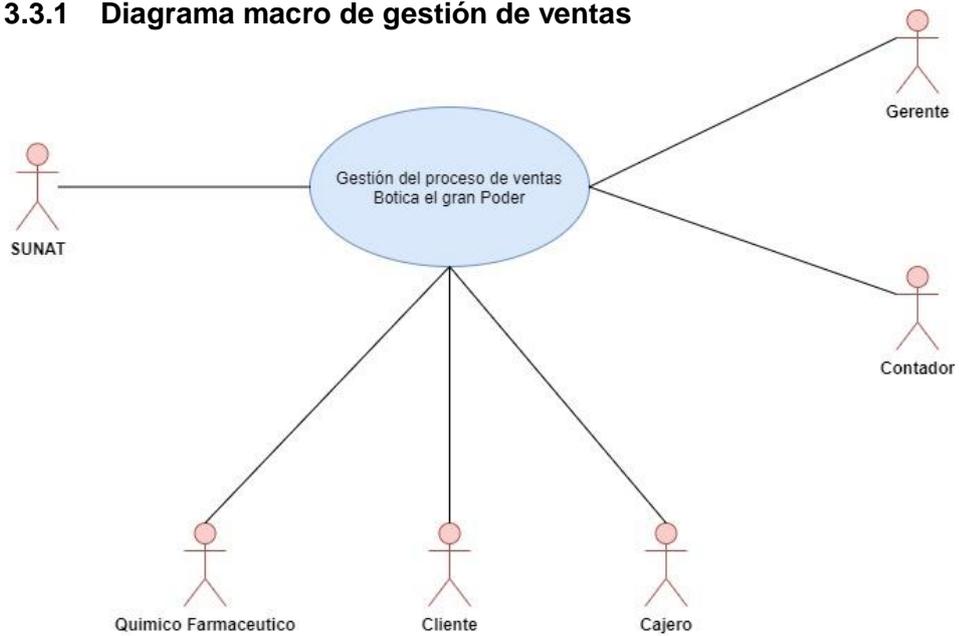


Figura 16. Gestión de ventas
Fuente: Elaboración propia.

3.3.2 Diagrama macro de gestión de medicamentos

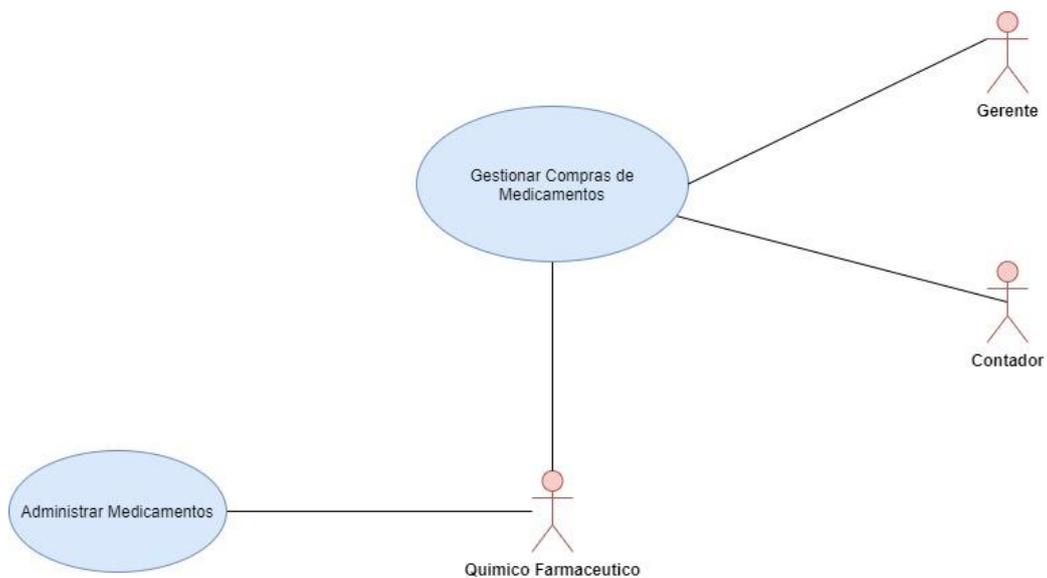


Figura 17. Gestión de medicamentos
Fuente: Elaboración propia.

3.3.3. Diagrama macro de gestión de operadores móviles.

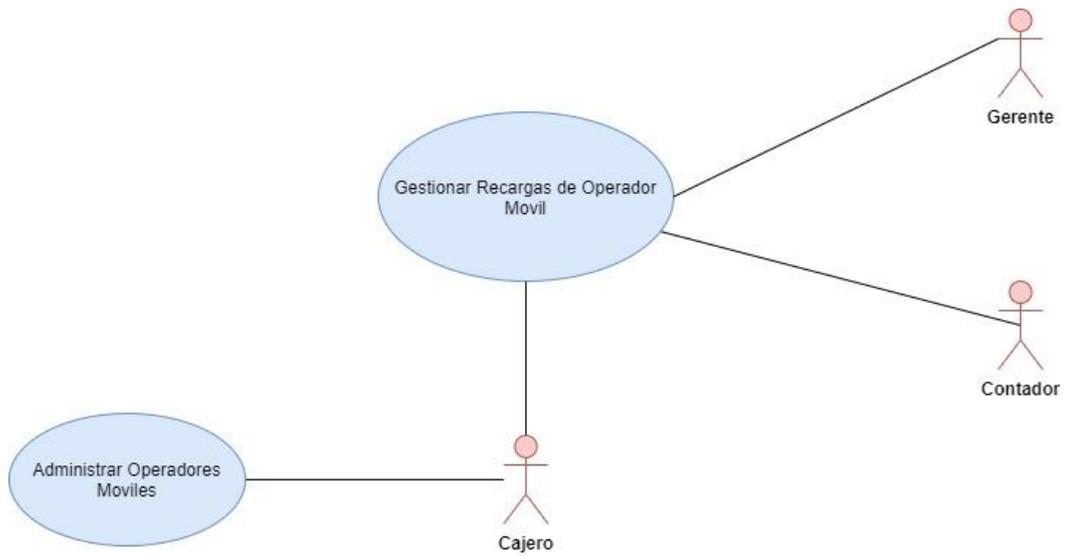


Figura 18. Gestión de operador móvil
Fuente: Elaboración propia.

3.3.4 Diagrama macro de gestión de operadores bancarios.

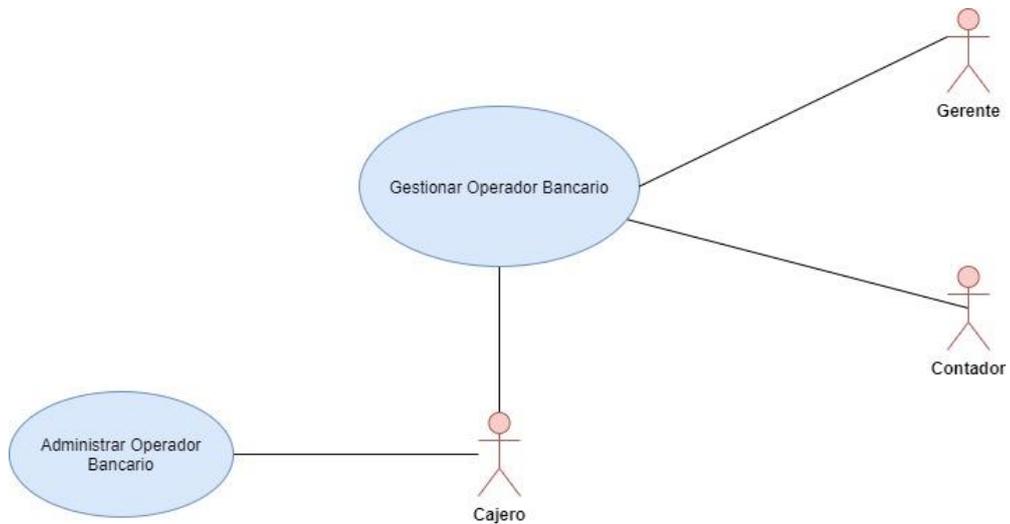


Figura 19. Gestión de operador bancario
Fuente: Elaboración propia.

3.4 Conclusiones

Se concluye lo siguiente, que, habiendo realizado la recopilación y

análisis de la información del proceso de negocio, la entrevista, toma de requerimientos, de forma individual a los actores principales del proceso de negocio a automatizar, se propone el uso de la metodología RUP para el análisis, desarrollo e implementación del sistema, y de esta manera mejorar el proceso de negocio y a su vez poder cumplir con cada uno de los requerimientos presentados por los actores de negocio ya mencionados, creando un software de calidad, intuitivo, rápido y eficiente para el proceso de pagos.

CAPÍTULO IV: ANÁLISIS DE REQUISITOS DEL SISTEMA

4.1 Descripción del sistema

Teniendo en cuenta el sistema “IpharmaExpress” versión 1.0 para el proceso de ventas de la botica el Gran Poder, diseñado e implementado por el bachiller Martin Diego Pinedo Davila basado en metodología RUP y con medición de calidad de software basado en el estándar ISO/IEC 25000 SQuaRE.

Esta versión del sistema incluirá los requerimientos funcionales y no funcionales formulados por los actores, las mismas que se establecieron en el catálogo de requisitos con sus respectivas descripciones. Según estos requisitos se estable como alcance el crear los siguientes módulos:

- **Módulo de clientes.** La funcionalidad de este módulo incluye el mantenimiento de clientes, clasificación de clientes en personas jurídicas y naturales.
- **Módulo de usuarios.** La funcionalidad de este módulo incluye el mantenimiento de usuarios, contraseña, perfiles y permisos.
- **Módulo de documentos electrónicos.** La funcionalidad de este módulo incluye envió de boletas y facturas a SUNAT, por el cual también se implementará la comunicación de baja de documentos electrónicos, nota de débito y nota de crédito.

- **Módulo de reportes.** La funcionalidad de este módulo incluye reporte de caja, reporte de medicamentos, reporte de clientes, reporte de productos más vendidos, reporte de stock.
- **Módulo de medicamentos.** La funcionalidad de este módulo incluye mantenimiento de medicamentos, clasificación por categoría, laboratorio y lote, presentación de medicamentos por unidad, caja, blíster, precios de promociones, indicador si el medicamento requiere receta médica y el control de stock.
- **Módulo de ventas.** La funcionalidad de este módulo incluye, búsqueda de medicamentos, ventas al contado y al crédito, por soles, dólares o euros.
- **Módulo de cajas.** La funcionalidad de este módulo incluye apertura de caja, cierre de caja, resumen de caja.
- **Indicadores.** El sistema “IpharmaExpress” mostrada indicadores en tiempo real, los cuales serán indicados por el stack holder principal.

4.2 Reuniones principales

El sistema IpharmaExpresss recopila toda la información de los requerimientos del negocio para el proceso de ventas. En la investigación se cuenta con la participación de los principales “stackholders” del negocio.

4.3 Objetivos del sistema

El objetivo del sistema IpharmaExpress es el de automatizar el proceso de ventas, poner a disposición la información requerida a los distintos usuarios del sistema para agilizar la búsqueda, control y emisión de documentos electrónicos. Así mismo ser una herramienta que la gerencia utilice para el apoyo de la toma de decisiones.

4.4 Catálogo de requisitos del sistema

4.4.1 Requisitos funcionales del sistema

Los requisitos funcionales del sistema de acuerdo con el análisis que se realizó en la toma de requerimientos en las entrevistas con los “stackholders” y su dependencia entre ellos, se presentan en la siguiente ilustración:

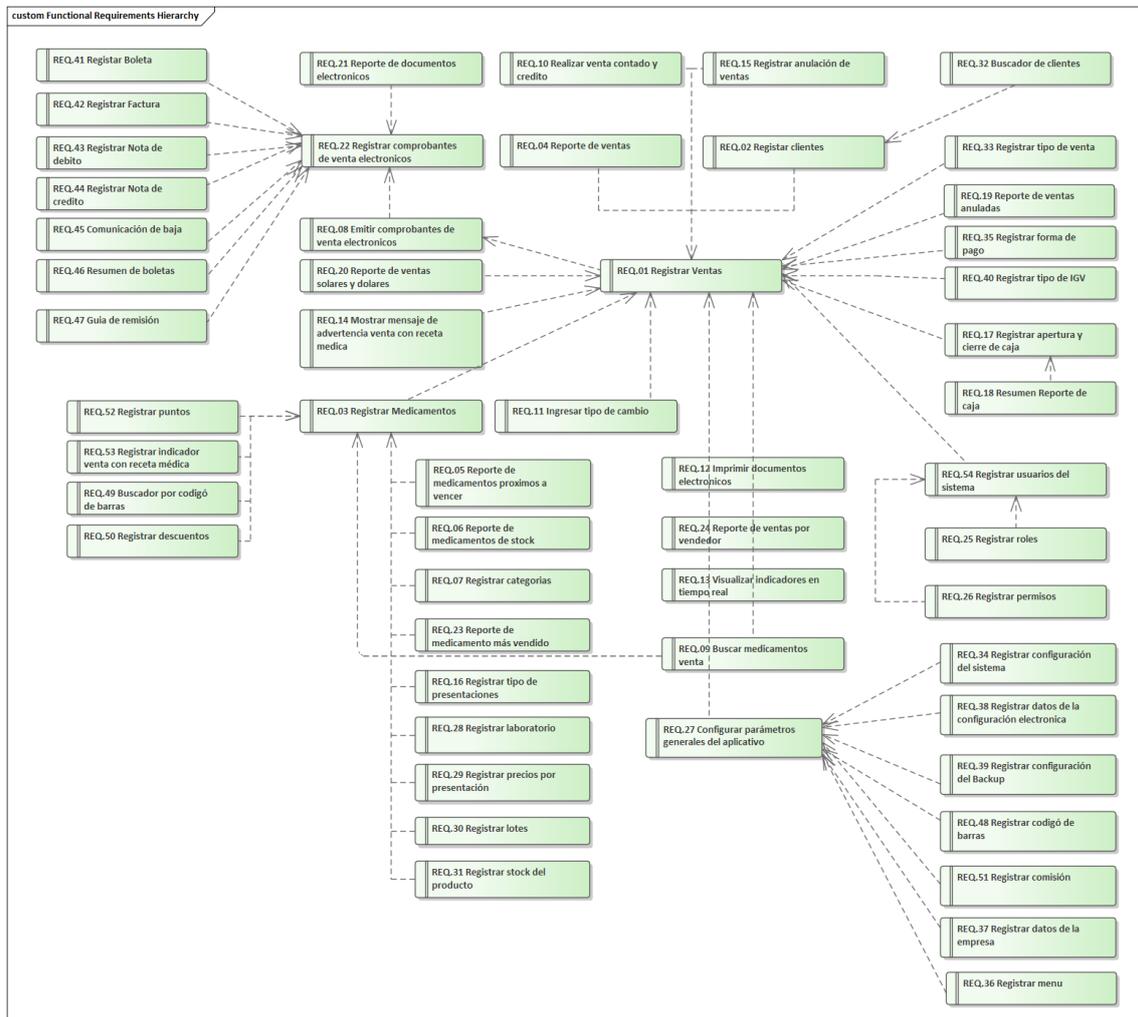


Figura 20. Catálogo de requisitos del sistema

Fuente: Elaboración propia.

4.5 Definición de actores

Nº	Cargo	Descripción
AC1	Gerente/Dueño	Es el dueño y responsable de tomar las dediciones administrativas en la botica el gran poder
AC2	Químico Farmacéutico	Es la persona responsable experta en el suministro de medicamentos
AC3	Cajero	Es la persona responsable que atiende a los clientes y registra las ventas de los medicamentos
AC4	Contador	Es la persona profesional dedicado a aplicar, manejar e interpretar la contabilidad de una organización o persona, con la finalidad de producir informes para la gerencia y para terceros (tanto de manera independiente como dependiente), que sirvan para la toma de decisiones
AC5	Cliente	Es la persona que busca comprar las recetas de sus medicamentos
AC6	Cliente Externo	Persona jurídicas o naturales, entidades públicas.

Tabla 20. Actores del sistema

Fuente: Boticas el Gran Poder – Elaboración propia

4.6 Diagrama de casos de uso

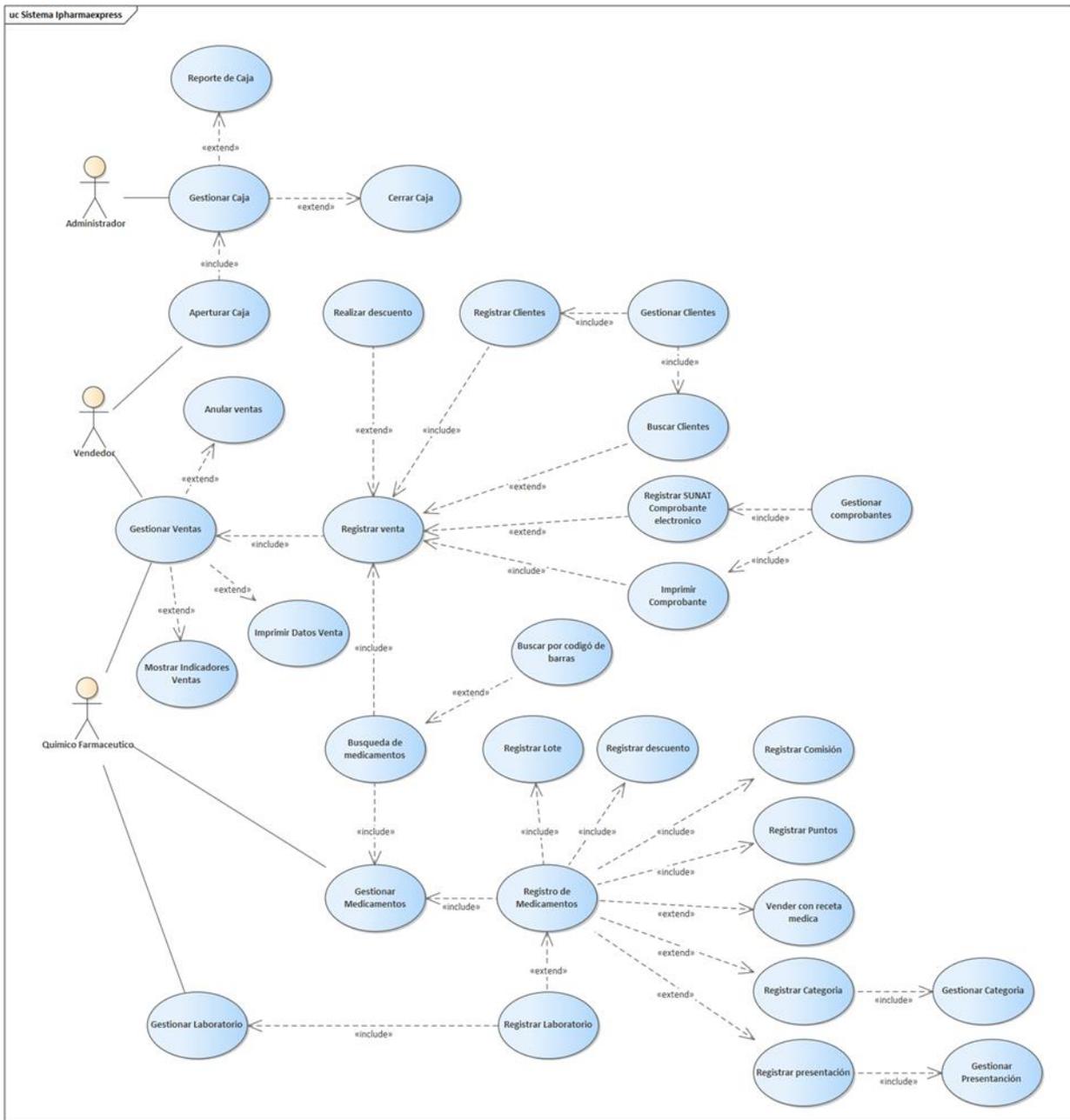


Figura 21. Diagrama de caso de uso del sistema proceso de ventas.

Fuente: Elaboración propia

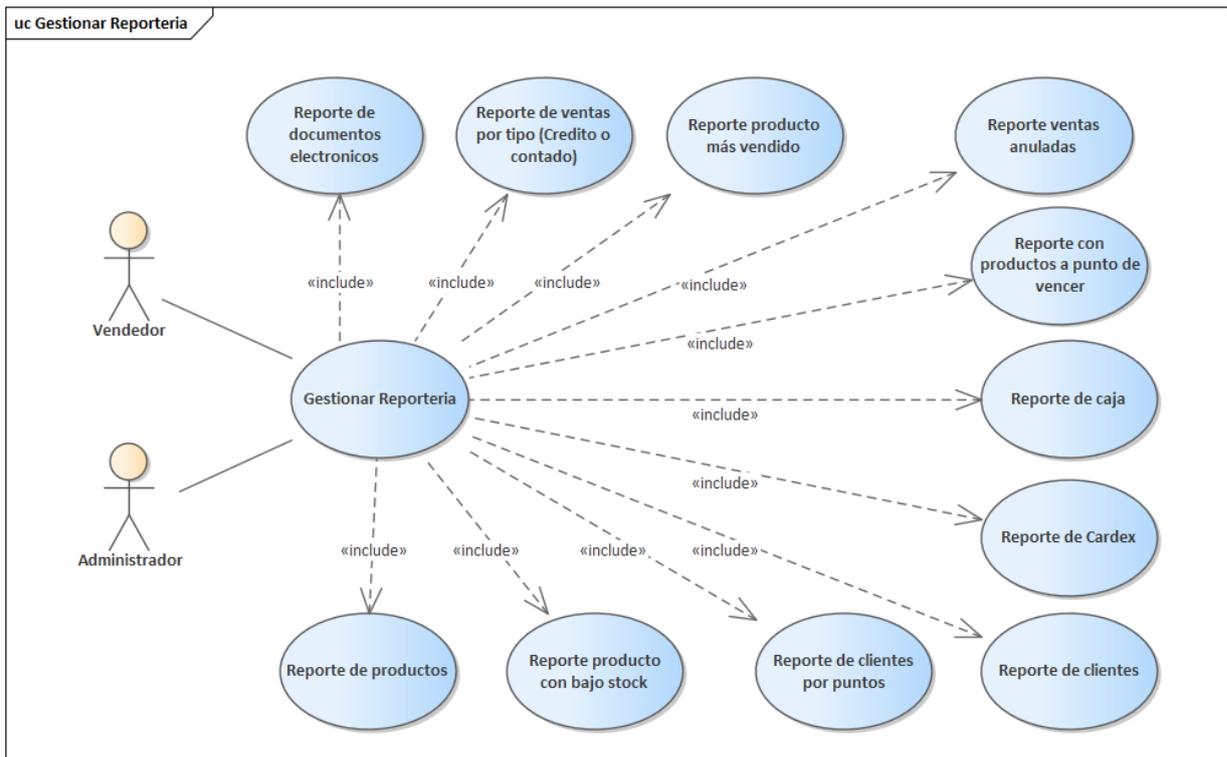


Figura 22. Caso de uso del sistema reportes
Fuente: Elaboración propia

4.7 Casos de uso del sistema

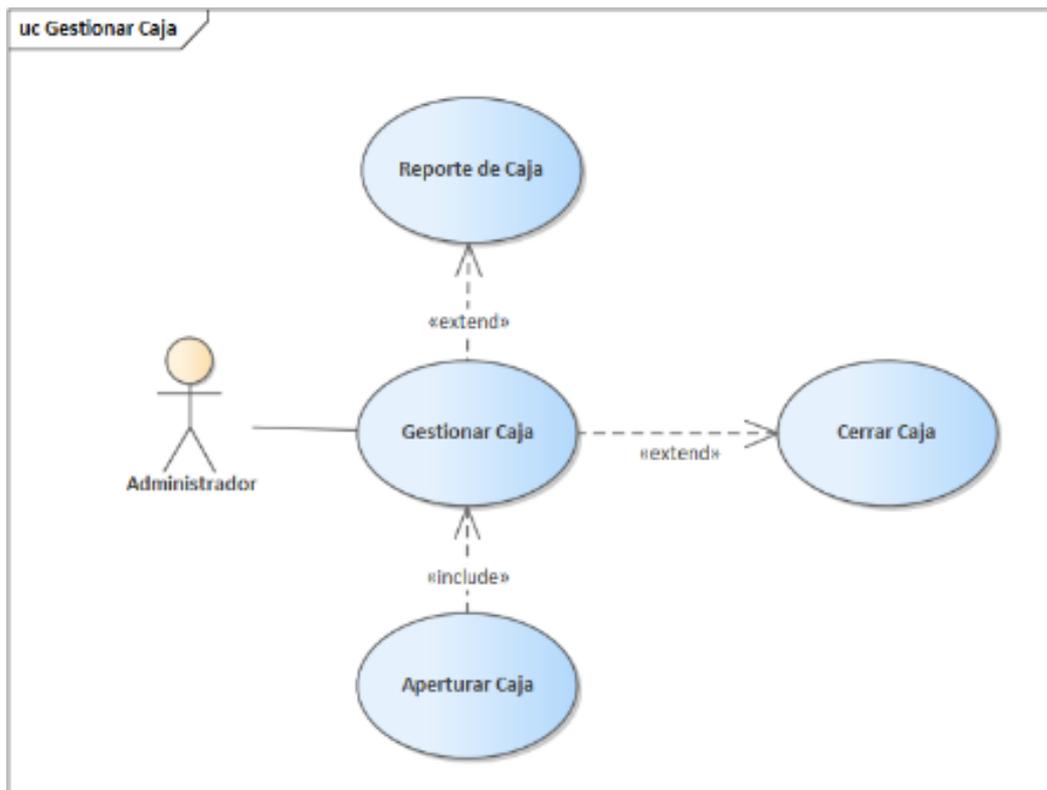


Figura 23. Caso de uso Gestionar Caja
Fuente: Elaboración propia.

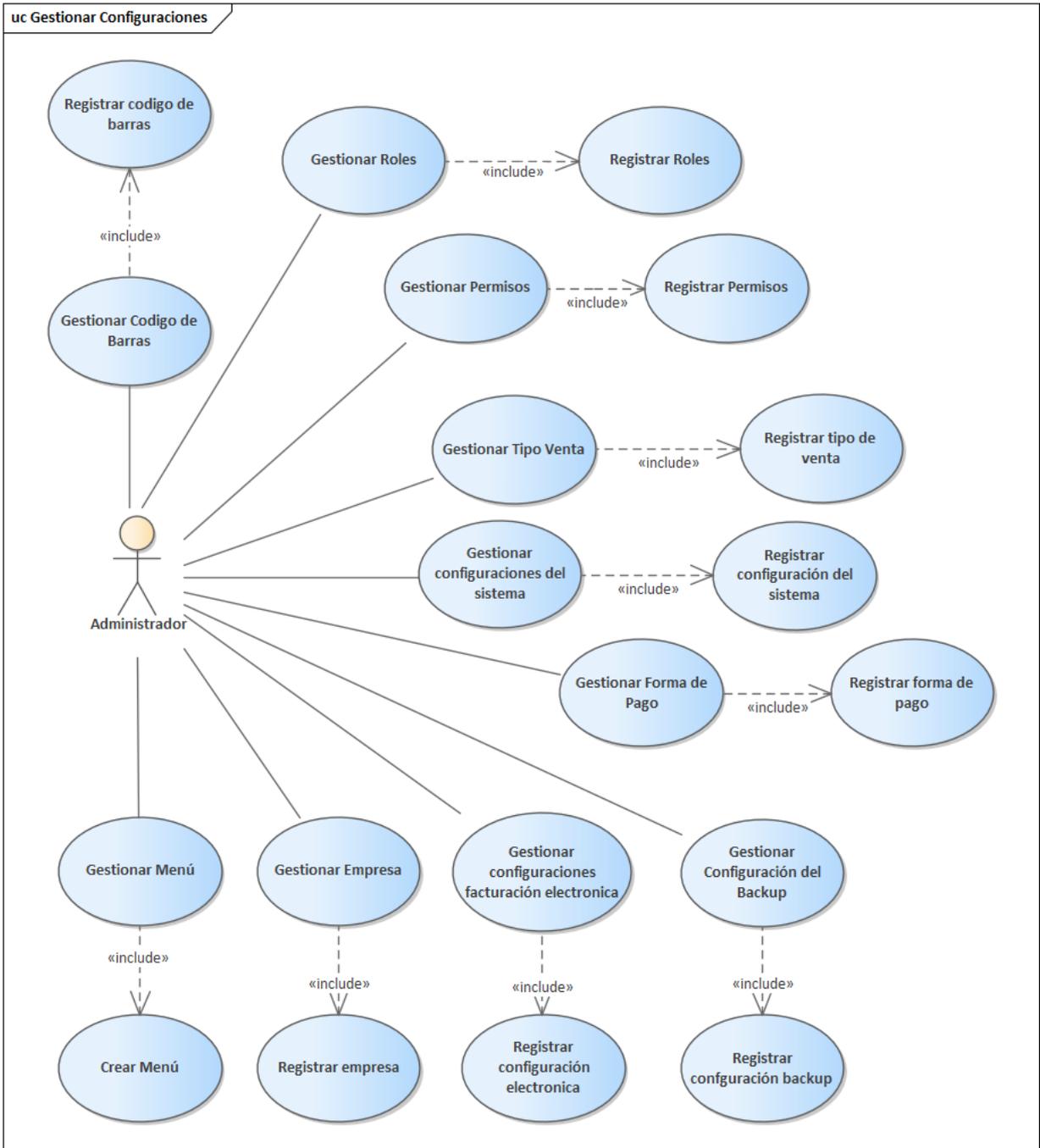


Figura 24. Caso de uso Gestionar Configuraciones

Fuente:

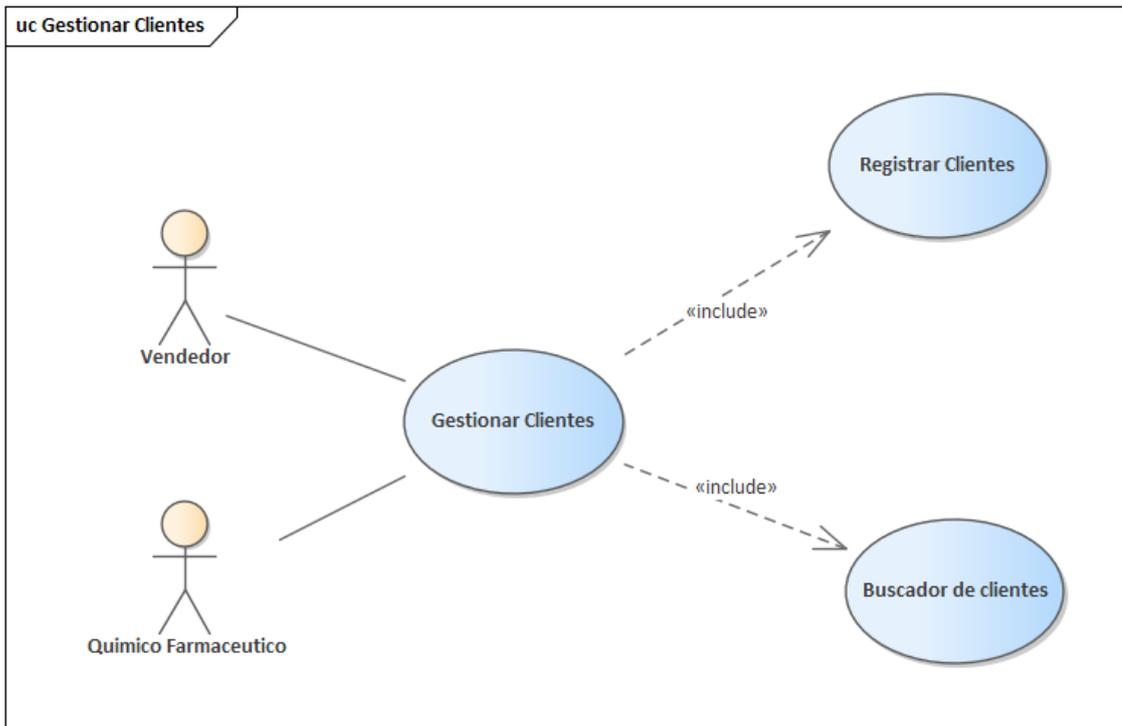


Figura 25. Caso de uso Gestionar Clientes
Fuente: Elaboración propia.

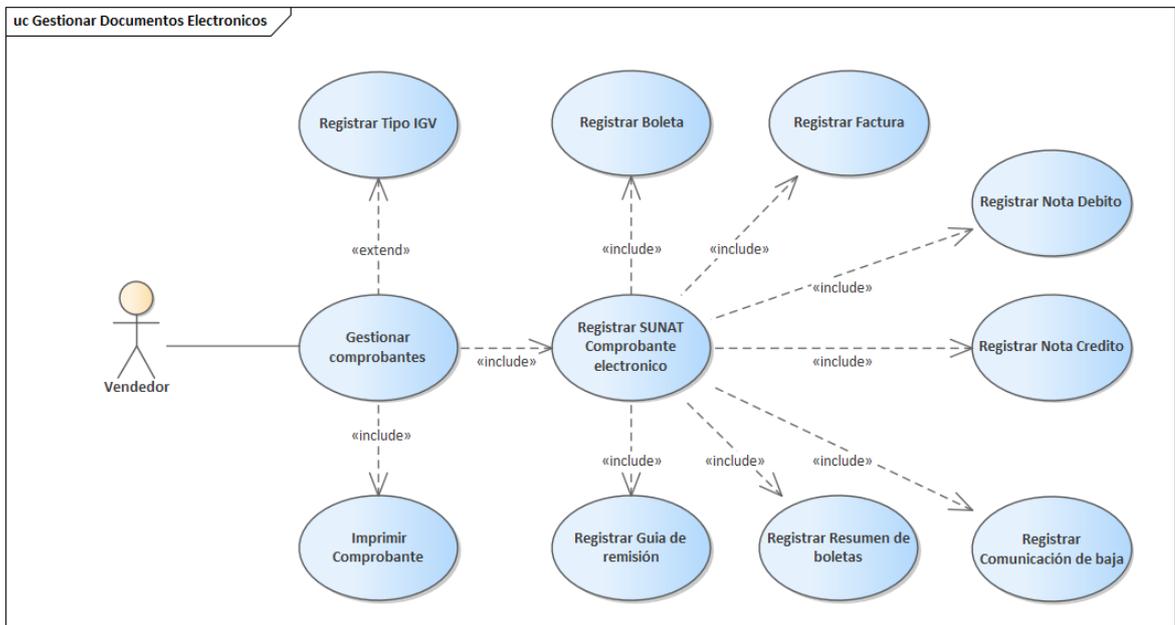


Figura 26. Caso de uso Documentos Electrónicos
Fuente: Elaboración propia.

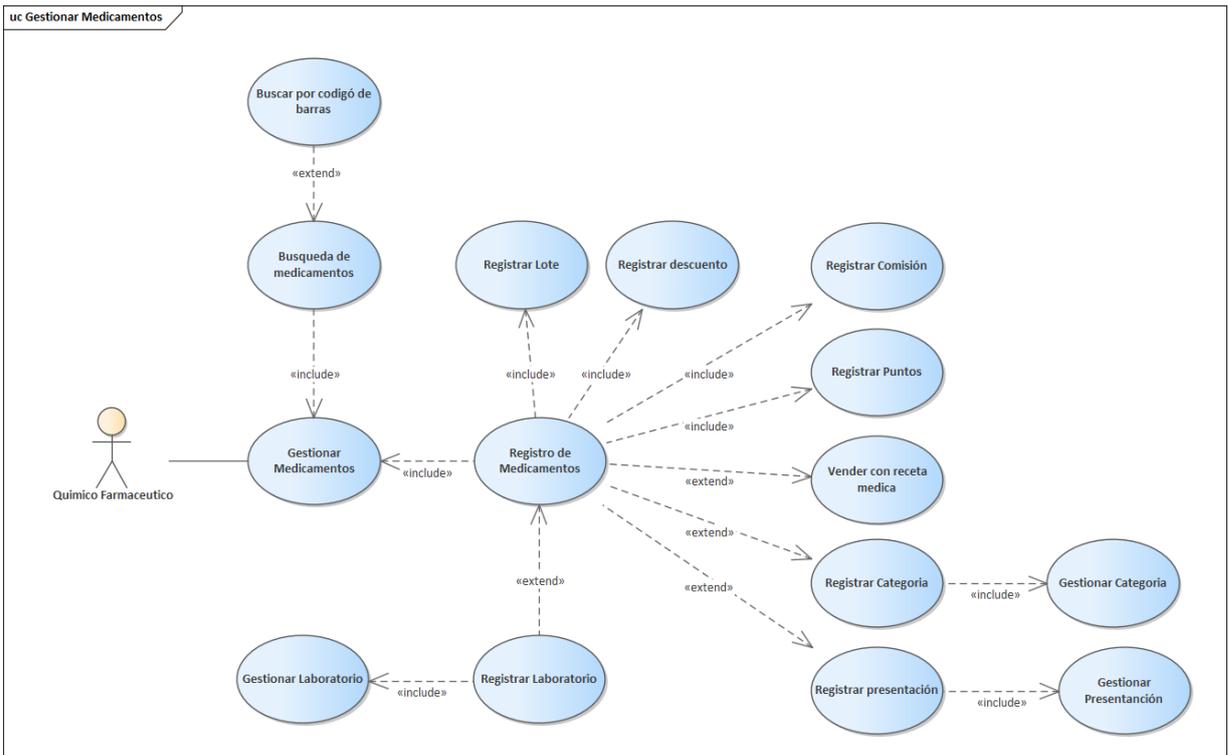


Figura 27. Caso de uso Gestionar Medicamentos
Fuente: Elaboración propia.

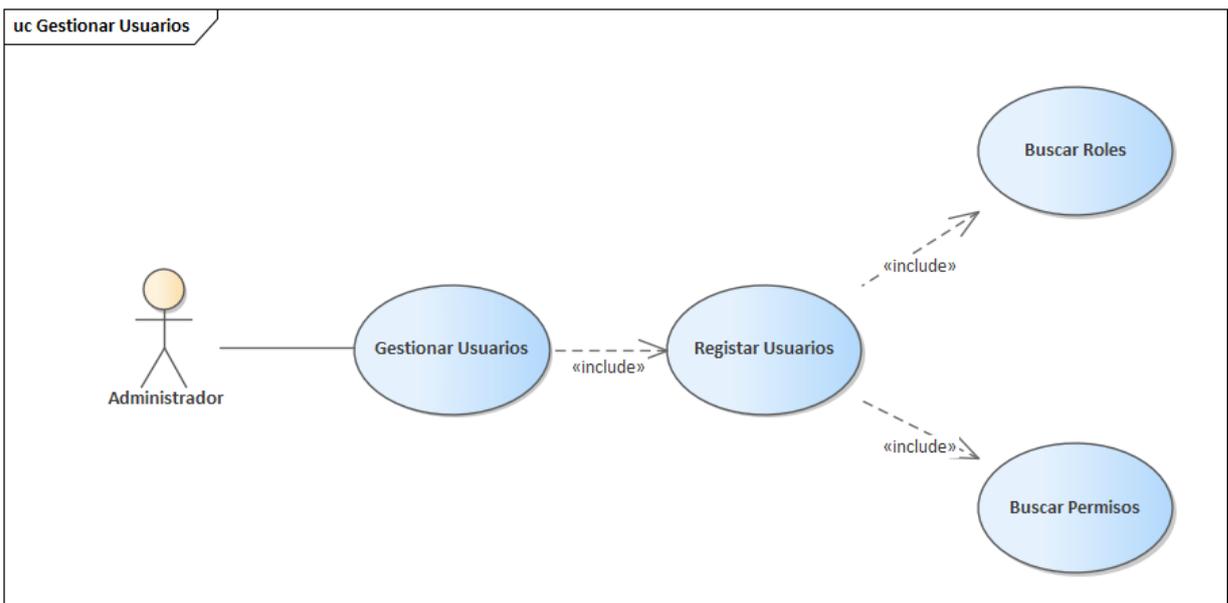


Figura 28. Casos de uso Gestionar Usuarios
Fuente: Elaboración propia.

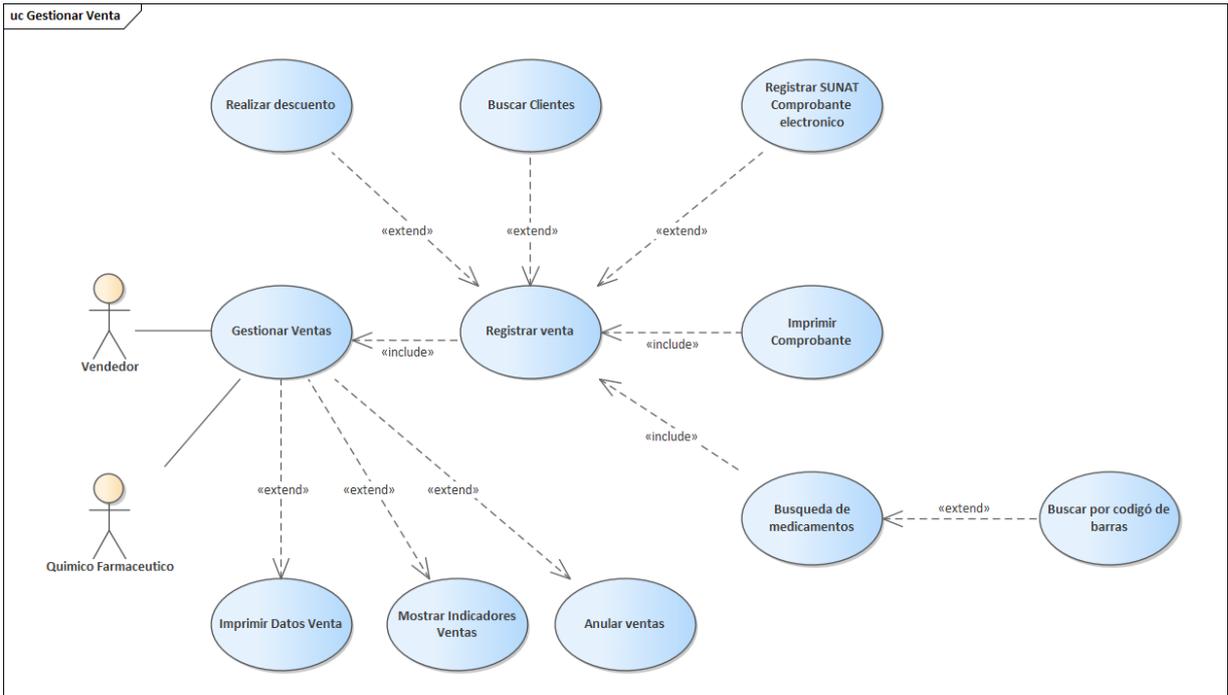


Figura 29. Caso de uso Gestionar Venta
Fuente: Elaboración propia.

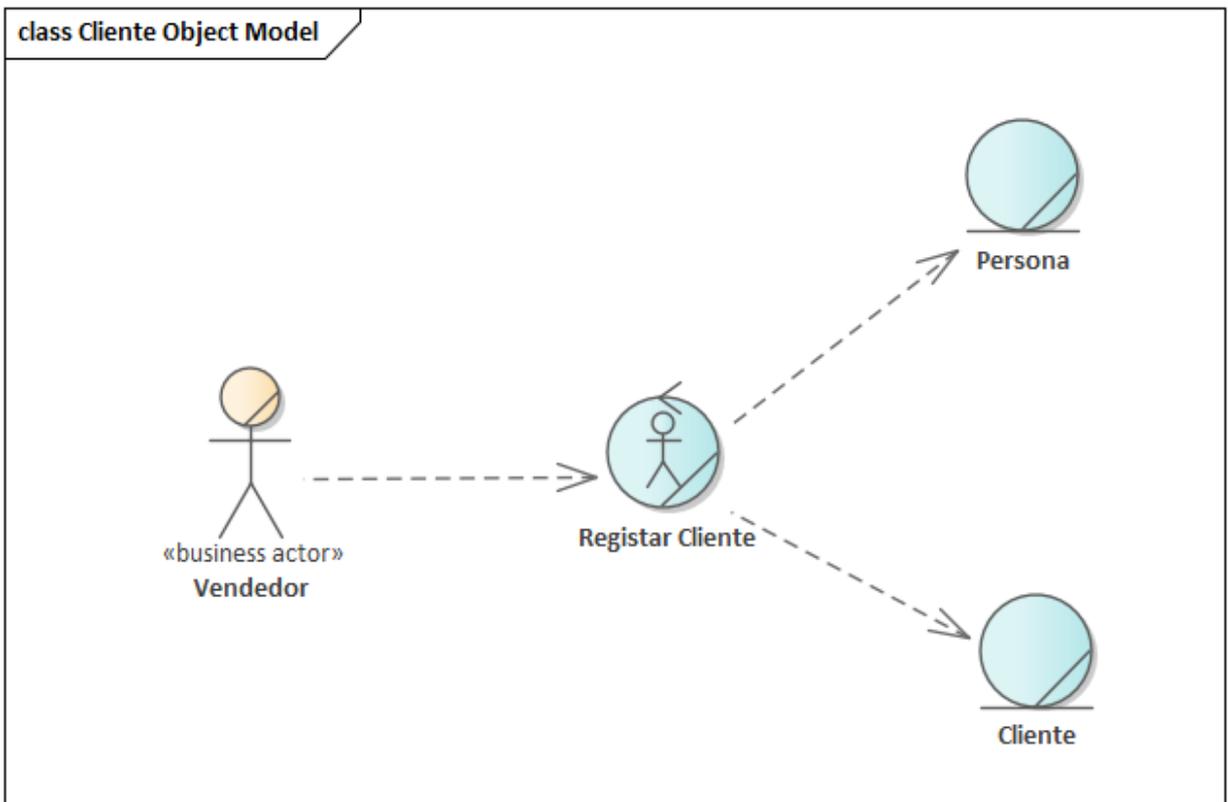


Figura 30. Diagrama de objetos - Cliente
Fuente: Elaboración propia.

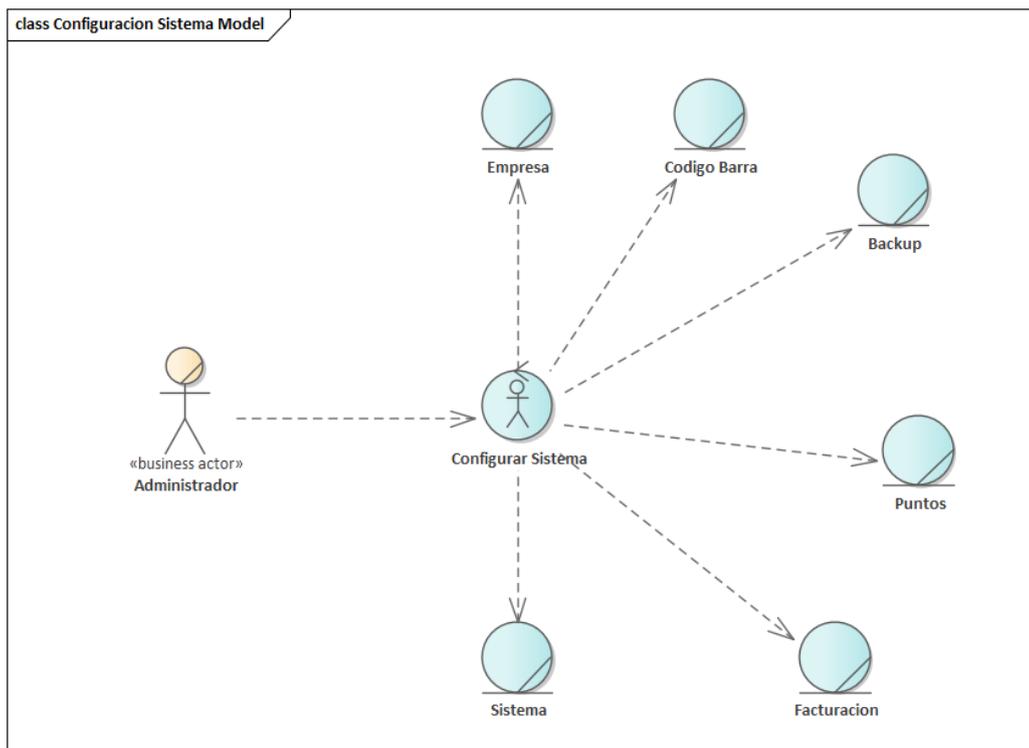


Figura 31. Diagrama de objetos - Configuración

Fuente: Elaboración propia.

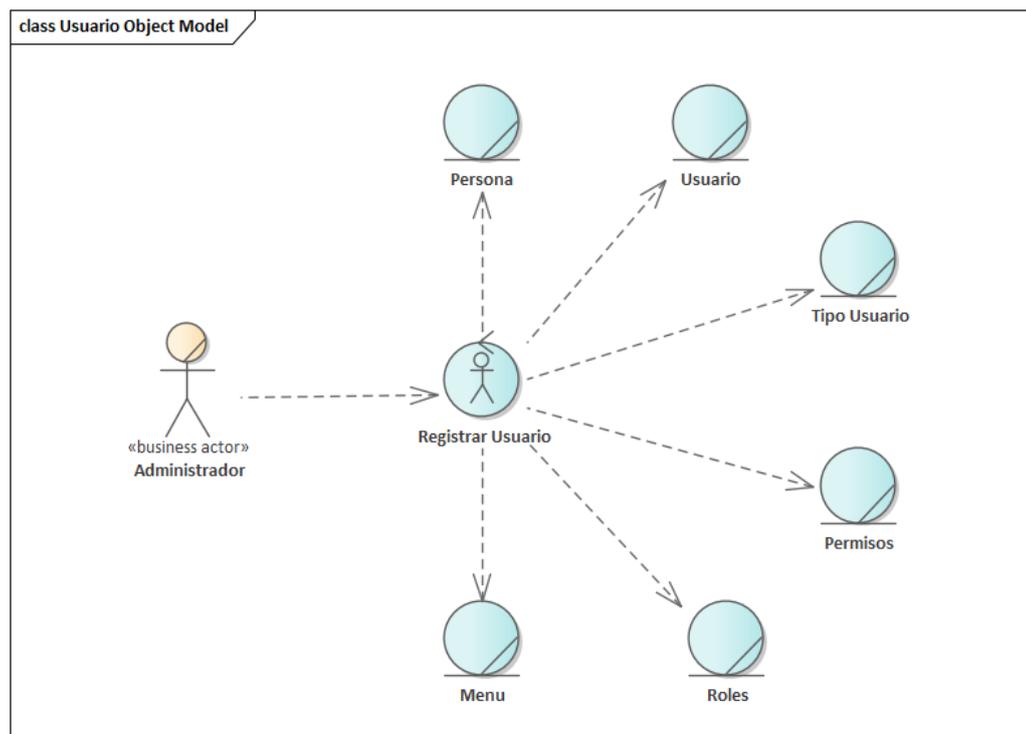


Figura 32. Diagrama de objetos - Usuario

Fuente: Elaboración propia.

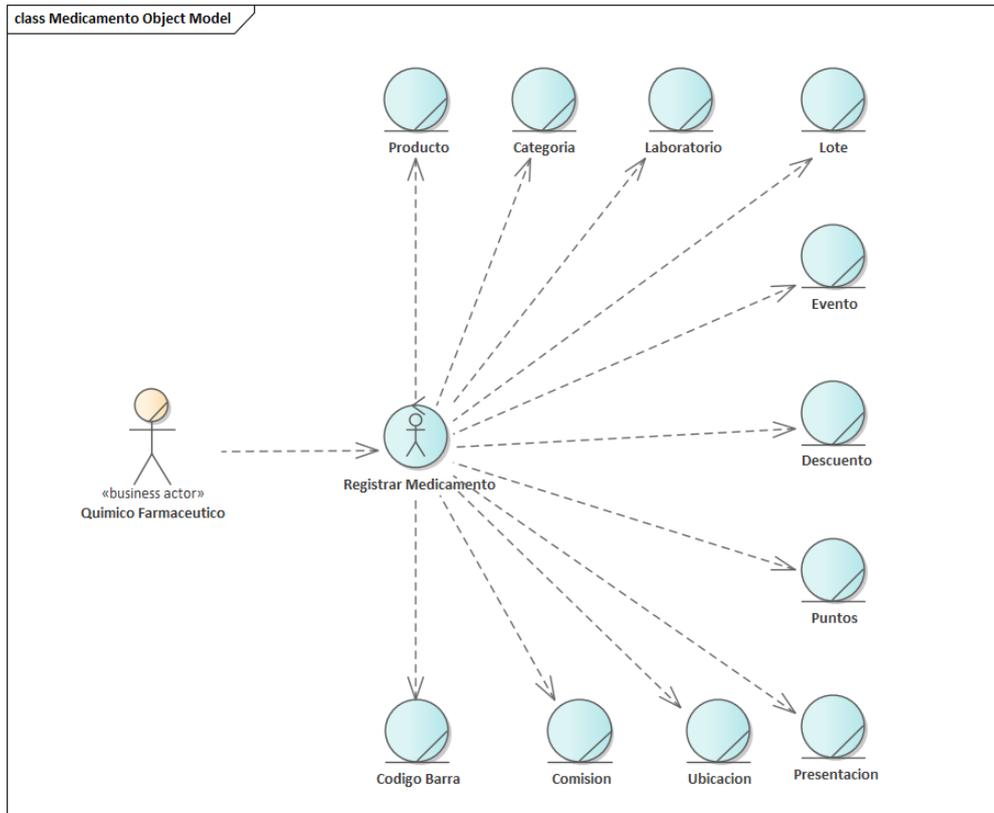


Figura 33. Diagrama de objetos - Medicamento
Fuente: Elaboración propia.

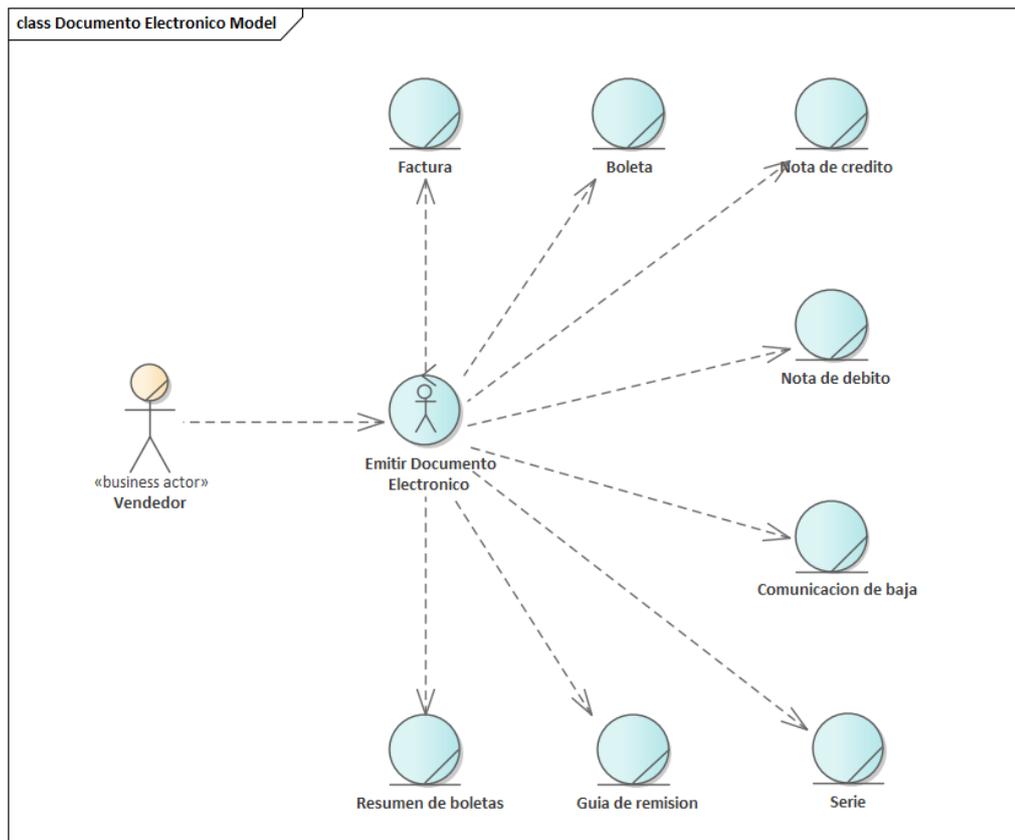


Figura 34. Diagrama de objetos - Documento Electrónico

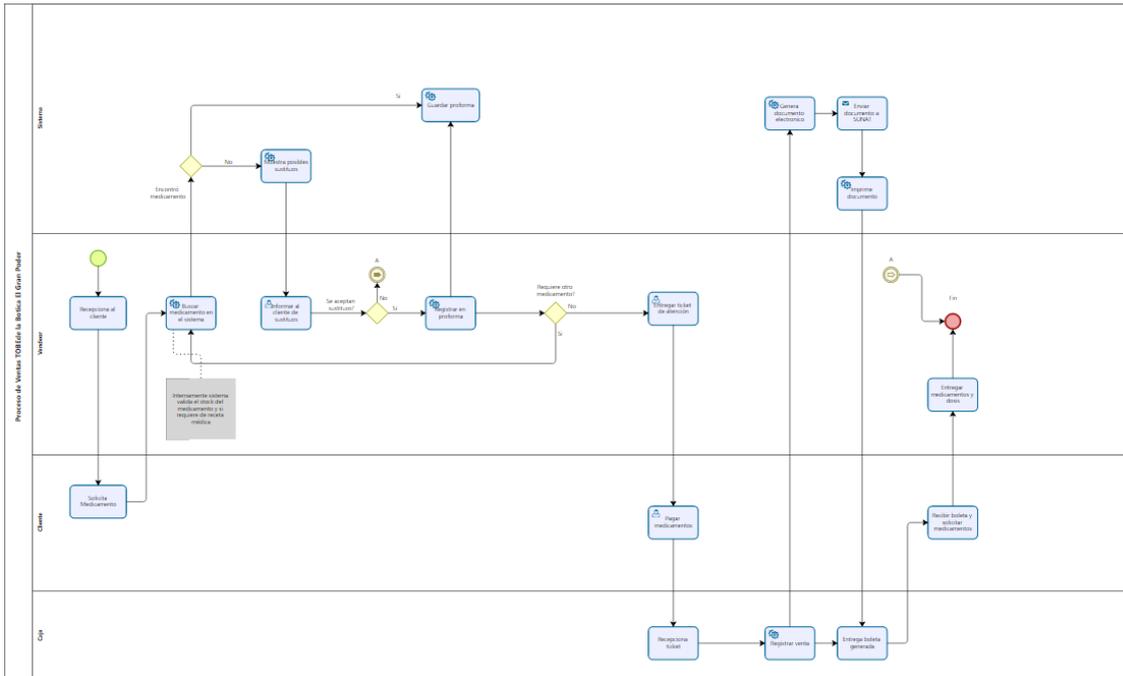


Figura 38. Diagrama de Actividades - TOBE del Proceso de Ventas

Fuente: Elaboración propia

4.9 Reglas de negocio

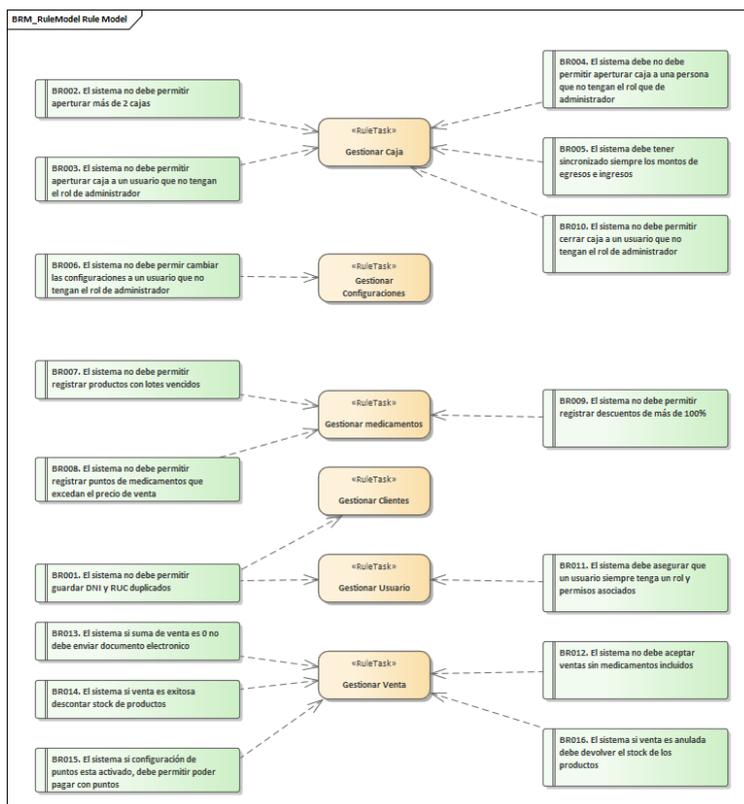


Figura 39. Diagrama de reglas de negocio

Fuente: Elaboración propia

CAPÍTULO V: DISEÑO DEL SISTEMA

5.1 Programación

```
<?php
defined('BASEPATH') or exit('No direct script access allowed');

class Ventas extends CI_Controller
{

    private $permisos;

    private $turnos;

    private $caja_abierta;

    private $caja_abierta_texto;

    public function __construct()
    {
        parent::__construct();
        $this->load->model("Usuarios_model");
        if (!$this->session->userdata("login")) {
            redirect(base_url() . "inicio");
        }
        $idacceso = $this->session->userdata("id");
        $ip_DB = $this->Usuarios_model->getUsuario($idacceso);
        if ($this->session->userdata("ip_actual") != $ip_DB->ip_actual) {
            $this->session->sess_destroy();
            redirect(base_url() . "inicio");
        }
        $this->load->helper('util');
        $this->load->model("Ventas_model");
        $this->load->model("Clientes_model");
    }
}
```

```

$this->load->model("Medico_model");
$this->load->model("Productos_model");
$this->load->model("Configuracion_model");
$this->load->model('Tipo_documento_model');
$this->load->model('Formas_de_pago_model');
$this->load->model('Forma_de_pago_mixto_model');
$this->load->model('Cuenta_cobrar_model');
$this->load->model('Kardex_model');
$this->load->model('Caja_model');
$this->load->model('Turno_model');
$this->load->model('Cajadetalle_model');
$this->load->model('Lote_model');
$this->load->model('Moneda_model');
$this->load->model('ConfiguracionPuntos_model');
$this->load->model('PuntosPersona_model');
$this->load->model('Eventos_model');
$this->load->model("Tipo_igv_model");
$this->load->model("Rol_model");
$this->permisos = $this->backend_lib->control();
$this->turnos = $this->Turno_model->listar_turnos();
$this->caja_abierta = $this->Caja_model->traer_Caja_aperturada(); // caja abierta actualmente

if ($this->caja_abierta != null) {
    $this->caja_abierta_texto = "Nro:" . $this->caja_abierta-
>idcaja . " " . "Monto de apertura S/." . $this->caja_abierta-
>monto_apertura . " - Fecha Apertura: " . $this->caja_abierta->fecha_apertura . " " . $this-
>caja_abierta->hora_apertura;
    } else {
        $this->caja_abierta_texto = "No hay una caja aperturada";
    }
    date_default_timezone_set('America/Lima');
}

public function index()
{
    $fecha_inicioTemporal = $this->input->post("fechainicio");
    $fecha_fin_temporal = $this->input->post("fechafin");
    $idPersona = $this->input->post("idPersonaVenta");
    $idTurno = $this->input->post("turno");
    $idTab = $this->input->post("nombreTab");
    $nroCajaAbierta = 0;
    $idCajaDetalle = 1; // por defecto la caja es 1
    $idAcceso = 0;
    $fecha_inicio = "";
    $fecha_fin = "";

    if (isset($idTab)) {
        $idTab = 0;
    }
}

```

```

}

if (!isset($fecha_inicioTemporal) || !isset($fecha_fin_temporal)) {
    $fecha_inicio = null;
    $fecha_fin = null;
} else {
    $fecha_inicio = obtener_fecha_hora(new DateTime($fecha_inicioTemporal), 00, 00, 00);
    $fecha_fin = obtener_fecha_hora(new DateTime($fecha_fin_temporal), 23, 59, 59);
}
}
$idRol = $this->session->userdata("rol"); // obtiene la variable de sesion

$dataEstados = array(
    VENDIDO
);

$dataEstados_baja = array(
    'DADO_BAJA'
);

if ($idRol == ROL_VENDEDOR) { // si rol es vendedor solo puede observar sus ventas
    $idPersona = $this->session->userdata("idPersona");
    $idAcceso = $this->Clientes_model->getAccesold($idPersona);
} elseif ($idRol == ROL_ADMINISTRADOR || $idRol == ROL_CAJERO) {
    if ($idPersona > 0) {
        $idAcceso = $this->Clientes_model->getAccesold($idPersona);
    }

    if (!isset($idTurno)) {
        if ($this->caja_abierta != null) {
            $idTurno = $this->caja_abierta->idTurno;
        }
    }

    if (!isset($idPersona) || $idPersona == 0) { // buscar todos
        $idCajaDetalle = 0;
    }
}

if ($this->caja_abierta != null) {
    $nroCajaAbierta = $this->caja_abierta->idcaja;

    $data = array(
        'permisos' => $this->permisos,
        'ventas_borrador' => $this->Ventas_model-
>getVentasFiltrosBorradorByCaja($idPersona, $fecha_inicio, $fecha_fin, $nroCajaAbierta), // contado
        'ventas' => $this->Ventas_model-
>getVentasFiltrosByCaja($idPersona, null, $dataEstados, $fecha_inicio, $fecha_fin, $idCajaDetalle, $idTu
rno, $nroCajaAbierta), // nuevo filtro solo caja abierta

```

```

        'ventas_credito' => $this->Ventas_model-
>getVentasFiltros($idPersona, CREDITO_VARIABLE, null, $fecha_inicio, $fecha_fin, $idCajaDetalle, $idTurno), // credito
        'ventas_anuladas' => $this->Ventas_model-
>getVentasAnuladasFiltrosByCaja($idPersona, null, $dataEstados_baja, $fecha_inicio, $fecha_fin, $idCajaDetalle, $idTurno, $nroCajaAbierta), // anuladas
        'idRol' => $idRol,
        'fechainicio' => $fecha_inicioTemporal,
        'fechafin' => $fecha_fin_temporal,
        'formas' => $this->Formas_de_pago_model->getFormas_de_pago(),

        'totalVentas' => $this->Ventas_model-
>getVentasByUsuarioDateByCaja($idAcceso, $dataEstados, $fecha_inicio, $fecha_fin, $idCajaDetalle, $idTurno, $nroCajaAbierta, SOLES),
        'totalDolares' => $this->Ventas_model-
>getVentasByUsuarioDateByCaja($idAcceso, $dataEstados, $fecha_inicio, $fecha_fin, $idCajaDetalle, $idTurno, $nroCajaAbierta, DOLARES),
        'totalEuros' => $this->Ventas_model-
>getVentasByUsuarioDateByCaja($idAcceso, $dataEstados, $fecha_inicio, $fecha_fin, $idCajaDetalle, $idTurno, $nroCajaAbierta, EUROS),

        'usuarioRolList' => $this->Clientes_model->getClienteByRol(),
        'idPersona' => $idPersona,
        'turnos' => $this->turnos,
        'idTurno' => $idTurno,
        'caja_abierta_texto' => $this->caja_abierta_texto,
        'idTab' => $idTab
    );
    $this->load->view("layouts/header");
    $this->load->view("layouts/aside");
    $this->load->view("admin/ventas/list", $data); // se envia a la lista el array data
    $this->load->view("layouts/footer");
} else {
    if ($idRol == ROL_VENDEDOR) {
        $this->abrir_alerta_vendedor();
    } else {
        $this->abrir_alerta();
    }
}
}
}

public function add()
{
    $idRol = $this->session->userdata("rol");
    date_default_timezone_set('America/Lima');
    $fecha_actual = date("Y-m-d");
    $configuracion = $this->Configuracion_model->getConfiguracion();
    $configuracion_puntos = $this->ConfiguracionPuntos_model->getConfiguracion(); //code piero
    $flagRangoFecha = false;

```

```

if ($configuracion_puntos != null) {
    $flagRangoFecha = $this->verifica_rango_fecha_c($configuracion_puntos-
>fecha_inicio, $configuracion_puntos->fecha_fin, date('Y-m-
d')); //validando si aun esta vigente el bono de puntos
}
$iva = $configuracion->iva;
$sistemaweb = $configuracion->publicado_web; //code
$precioventaeditar = $configuracion->editar_precio_venta; //code p
if ($flagRangoFecha) {
    $conf_puntos = $configuracion_puntos;
} else {
    $conf_puntos = 0;
}

$data = array(
    'tipocombrabantes' => $this->Tipo_documento_model->getComprobantesVenta(),
    'series' => $this->Tipo_documento_model->loadSeries(1),
    'clientes' => $this->Clientes_model->getClientes(),
    'moneda' => $this->Moneda_model->getMoneda(),
    'formas' => $this->Formas_de_pago_model->getFormas_de_pago(),
    'configuracion' => $this->Configuracion_model->getConfiguracion(),
    'iva' => $iva,
    'imprimir_directo' => $configuracion->imprimir_directo,

    'productos' => $this->Productos_model->getProductos(1),
    'today' => $fecha_actual,
    'idRol' => $idRol,
    'sistemaweb' => $sistemaweb, //code
    'configuracion_puntos' => $conf_puntos, //code
    'porcentaje_descuento_evento' => $this-
>obtenerEventoCliente_PorcentajeDescuento(1, true), //code
    'precioventaeditar' => $precioventaeditar, //code p
    'ct_precio_venta_credito' => $configuracion->ct_precio_venta_credito

);
if ($idRol == ROL_VENDEDOR) { // si rol es vendedor dejar pasar libremente
    $this->load->view("layouts/header");
    $this->load->view("layouts/aside");
    $this->load->view("admin/ventas/add", $data); // se envia a la lista el array data
    $this->load->view("layouts/footer");
} elseif ($idRol == ROL_ADMINISTRADOR || $idRol == ROL_CAJERO) { // si rol es administrador o caj
ero se debe validar que exista una caja abierta
    $caja_abierta = $this-
>caja_abierta; // comprueba si hay al menos una caja abierta, por ahora solo se maneja 1 caja
    if ($caja_abierta != null) {
        $this->load->view("layouts/header");
        $this->load->view("layouts/aside");
        $this->load->view("admin/ventas/add", $data); // se envia a la lista el array data
        $this->load->view("layouts/footer");
    }
}

```

```

    } else {
        $this->abrir_alerta();
    }
}
}

function obtenerEventoCliente_PorcentajeDescuento($idCliente, $return = false)
{
    $porcentaje_descuento_evento = 0;
    $now = new DateTime();
    $date = $now->format('Y-m-d');

    $evento = $this->Eventos_model->getEventoByIdCliente_ByDate($idCliente, $date);
    if ($evento == null || $evento == '') {
        $evento = $this->Eventos_model->getEventoByIdRol_ByDate(2, $date);
        if ($evento) {
            $porcentaje_descuento_evento = $evento->porcentaje_descuento;
        }
    } else {
        $porcentaje_descuento_evento = $evento->porcentaje_descuento;
    }

    if ($return) {
        return $porcentaje_descuento_evento;
    } else {
        echo $porcentaje_descuento_evento;
    }
}

/* validar el rango de fechas */
function verifica_rango_fecha_c($date_inicio, $date_fin, $hoy)
{
    $date_inicio = strtotime($date_inicio);
    $date_fin = strtotime($date_fin);
    $hoy = strtotime($hoy);
    if (($hoy >= $date_inicio) && ($hoy <= $date_fin)) {
        return true;
    } else {
        return false;
    }
}

public function getproductos()
{
    $valor = $this->input->post("valor");
    $productos = $this->Ventas_model->getproductos($valor);
    echo json_encode($productos);
}

```

```

public function view()
{
    $idventa = $this->input->post("id");
    $data = array(
        'datosVenta' => $this->Ventas_model->getVenta($idventa),
        'listaDetalleVenta' => $this->Ventas_model->getDetalle($idventa),
        'datosEmpresa' => $this->Configuracion_model->getConfiguracion()
    );
    $this->load->view("admin/ventas/view", $data);
}
public function view_puntos()
{
    $idventa = $this->input->post("id");
    $data = array(
        'datosVenta' => $this->Ventas_model->getVenta($idventa),
        'listaDetalleVenta' => $this->Ventas_model->getDetalle($idventa),
        'datosEmpresa' => $this->Configuracion_model->getConfiguracion()
    );
    $this->load->view("admin/ventas/view_puntos", $data);
}

public function view_motivoDeBaja()
{
    $idventa = $this->input->post("id");
    $data = array(
        'motivoDeBaja' => $this->Ventas_model->getVisualizarMotivoDeBaja($idventa)
    );
    $this->load->view("admin/ventas/view_motivoDeBaja", $data);
}

public function getProductoCodigo()
{
    $codigo = $this->input->post("codigo_p");
    $id = $this->input->post("id_p");
    if ($codigo != null) {
        $resultado = $this->Productos_model->getProductoByCodigo($codigo);
    } else {
        $resultado = $this->Productos_model->getProductoById($id);
    }
    $resultado->isInventario = $this->Productos_model->validarProductoInventario($resultado->id);
    if ($this->Lote_model->getLotesporIdProducto($id)) {
        $resultado->lotes = $this->Lote_model->getLotesporIdProducto($id);
    }
    echo json_encode($resultado);
}

public function getProductoCodigo_2()

```

```

{
    $codigo = $this->input->post("codigo_p");
    $id = $this->input->post("id_p");
    if ($codigo != null) {
        $resultado = $this->Productos_model->getProductoByCodigo($codigo);
    } else {
        $resultado = $this->Productos_model->getProductoById($id);
    }
    $resultado->isInventario = $this->Productos_model->validarProductoInventario($resultado->id);
    if ($this->Lote_model->getLotesporIdProducto2($id)) {
        $resultado->lotes = $this->Lote_model->getLotesporIdProducto2($id);
    }
    echo json_encode($resultado);
}

public function getTiposIgv()
{
    $tipos_igv = $this->Tipo_igv_model->get_tipos_igv();
    echo json_encode($tipos_igv);
}

public function consulta_ultimo_()
{
    $resultado = $this->Ventas_model->ultimoID();
    echo json_encode($resultado);
}

public function ver_venta_byID()
{
    $idventa = $this->input->post("idv");
    $resultado = $this->Ventas_model->getVenta($idventa);
    date_default_timezone_set('America/Lima');

    $caja_abierta = $this->Caja_model-
>traer_Caja_aperturada(); // comprueba si hay una caja abierta por ahora solo se trabaja con una caja a
bierta
    if ($caja_abierta != null) {
        $resultado->isCajaAbierta = true;
    } else {
        $resultado->isCajaAbierta = false;
    }
    echo json_encode($resultado);
}

public function guardar_venta()
{
    $configuracion = $this->Configuracion_model->getConfiguracion();
    $idventas = "";
    $numero = 0;

```

```

$numero_formateado = "";
$fecha = $this->input->post("fecha");
$subtotal = $this->input->post("subtotal");
$sub12 = $this->input->post("subtotal_12");
$Iva = $configuracion->iva;
$total = $this->input->post("total");
$total_exonerada = $this->input->post("exonerada");
$total_inafecta = $this->input->post("inafecta");
$total_gravada = $this->input->post("grabada");
$total_igv = $this->input->post("igvp_tip");
$total_gratuita = $this->input->post("gratuita");
$DescuentoPuntos = $this->input->post("DescuentoPuntosTotal");
$nroPuntos = $this->input->post("nroPuntos");
$nroSolesXPunto = $this->input->post("nroSolesXPunto");
$importeeditar = empty($this->input->post("importeeditar")) ? null : $this->input-
>post("importeeditar"); //1403/2021
$cambio = $this->input->post("cambio");
$idcomprobante = $this->input->post("comprobantes");
$id_serie = $this->input->post("id_serie");
$id_tipo_moneda = $this->input->post("id_moneda");
$tipo_cambio = $this->input->post("tcambio");
$idCliente = $this->input->post("idCliente");
$idacceso = $this->session->userdata("id");
$serie = $this->input->post("serie_desc");
$idproductos = $this->input->post("idproductos");
$precios = $this->input->post("precios");
$cantidades = $this->input->post("cantidades");
$tipoigvs = $this->input->post("tipoigvs");
$importes = empty($this->input->post("importeeditar")) ? $this->input->post("importes") : $this-
>input->post("importeeditar"); //code piero 15/03/2021
$importesaxu = $this->input->post("importes");
$lotes = $this->input->post("lotes");
$f_pago = $this->input->post("forma_pago");
$m_recibido = $this->input->post("recibido");

$cambio = $this->input->post("cambio");
$consulta_compra = $this->input->post("compra_por");
$id_medico = $this->input->post("idMedico");
$mixto = $this->input->post("montomixto");
if ($idproductos != 0) {
    if ($idCliente != 0) {
        $numero = $this->obtenerSecuencia($id_serie); // se obtiene la secuencia de base de datos
        $numero_formateado = str_pad($numero, 8, "0", STR_PAD_LEFT);

        if ($f_pago == 1) { // 1 = EFECTIVO, 2 = CREDITO, 3 = TARJETA
            $ingresos = $this->caja_abierta->ingresos;
            $data_caja = array(
                'ingresos' => $ingresos + $total
            );
        }
    }
}

```

```

        $this->Caja_model->update_caja($this->caja_abierta->idcaja, $data_caja);
        $this->caja_abierta = $this->Caja_model-
>traer_Caja_aperturada()); // actualizar el objeto caja abierta

        $ingresos_caja = $this->caja_abierta->ingresos;
        $egresos = $this->caja_abierta->egresos;
        $monto_aper = $this->caja_abierta->monto_apertura;

        if ($egresos != null || $egresos != 0) {
            $estimado = ($monto_aper + $ingresos_caja) - $egresos;
        } else {
            $estimado = ($monto_aper + $ingresos_caja);
        }

        $data_caja_update = array(
            'monto_estimado' => $estimado
        );
        $this->Caja_model->update_caja($this->caja_abierta->idcaja, $data_caja_update);
        $data["idCaja"] = $this->caja_abierta-
>idcaja; // se agrego el id de caja a la venta cuando la venta es al contado
    }
    if ($f_pago == FORMA_PAGO_CREDITO) { // sobreescribir el total si es al credito
        $cuotas = $this->input->post("mmcuota"); //monto cuota mensual
        $suma = 0;
        for ($x = 0; $x < sizeof($cuotas); ++$x) {
            $suma = $suma + $cuotas[$x];
        }
        $total = $suma;
    }
    $importeVentaTotal = 0;
    $comisionPorcentaje = 0;
    $comisionTotal = 0;

    for ($i = 0; $i < count($idproductos); $i++) {
        $producto_actual = $this->Productos_model->getProducto($idproductos[$i]);
        if ($producto_actual->excluye_comision < 1 || $producto_actual-
>excluye_comision == null) {
            $importeVenta = $importes[$i];
            $importeVentaTotal = $importeVentaTotal + $importeVenta;
        }
    }

    if ($importeVentaTotal > 0) {
        $idRol = $this->session->userdata("rol");
        $rol = $this->Rol_model->getRol($idRol);
        $comisionPorcentaje = $rol->comision;
        $comisionTotal = $importeVentaTotal * ($comisionPorcentaje / 100);
    }
}

```

```

$data = array(
    'fecha' => $fecha,
    'subtotal' => $subtotal,
    'subtotal12' => $sub12,
    'iva' => $iva,
    'descuento' => $DescuentoPuntos,
    'total' => $total,
    'recibido' => $m_recibido,
    'cambio' => $cambio,
    'tipo_venta_id' => $idcomprobante,
    'forma_pago_id' => $f_pago,
    'idCliente' => $idCliente,
    'idacceso' => $idacceso,
    'idCaja' => $this->caja_abierta->idcaja,
    'num_venta_id' => $numero_formateado,
    'serie' => $serie,
    'estado' => VENDIDO,
    'id_tipo_moneda' => $id_tipo_moneda,
    'tipo_cambio' => $tipo_cambio,
    'total_exonerada' => $total_exonerada,
    'total_inafecta' => $total_inafecta,
    'total_gravada' => $total_gravada,
    'total_igv' => $total_igv,
    'total_gratuita' => $total_gratuita,
    'tot_porventa' => $comisionPorcentaje,
    'total_com_rol' => $comisionTotal,
    'id_medico' => $id_medico
);

if ($this->Ventas_model->save($data) {
    $idventas = $this->Ventas_model->ultimoID();
    $this->updateCantidadSerie($id_serie, $numero);
    $numero_documento = $serie . "-" . $numero_formateado;
    $this->guarda_cuenta_por_cobrar($idCliente, $idventas, $total, $f_pago, $m_recibido);
    $this->guarda_detalle($idproductos, $idventas, $precios, $cantidades, $importes, $importesaxu, $importesed
    itar, $consulta_compra, $lotes, $tipoigvs, $fecha, $numero_documento);
    $this->guardar_detalle_venta_lote($lotes, $idventas);
    $dataPuntosPersona = array(
        "nroPuntos" => $nroPuntos,
        "nroSolesXPunto" => $nroSolesXPunto
    );

    $puntosPersona = $this->PuntosPersona_model->getPuntosPersonaByIdPersona($idCliente);
    if ($puntosPersona != null) {
        $this->PuntosPersona_model->update($puntosPersona-
    >id_puntos_persona, $dataPuntosPersona);
    }
}

```

```

        if ($mixto > 0) {
            $caja_abierta2 = $this->Caja_model->traer_Caja_aperturada();
            $idCaja = $caja_abierta2->idcaja;
            $this->guardar_mixto(1, $idventas, $idCaja, $m_recibido);
            $this->guardar_mixto(3, $idventas, $idCaja, $mixto);
        }
        $this->guardar_puntos($idCliente, $idventas);
        echo $idventas;
    }
}
}
}
}

```

```

protected function guardar_mixto($forma_de_pago, $id_venta, $idcaja, $monto)
{
    $data = array(
        'id_forma_de_pago' => $forma_de_pago,
        'id_venta' => $id_venta,
        'idCaja' => $idcaja,
        'monto' => $monto
    );
    $this->Forma_de_pago_mixto_model->store($data);
}

```

```

protected function guardar_puntos($idCliente, $idventa)
{
    $venta = $this->Ventas_model->getVenta($idventa);
    $config_puntos = $this->ConfiguracionPuntos_model->getConfiguracion();
    if ($config_puntos != null) {
        $flagRangoFecha = $this->verifica_rango_fecha($config_puntos->fecha_inicio, $config_puntos->fecha_fin, date('Y-m-d'));
    }
}

```

```

if ($flagRangoFecha && $config_puntos->estado == 1) {
    $monto_venta = $venta->total;
    $monto_por_punto = $config_puntos->monto_por_punto;
    $monto_por_punto_dinero = $config_puntos->monto_por_punto_dinero;
    $puntosPersona = $this->PuntosPersona_model->getPuntosPersonaByIdPersona($idCliente);
    $totalPuntos = $monto_venta / $monto_por_punto;
    $totalPuntosDinero = $totalPuntos / $monto_por_punto_dinero;

    if ($puntosPersona != null) {
        $puntoActual = $puntosPersona->nroPuntos;
        $puntoDineroActual = $puntosPersona->nroSolesXPunto;
        $puntoNuevo = $puntoActual + $totalPuntos;
        $puntoDineroNuevo = $puntoDineroActual + $totalPuntosDinero;
        $dataupdate = array(
            'nroPuntos' => $puntoNuevo,

```

```

        'nroSolesxPunto' => $puntoDineroNuevo
    );
    $dataPuntos = array(
        'nroPuntos' => $puntoNuevo,
        'puntoGanados' => $totalPuntos
    );
    $this->Ventas_model->update_venta_puntos($idventa, $dataPuntos);
    $this->PuntosPersona_model->update($puntosPersona->id_puntos_persona, $dataupdate);
} else {
    $dataPuntos = array(
        'nroPuntos' => $totalPuntos,
        'puntoGanados' => $totalPuntos
    );
    $this->Ventas_model->update_venta_puntos($idventa, $dataPuntos);
    $datainsert = array(
        'id_persona' => $idCliente,
        'nroPuntos' => $totalPuntos,
        'nroSolesxPunto' => $totalPuntosDinero
    );
    $this->PuntosPersona_model->save($datainsert);
}
}
}
}
}

```

```

function verifica_rango_fecha($date_inicio, $date_fin, $hoy)
{
    $date_inicio = strtotime($date_inicio);
    $date_fin = strtotime($date_fin);
    $hoy = strtotime($hoy);
    if (($hoy >= $date_inicio) && ($hoy <= $date_fin)) {
        return true;
    } else {
        return false;
    }
}

```

```

public function guarda_cuenta_por_cobrar($idcliente, $idventa, $total, $f_pago, $monto_rec)
{
    $saldo_adeudar = 0;
    if ($f_pago == FORMA_PAGO_CREDITO) { // si la forma de pago es credito agregar las cuotas
        $cuotas = $this->input->post("mmcuota"); //monto cuota mensual
        $intereses = $this->input->post("imcuota"); //interes calculado mensual
        $fechaCuotas = $this->input->post("fmcuota"); //fecha a pagar me por mes

        $suma = 0;
        for ($x = 0; $x < sizeof($cuotas); ++$x) {
            $suma = $suma + $cuotas[$x];
            $data = array(

```

```

        'id_venta' => $idventa,
        'id_caja' => 0,
        'total' => $cuotas[$x],
        'interes' => $intereses[$x],
        'cuota' => $x + 1,
        'fecha' => date('Y-m-d', strtotime($fechaCuotas[$x])),
        'estado' => 'p'
    );
    $this->Ventas_model->save_venta_credito($data);
}
$saldo_adeudar = $suma;
$array_cta_cobrar = array(
    'ide_cliente' => $idcliente,
    'ide_venta' => $idventa,
    'monto' => $saldo_adeudar,
    'saldo' => $saldo_adeudar,
    'estado' => "A"
);
return $this->Cuenta_cobrar_model->guarda_cuenta_cobrar($array_cta_cobrar);
}
}
}

```

protected function guarda_detalle(\$productos, \$idventa, \$precios, \$cantidades, \$importes, \$importes axu, \$importeseditar, \$comprapor, \$lotes, \$tipoigvs, \$fecha, \$numero_documento)

```

{
    $puntaje_medico_total = 0;
    $venta = $this->Ventas_model->getVentaDatos($idventa);
    $conf_pvc = $this->Configuracion_model->getConfiguracion()->ct_precio_venta_credito;

    if ($venta->id_medico > 0) {
        $configuracion = $this->Configuracion_model->getConfiguracion();
    }
    if ($importeseditar != null) {
        $configuracion = $this->Configuracion_model->getConfiguracion();
    }

    for ($i = 0; $i < count($productos); $i++) {
        $producto_actual = $this->Productos_model->getProducto($productos[$i]);
        $destino_in = "VENTA";
        $stock = $producto_actual->stock;
        $cantidad_final = 0;
        $idacceso = $this->session->userdata("id");
        $ventaBonificacion = 'VENTA POR BONIFICACION';
        $ventaTransferenciaGratuita = 'VENTA POR TRANSFERENCIA GRATUITA';
        $ventaOperacionGratuita = 'VENTA OPERACION GRATUITA';
        if ($tipoigvs[$i] == 15) {
            $destino = $ventaBonificacion;
        } else if ($tipoigvs[$i] == 16) {
            $destino = $ventaTransferenciaGratuita;
        }
    }
}

```

```

    } else if ($tipoigvs[$i] == 11 || $tipoigvs[$i] == 12 || $tipoigvs[$i] == 13 || $tipoigvs[$i] == 14 || $t
    ipoigvs[$i] == 17 || $tipoigvs[$i] == 31 || $tipoigvs[$i] == 32 || $tipoigvs[$i] == 33 || $tipoigvs[$i] == 34
    || $tipoigvs[$i] == 35 || $tipoigvs[$i] == 36) {
        $destino = $ventaOperacionGratuita;
    } else {
        $destino = $destino_in;
    }
    $data_kardex = array(
        'id_producto' => $productos[$i],
        'id_venta' => $idventa,
        'idacceso' => $idacceso,
        'tipo' => "s",
        'pu' => $precios[$i],
        'stock' => $stock,
        'fecha_emision' => $fecha,
        'numero_documento' => $numero_documento,
        'destino' => $destino
    );
    $und_caja = $producto_actual->unidades_caja;
    $und_blister = $producto_actual->cblister;
    $cantidades[$i] = intval($cantidades[$i]);
    if ($comprapor[$i] == "UNIDAD") {
        $cantidad_final = $cantidades[$i];
    } else if ($comprapor[$i] == "UNIDAD2") {
        $cantidad_final = ($conf_pvc == 1) ? $cantidades[$i] : 2 * $cantidades[$i]; //configuracin de ven
    ta a credito
    } else if ($comprapor[$i] == "UNIDAD3") {
        $cantidad_final = 3 * $cantidades[$i];
    } else if ($comprapor[$i] == "CAJA") {
        $cantidad_final = $und_caja * $cantidades[$i];
    } else if ($comprapor[$i] == "BLISTER") {
        $cantidad_final = $und_blister * $cantidades[$i];
    }
    $data_kardex["cantidad"] = $cantidad_final;
    $data_kardex["stock"] = intval($stock) - intval($cantidad_final);
    $this->Kardex_model->save_kardex($data_kardex);

    if ($cantidades[$i] <= $producto_actual->stock) {
        $und_caja = $producto_actual->unidades_caja;
        $und_blister = $producto_actual->cblister;
        $stock_caja = 0;
        if ($comprapor[$i] == "UNIDAD") {
            $stock = $cantidades[$i];
        } else if ($comprapor[$i] == "UNIDAD2") {
            $stock = ($conf_pvc == 1) ? $cantidades[$i] : 2 * $cantidades[$i]; //configuracino de venta a c
    redito
        } else if ($comprapor[$i] == "UNIDAD3") {
            $stock = 3 * $cantidades[$i];
        }
    }

```

```

} else if ($comprapor[$i] == "CAJA") {
    $stock = $und_caja * $cantidades[$i];
    $stock_caja = $cantidades[$i];
} else if ($comprapor[$i] == "BLISTER") {
    $stock = $und_blister * $cantidades[$i];
}
}
if ($producto_actual->por_com > 0) {
    $total_comision = $producto_actual->por_com * $importes[$i] / 100;
} else if ($producto_actual->efectivo_com > 0) {
    $total_comision = $producto_actual->efectivo_com * $importes[$i];
} else {
    $total_comision = $producto_actual->por_com * $importes[$i] / 100;
}

if (isset($configuracion) && $configuracion->editar_precio_venta == 1) {
    $descuento_item = $importesaxu[$i] - $importeseditar[$i];
    $descuento_detalle = $descuento_item < 0 ? 0.00 : $descuento_item;
} else {
    $descuento_item = NULL;
    $descuento_detalle = 0.00;
}

$data = array(
    'producto_id' => $productos[$i],
    'id_tipo_igv' => $tipoigvs[$i],
    'venta_id' => $idventa,
    'precio' => $precios[$i],
    'tipo_pvp' => $comprapor[$i],
    'cantidad' => $cantidades[$i],
    'importe' => $importes[$i],
    'descuento_detalle' => $descuento_detalle,
    'cantidad_unitaria' => $stock,
    'por_com' => $producto_actual->por_com,
    'tot_com' => $total_comision
);

error_log('validar configuracion');
if (isset($configuracion) && $configuracion->puntaje_medico == 1 && $tipoigvs[$i] == '10') {
    error_log('realizar calculo');
    $detalle_puntaje = $stock * $producto_actual->puntaje_medico;
    $data["detalle_puntaje"] = $detalle_puntaje;
    $puntaje_medico_total += $detalle_puntaje;
}

$this->updateProducto($productos[$i], $stock, $stock_caja);

if ($lotes[$i]) {
    $lote_array = json_decode($lotes[$i]);

```

```

$total_unidades = 0;
foreach ($lote_array as $mydata) {
    $total_unidades += $mydata->cantidadPresentacion;
}

if ($total_unidades == $stock) {
    foreach ($lote_array as $mydata) {
        $lotebd = $this->Lote_model->getLoteIdLote($mydata->idlote);
        $nuevo_stock = floatval($lotebd->stock_lote) - floatval($mydata-
>cantidadPresentacion);

        if ($nuevo_stock >= 0) {
            $estado = 'A';
            $data_lote = array(
                'stock_lote' => $nuevo_stock,
                'estado_lote' => $estado
            );
            $this->Lote_model->update_lote($mydata->idlote, $data_lote);
        }
    }
} else {
    $listaLotes = $this->Lote_model->getLotesporIdProducto($productos[$i]);
    $stock_registrado = $stock;
    foreach ($listaLotes as $lote) {
        if ($stock_registrado > 0 && $lote["stock_lote"] > 0) {
            $restaLote = $stock_registrado - $lote["stock_lote"];
            if ($restaLote >= 0) {
                $data_lote = array(
                    'stock_lote' => 0
                );
            } else {
                $data_lote = array(
                    'stock_lote' => ($lote["stock_lote"] - $stock_registrado),
                );
            }
            $stock_registrado = $restaLote;
            $this->Lote_model->update_lote($lote["idlote"], $data_lote);
        }
    }
}
}
$this->Ventas_model->save_detalle($data);
}

error_log('total_puntaje_medico: ' . $puntaje_medico_total);
if ($puntaje_medico_total > 0) {
    $total = $this->input->post("total");
    $this->Ventas_model-
>update_venta($idventa, array('puntaje_medico_total' => $puntaje_medico_total));
}

```

```

        $this->Ventas_model->update_venta($idventa, array('total' => $total));
    }
}
return true;
}

protected function guardar_detalle_venta_lote($lotes, $idventa)
{
    $detalle_venta = $this->Ventas_model->getDetalle($idventa);
    for ($i = 0; $i < count($detalle_venta); $i++) {
        $lote_array2 = json_decode($lotes[$i]);

        if ($lote_array2 != null) {
            foreach ($lote_array2 as $mydata2) {
                $data_lotedetalle = array(
                    'id_detalle_v' => $detalle_venta[$i]->id_detalle_v,
                    'id_producto' => $detalle_venta[$i]->producto_id,
                    'id_lote' => $mydata2->idlote,
                    'cantidad' => $mydata2->cantidadPresentacion
                );
                $this->Ventas_model->save_detalle_venta_lote($data_lotedetalle);
            }
        }
    }
}

public function updateCantidadSerie($idSerie, $cantidad)
{
    $data = array(
        'cantidad' => $cantidad
    );
    $this->Ventas_model->updateCantidadSerie($idSerie, $data);
}

public function obtenerSecuencia($id_serie)
{
    $cantidad = 0;
    $serie = $this->Tipo_documento_model->getNumeroSerieByIdSerie($id_serie);
    if ($serie->numero_inicial > $serie->cantidad) {
        $cantidad = $serie->numero_inicial;
    } else {
        $cantidad = $serie->cantidad + 1;
    }
    return $cantidad;
}

protected function updateProducto($idproducto, $cantidad, $cantidad_caja)
{

```

```

$productoActual = $this->Productos_model->getProducto($idproducto);
if ($cantidad_caja != null) {
    $stock_caja = $productoActual->stock / $productoActual->unidades_caja;
    $stock_caja_c = floor($stock_caja);
    $data = array(
        'stock' => $productoActual->stock - $cantidad,
        'stock_caja' => $stock_caja_c - $cantidad_caja
    );
} else {
    $data = array(
        'stock' => $productoActual->stock - $cantidad
    );
}
$this->Productos_model->update($idproducto, $data);
}

public function consulta_ultimo_id()
{
    $resultado = $this->Ventas_model->recuperar_id();
    echo json_encode($resultado);
}

public function DardeBaja()
{
    $idventa = $this->input->post("idv");
    $motivo = $this->input->post("motivo");

    $this->caja_abierta = $this->Caja_model-
>traer_Caja_aperturada(); // comprueba si hay una caja abierta el dia de hoy con estado 1 si viene TRUE
esque ya hay una caja abierta con esa fecha con ese usuario y el estado esta activo
    if ($this->caja_abierta != null) {
        $detalle_venta = $this->Ventas_model->getDetalle($idventa);
        $fecha_actual = date("Y-m-d");
        $hora_actual = date("H:i:s");
        $data_table = array(
            'fecha' => $fecha_actual,
            'hora' => $hora_actual,
            'idacceso' => $this->session->userdata('id'),
            'idventa' => $idventa,
            'motivo' => $motivo
        );

        $data_venta = array(
            'estado' => 0,
            'aceptado' => 'Anulada'
        );
        $this->Ventas_model->update_venta($idventa, $data_venta);
        $venta_datos = $this->Ventas_model->getVentaDatos($idventa);
        $documento = $venta_datos->serie . "-" . $venta_datos->num_venta_id;
    }
}

```

```

if ($detalle_venta != "") {
    foreach ($detalle_venta as $d) {
        $cantidadFinal = $d->cantidad_unitaria;
        $idproducto = "";
        $idproducto = $d->producto_id;
        $cantidad = $d->cantidad;
        $productoActual = $this->Productos_model->getProducto($idproducto);
        $tipo_pvp = $d->tipo_pvp;

        // devolver las unidades por stock
        $data_prod = array(
            'stock' => $productoActual->stock + $cantidadFinal
        );
        // devolver las unidades caja si el pvp es caja
        if ($tipo_pvp == "CAJA") {
            $data_prod['stock_caja'] = $productoActual->stock_caja + $cantidad;
        }
        $stock = $productoActual->stock + $cantidadFinal;
        $this->Productos_model-
>update($idproducto, $data_prod); // actualiza el stock del producto

        $detalleVentaLote = $this->Ventas_model->getDetalle_Venta_Lote($d->id_detalle_v);

        foreach ($detalleVentaLote as $dlv) {
            $lote = $this->Lote_model->getLotelIdLote($dlv->id_lote);
            $data_lote = array(
                'stock_lote' => $lote->stock_lote + $dlv->cantidad
            );
            $this->Lote_model->update_lote3($dlv-
>id_lote, $data_lote); // actualiza el stock del lote del producto
        }
        $this->Kardex_model-
>kardex_movimiento($idproducto, $cantidadFinal, $stock, INGRESO, $d->precio, $d-
>venta_id, $documento, "ANULACION VENTA"); // actualiza movimiento de kardex
    }
    // si la venta es al credito - dar de baja en cuentas por cobrar.
    if ($venta_datos->forma_pago_id == 2) {
        $cuentaCobrar = $this->Cuenta_cobrar_model->getCuentaPorCobrarByIdVenta($idventa);
        $cuentaCobrar->estado = 'E'; // Estado eliminado
        $devolverMonto = $cuentaCobrar->monto - $cuentaCobrar->saldo;
        $this->Caja_model-
>actualiza_caja_movimiento($devolverMonto, EGRESO); // actualizar la caja si es que hay monto por de
volver
        $this->Cuenta_cobrar_model->update($cuentaCobrar-
>ide_cta_cobrar, $cuentaCobrar); // eliminamos la cta por cobrar
    } else {
        // Obtener la venta
    }
}

```

```

        $this->Caja_model->actualiza_caja_movimiento($venta_datos-
>total, EGRESO); // actualizar la caja
    }
    echo $this->Ventas_model->Venta_Dada_de_Baja($data_table);
} else {
    echo false;
}
} else {
    $this->abrir_alerta();
}
}
public function DardeBajaTicket($idventa)
{
    $motivo = $this->input->post("motivo");
    $this->caja_abierta = $this->Caja_model-
>traer_Caja_aperturada(); // comprueba si hay una caja abierta el dia de hoy con estado 1 si viene TRUE
esque ya hay una caja abierta con esa fecha con ese usuario y el estado esta activo
    if ($this->caja_abierta != null) {
        $detalle_venta = $this->Ventas_model->getDetalle($idventa);
        $fecha_actual = date("Y-m-d");
        $hora_actual = date("H:i:s"); // 17:16:18
        $data_table = array(
            'fecha' => $fecha_actual,
            'hora' => $hora_actual,
            'idacceso' => $this->session->userdata('id'),
            'idventa' => $idventa,
            'motivo' => $motivo
        );

        $data_venta = array(
            'estado' => 0,
            'aceptado' => 'Anulada'
        );
        $this->Ventas_model->update_venta($idventa, $data_venta);
        $venta_datos = $this->Ventas_model->getVentaDatos($idventa);
        $documento = $venta_datos->serie . "-" . $venta_datos->num_venta_id;

        if ($detalle_venta != "") {
            foreach ($detalle_venta as $d) {
                $cantidadFinal = $d->cantidad_unitaria;
                $idproducto = "";
                $idproducto = $d->producto_id;
                $cantidad = $d->cantidad;
                $productoActual = $this->Productos_model->getProducto($idproducto);
                $tipo_pvp = $d->tipo_pvp;

                // devolver las unidades por stock
                $data_prod = array(
                    'stock' => $productoActual->stock + $cantidadFinal
                );
            }
        }
    }
}

```

```

);
// devolver las unidades caja si el pvp es caja
if ($tipo_pvp == "CAJA") {
    $data_prod['stock_caja'] = $productoActual->stock_caja + $cantidad;
}
$stock = $productoActual->stock + $cantidadFinal;
$this->Productos_model-
>update($idproducto, $data_prod); // actualiza el stock del producto
    $detalleVentaLote = $this->Ventas_model->getDetalle_Venta_Lote($d->id_detalle_v);

    foreach ($detalleVentaLote as $dlv) {
        $lote = $this->Lote_model->getLoteldLote($dlv->id_lote);
        $data_lote = array(
            'stock_lote' => $lote->stock_lote + $dlv->cantidad
        );
        $this->Lote_model->update_lote3($dlv-
>id_lote, $data_lote); // actualiza el stock del lote del producto
    }
    $this->Kardex_model-
>kardex_movimiento($idproducto, $cantidadFinal, $stock, INGRESO, $d->precio, $d-
>venta_id, $documento, "ANULACION VENTA"); // actualiza movimiento de kardex
    }
    if ($venta_datos->forma_pago_id == 2) {
        $cuentaCobrar = $this->Cuenta_cobrar_model->getCuentaPorCobrarByIdVenta($idventa);
        $cuentaCobrar->estado = 'E'; // Estado eliminado
        $devolverMonto = $cuentaCobrar->monto - $cuentaCobrar->saldo;
        $this->Caja_model-
>actualiza_caja_movimiento($devolverMonto, EGRESO); // actualizar la caja si es que hay monto por de
volver
        $this->Cuenta_cobrar_model->update($cuentaCobrar-
>ide_cta_cobrar, $cuentaCobrar); // eliminamos la cta por cobrar
    } else {
        $this->Caja_model->actualiza_caja_movimiento($venta_datos-
>total, EGRESO); // actualizar la caja
    }
    if ($this->Ventas_model->Venta_Dada_de_Baja($data_table)) :
        $this->session->set_flashdata('exito', 'EL TICKET HA SIDO ANULADO ');
        redirect(base_url() . "reportes/ventas/por_ventas");
    endif;
} else {
    $this->session->set_flashdata('error', 'NO SE ANULO EL TICKET ');
    redirect(base_url() . "reportes/ventas/por_ventas");
}
} else {
    $this->abrir_alerta();
}
}
}

public function guardar_venta_borrador()

```

```

{
    $configuracion = $this->Configuracion_model->getConfiguracion();
    $idventas = "";
    $numero = 0; // numero de la secuencia de ventas
    $numero_formateado = "";
    $fecha = $this->input->post("fecha");
    $subtotal = $this->input->post("subtotal");
    $sub12 = $this->input->post("subtotal_12");
    $iva = $configuracion->iva;
    $descuento = $this->input->post("descuento");
    $total = $this->input->post("total");
    $cambio = $this->input->post("cambio");
    $idcomprobante = $this->input->post("idcomprobante");
    $id_serie = $this->input->post("id_serie");
    $total_exonerada = $this->input->post("exonerada");
    $total_inafecta = $this->input->post("inafecta");
    $total_gravada = $this->input->post("grabada");
    $total_igv = $this->input->post("igvp_tip");
    $total_gratuita = $this->input->post("gratuita");
    $importeseditar = empty($this->input->post("importeeditar")) ? null : $this->input-
>post("importeeditar"); //1403/2021

    $idCliente = $this->input->post("idCliente");
    $idacceso = $this->session->userdata("id");
    $serie = $this->input->post("serie_desc");
    $tipoigvs = $this->input->post("tipoigvs");
    $idproductos = $this->input->post("idproductos");
    $precios = $this->input->post("precios");
    $cantidades = $this->input->post("cantidades");
    $importes = empty($this->input->post("importeeditar")) ? $this->input->post("importes") : $this-
>input->post("importeeditar"); //code piero 15/03/2021
    $importesaxu = $this->input->post("importes");
    $f_pago = $this->input->post("forma_pago");
    $m_recibido = $this->input->post("recibido");
    $lotes = $this->input->post("lotes");
    $cambio = $this->input->post("cambio");

    $id_tipo_moneda = $this->input->post("id_moneda");
    $tipo_cambio = $this->input->post("tcambio");
    $consulta_compra = $this->input->post("compra_por");

    if ($idproductos != 0) {
        if ($idCliente != 0) {
            $numero = $this->obtenerSecuencia($id_serie); // se obtiene la secuencia de base de datos
            $numero_formateado = str_pad($numero, 8, "0", STR_PAD_LEFT);
            $data = array(
                'fecha' => $fecha,
                'subtotal' => $subtotal,
                'subtotal12' => $sub12,

```

```

'iva' => $iva,
'descuento' => $descuento,
'total' => $total,
'recibido' => $m_recibido,
'cambio' => $cambio,
'tipo_venta_id' => $idcomprobante,
'forma_pago_id' => $f_pago,
'idCliente' => $idCliente,
'idacceso' => $idacceso,
'num_venta_id' => $numero_formateado,
'serie' => $serie,
'estado' => VENDIDO Borrador,
'id_tipo_moneda' => $id_tipo_moneda,
'tipo_cambio' => $tipo_cambio,
'total_exonerada' => $total_exonerada,
'total_inafecta' => $total_inafecta,
'total_gravada' => $total_gravada,
'total_igv' => $total_igv,
'total_gratuita' => $total_gratuita
);

$data["idCaja"] = $this->caja_abierta->idcaja;
if ($this->Ventas_model->save($data)) {
    $idventas = $this->Ventas_model->ultimoID();
    $numero_documento = $serie . "-" . $numero_formateado;
    $this->updateCantidadSerie($id_serie, $numero);
    $this->guarda_detalle($idproductos, $idventas, $precios, $cantidades, $importes, $importesaxu, $importeseditar, $consulta_compra, $lotes, $tipoigvs, $fecha, $numero_documento);
    $this->guardar_detalle_venta_lote($lotes, $idventas);
    echo $idventas;
} else {
    echo false;
}
}
}

// confirmar la venta borrador
public function confirmar_venta_borrador()
{
    $idVenta = $this->input->post("idVentaCobrar");
    $recibido = $this->input->post("recibido");
    $cambio = $this->input->post("cambio");
    $formaPago = $this->input->post("forma_pago");
    $resultado = $idVenta;

    if ($this->caja_abierta != null) {
        try {

```

```

// actualizo la venta a estado vendido
$data_venta = array(
    'estado' => VENDIDO,
    'recibido' => isset($recibido) ? $recibido : 0,
    'cambio' => isset($cambio) ? $cambio : 0,
    'forma_pago_id' => $formaPago,
    'idCaja' => $this->caja_abierta->idcaja // se agrego el id caja
);
$this->Ventas_model->update_venta($idVenta, $data_venta);
// obtenemos la venta
$venta = $this->Ventas_model->getVentaDatos($idVenta);

// obtenemos el detalle de la venta
$dDetalleVenta = $this->Ventas_model->getDetalle($idVenta);

if ($venta->forma_pago_id == FORMA_PAGO_CONTADO) {
    $idcaja = $this->caja_abierta->idcaja;
    $ingresos = $this->caja_abierta->ingresos;

    $data_caja = array(
        'ingresos' => $ingresos + $venta->total
    );
    $this->Caja_model->update_caja($idcaja, $data_caja);

    // metodo para actualizar el monto estimado
    $this->caja_abierta = $this->Caja_model->traer_Caja_aperturada();
    $id_caja = $this->caja_abierta->idcaja;
    $ingresos_caja = $this->caja_abierta->ingresos;
    $egresos = $this->caja_abierta->egresos;
    $monto_aper = $this->caja_abierta->monto_apertura;

    if ($egresos != null || $egresos != 0) {
        $estimado = ($monto_aper + $ingresos_caja) - $egresos;
    } else {
        $estimado = ($monto_aper + $ingresos_caja);
    }
    $data_caja_update = array(
        'monto_estimado' => $estimado
    );
    $this->Caja_model->update_caja($id_caja, $data_caja_update);
}
$this->guardar_confirmar_kardex($venta, $dDetalleVenta, 'VENTA');
$this->guarda_cuenta_por_cobrar($venta->idCliente, $venta->id, $venta->total, $venta->forma_pago_id, $venta->recibido);
} catch (Exception $e) {
    echo $e->getMessage();
    error_log($e->getMessage());
    $resultado = 0;
}

```

```

    echo $resultado;
} else {
    echo -1;
}
}

public function guardar_confirmar_kardex($venta, $detalleVenta, $destino)
{
    $idacceso = $venta->idacceso;
    $fecha = date("Y-m-d H:i:s");
    $numero_documento = $venta->serie . "-" . $venta->num_venta_id;
    foreach ($detalleVenta as $dv) {
        $producto_actual = $this->Productos_model->getProducto($dv->producto_id);
        $stock = $producto_actual->stock;
        $data_kardex = array(
            'id_producto' => $dv->producto_id,
            'id_venta' => $venta->id,
            'idacceso' => $idacceso,
            'tipo' => "s",
            'pu' => $dv->precio,
            'stock' => $stock,
            'fecha_emision' => $fecha,
            'numero_documento' => $numero_documento,
            'destino' => $destino,
            'cantidad' => $dv->cantidad_unitaria
        );
        $this->Kardex_model->save_kardex($data_kardex);
    }
}

public function eliminar_venta_borrador()
{
    $idVenta = $this->input->post("idVenta");
    $data_venta = array(
        'estado' => VENTA_CANCELADA
    );
    $this->Ventas_model->update_venta($idVenta, $data_venta);
    $detalle_venta = $this->Ventas_model->getDetalle($idVenta);

    if ($detalle_venta != "") {
        foreach ($detalle_venta as $d) {
            $cantidadFinal = 0;
            $cantidadFinalCaja = 0;
            $idproducto = $d->producto_id;
            $productoActual = $this->Productos_model->getProducto($idproducto);
            $cantidadFinal = $d->cantidad_unitaria; // devolver el stock

            $data_prod = array(
                'stock' => $productoActual->stock + $cantidadFinal
            );
        }
    }
}

```

```

);

if ($d->tipo_pvp == "CAJA") {
    $cantidadFinalCaja = $d->cantidad;
    $data_prod = array(
        'stock_caja' => $productoActual->stock_caja + $cantidadFinalCaja
    );
}
$this->Productos_model->update($idproducto, $data_prod);

$detalleVentaLote = $this->Ventas_model->getDetalle_Venta_Lote($d->id_detalle_v);

foreach ($detalleVentaLote as $dlv) {
    $lote = $this->Lote_model->getLotelotidLote($dlv->id_lote);
    $data_lote = array(
        'stock_lote' => $lote->stock_lote + $dlv->cantidad
    );
    $this->Lote_model->update_lote3($dlv-
>id_lote, $data_lote); // actualiza el stock del lote del producto
}
}
echo true;
} else {
    echo false;
}
}

// otros metodos
public function abrir_alerta()
{
    $this->load->view("layouts/header");
    $this->load->view("layouts/aside");
    $this->load->view("admin/ventas/alerta");
    $this->load->view("layouts/footer");
}

public function abrir_alerta_vendedor()
{
    $this->load->view("layouts/header");
    $this->load->view("layouts/aside");
    $this->load->view("admin/ventas/alertaVendedor");
    $this->load->view("layouts/footer");
}

protected function comprobarDisponibilidadCajaFecha()
{
    date_default_timezone_set('America/Lima');
    $fecha_actual = date("Y-m-d");
    $idacceso = $this->session->userdata("id");

```

```

$caja_abierta = $this->Caja_model->comprueba_caja_abierta2($fecha_actual, $idacceso, 0);
if ($caja_abierta) {
    return $caja_abierta;
}
return false;
}

protected function comprobarDisponibilidadCaja()
{
    date_default_timezone_set('America/Lima');
    $idacceso = $this->session->userdata("id");
    $caja_abierta = $this->Caja_model->traer_caja_abierta($idacceso); // todos los tiempos
    if ($caja_abierta) {
        return $caja_abierta;
    }
    return false;
}

public function enviar_correo($idventa, $folder, $name)
{
    $venta = $this->Ventas_model->getVenta($idventa);
    $mensaje = "<p>Gracias por su compra</p>";
    $email = $venta->email;
    $nombres = $venta->nombres . " " . $venta->apellidos;

    $dominio_email = $this->obtiene_email($email);
    if ($this->send_mail($idventa, $email, $nombres, $mensaje, $folder, $name, $dominio_email)) {
    }
}

public function obtiene_email($email)
{
    return $domain = strstr($email, '@');
}

public function send_mail($idventa, $email, $nombres, $mensaje_html, $folder, $name, $dominio)
{
    // *****CREAR PDF*****
    *****
    $data = array(
        'venta' => $this->Ventas_model->getVenta($idventa),
        'detalle' => $this->Ventas_model->getDetalle($idventa),
        'confi' => $this->Configuracion_model->getConfiguracion()
    );
    $this->load->view('admin/pdf/pdf_email_factura', $data);
    $html = $this->output->get_output();
    // require_once "libraries/pdf.php";
    $this->load->library('pdf');
    $this->dompdf->loadHtml($html);

```

```

$this->dmpdf->setPaper('A4', 'vertical');
$this->dmpdf->render();
$output = $this->dmpdf->output();
$file_to_save = $folder . $name;
if (!file_exists($folder)) {
    if (mkdir($folder, 0777, true)) {
        file_put_contents($file_to_save, $output);
    }
} else {
    file_put_contents($file_to_save, $output);
}

// *****ENVIAR PDF EMAIL*****
*****

$msg = null;
$this->load->library('My_PHPMailer');
$mail = new PHPMailer();
// indico a la clase que use SMTP
$mail->isSMTP();
// Debo de hacer autenticación SMTP
$mail->SMTPAuth = true;

if ($dominio == "@gmail.com") {
    $mail->SMTPSecure = "ssl";
    // indico el servidor de Gmail para SMTP
    $mail->Host = "smtp.gmail.com";
    // indico el puerto que usa Gmail
    $mail->Port = 465;
    // indico un usuario / clave de un usuario de gmail
    $mail->Username = "pharmaexpress@gmail.com";
    $mail->Password = "localvida2000";
    $mail->From = "pharmaexpress@gmail.com";
} elseif ($dominio == "@hotmail.com") {
    $mail->SMTPSecure = "tls";
    $mail->Host = "smtp.live.com";
    $mail->Port = 25;
    $mail->Username = "pharmaexpress@hotmail.com";
    $mail->Password = "locavida2000";
    $mail->From = "pharmaexpress@hotmail.com";
} elseif ($dominio == "@yahoo.com") {
    $mail->SMTPSecure = "ssl";
    $mail->Host = "smtp.mail.yahoo.com";
    $mail->Port = 25;
    $mail->Username = "pharmaexpress@yahoo.com";
    $mail->Password = "locavida2000";
    $mail->From = "pharmaexpress@yahoo.com";
}
$mail->FromName = "Pharma Express";
$mail->Subject = "Compra realizada";

```

```

$mail->addAddress($email, $nombres); // DESTINATARIO

if ($file_to_save) {
    $mail->addAttachment($file_to_save);
}

$mail->MsgHTML($mensaje_html);

if ($mail->Send()) {
    $msg = true;
} else {
    $msg = false;
}
return $msg;
}
public function validar_stock()
{
    $id_producto = $this->input->post("id");
    $cantidadUnidades = $this->input->post("cantidadUnidades");
    $stock_sistema = $this->Productos_model->validar_stock_real($id_producto);
    $stock_sistema->cantidad_to_vender = $cantidadUnidades;
    echo json_encode($stock_sistema);
}
}

```

5.2 Interfaces graficas

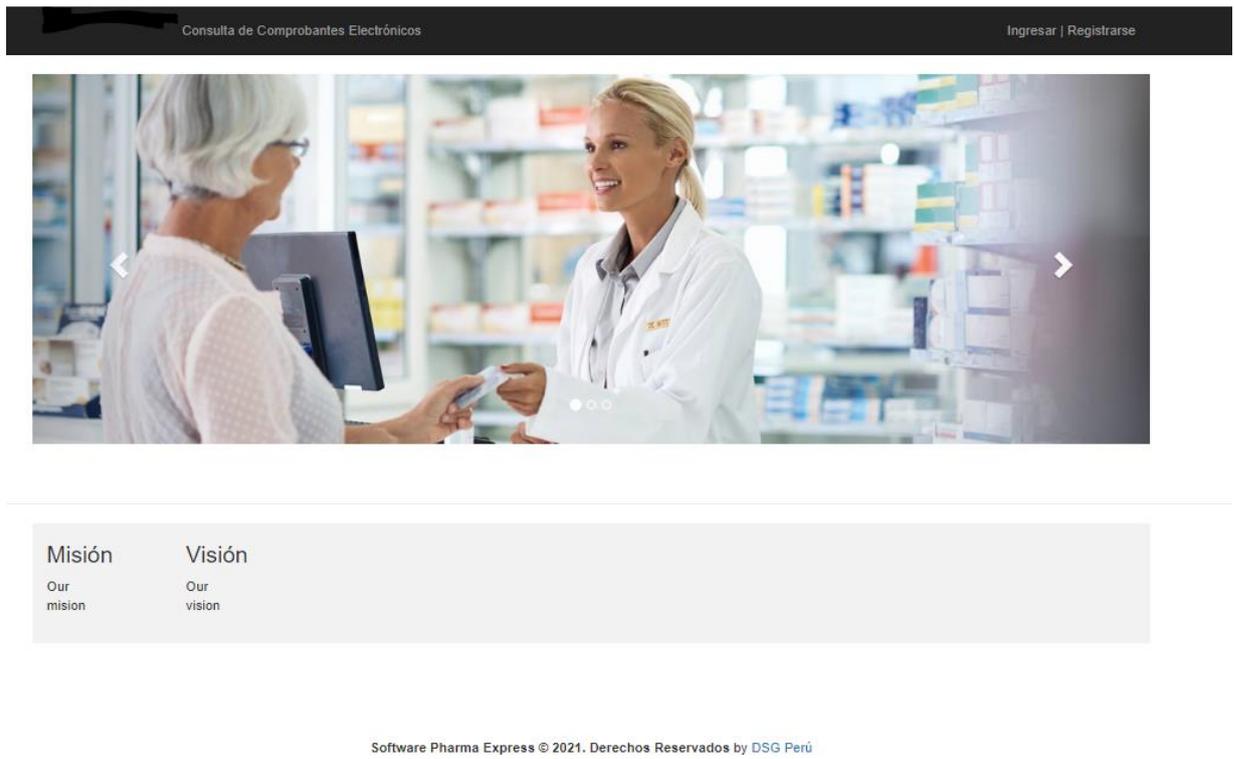


Figura 40. Ventana de bienvenida al sistema

Fuente: Elaboración propia

The screenshot shows the login window of the Pharma Express system. It features a blue button at the top that says "Ingrese sus datos de acceso". Below this are two input fields: "DNI/RUC" and "Password". At the bottom of the input fields is a blue button that says "Ingresar". Below the "Ingresar" button are three links: "Ingresar", "Registrarse", and "Página Principal".

Figura 41. Ventana de logeo al sistema.

Fuente: Elaboración propia

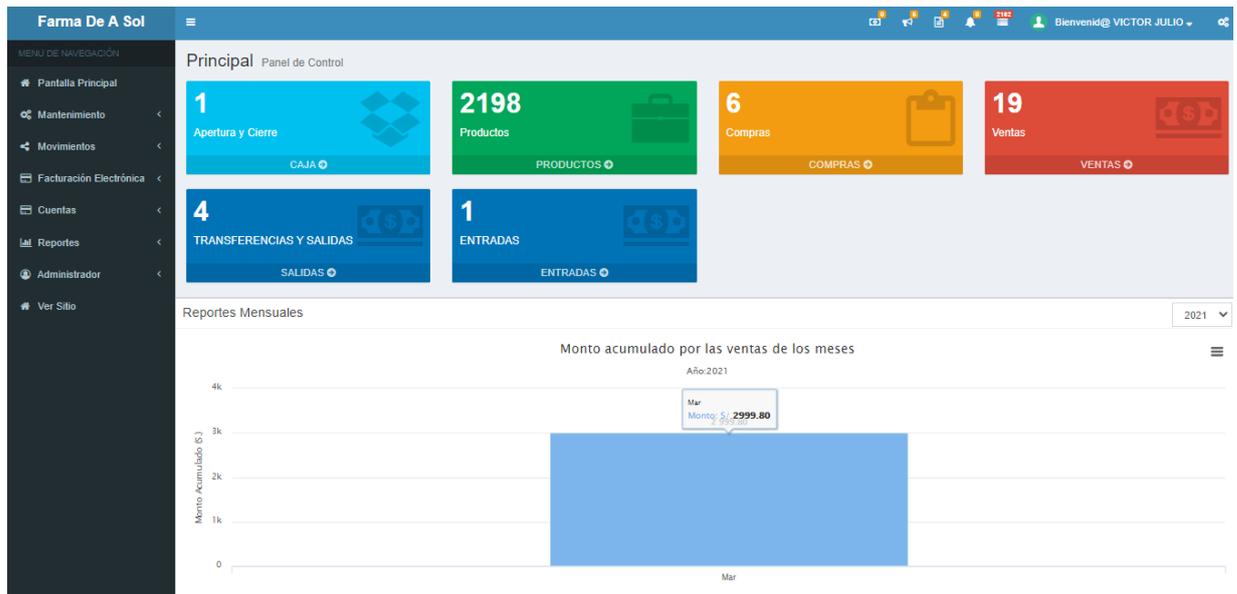


Figura 42. Dashboard del sistema.

Fuente: Elaboración propia

Ventas Listado CAJA: Nro:10. Monto de apertura \$/. 1389.20 - Fecha Apertura: 2021-03-09 09:41:40 AM

+ Agregar Ventas (F2)

Usuario: TODOS

Total Ventas: \$: 381.00 \$: 0 €: 0

Desde: dd/mm/aaaa Hasta: dd/mm/aaaa

Buscar Restablecer

Ventas Aprobados / Anulados Ventas Pendientes Ventas Anuladas

Mostrar 10 registros

#	Nombres del Cliente	Vendedor	Documento	Fecha	Total \$.	F. Pago	Estado	Opciones
84	PUBLICO EN GENERAL	VICTOR JULIO CARREÑO ARONE	B001-00000035	2021-03-29 11:53:00	40.00	EFFECTIVO	ANULADO	
83	PUBLICO EN GENERAL	VICTOR JULIO CARREÑO ARONE	B001-00000034	2021-03-29 11:06:00	67.90	EFFECTIVO	ANULADO	
82	PUBLICO EN GENERAL	VICTOR JULIO CARREÑO ARONE	B001-00000033	2021-03-26 21:53:00	42.00	EFFECTIVO	ANULADO	
81	PUBLICO EN GENERAL	VICTOR JULIO CARREÑO ARONE	B001-00000032	2021-03-26 21:48:00	67.90	EFFECTIVO	ANULADO	
80	PUBLICO EN GENERAL	VICTOR JULIO CARREÑO ARONE	B001-00000031	2021-03-26 21:38:00	67.90	EFFECTIVO	ANULADO	

Figura 43. Ventana listado de ventas

Fuente: Elaboración propia

Farma De A Sol

VENTAS Nueva

Comprobante: (*) Serie: Numero: Cliente: (*)

Ingrese Nombre del Producto: (*) Ingrese Nombre del Servicio: (*) PUNTOS:

Fecha de Venta: Items: Moneda: Tipo Cambio: Evento: 0 % Dcto.

Nombre	PIU	Stock Unidades	Unid x Caja	Unid x Blister	Tipo Igv	Presentación Venta	Cantidad	Importe	Eliminar
Subtotal:	SubTotal		Decto: 0.00		Total:	Total			

Totales

Exonerada	0.00
Inafecta	0.00
Gravada	0.00
IGV	0.00
Gratuita	0.00
Descuento	0.00

Figura 44. Realizar Venta

Fuente: Elaboración propia

Farma De A Sol

LISTA DE PRODUCTOS

Registros: Buscar Por:

Procesando, espere por favor...

Código Barra	Producto	U.M	Stock Unitario	Stock X Caja	Precio Caja	Precio Unitario	Precio Blister	Laboratorio	Principio Activo	Patología	Similar	Para que Sirve	Ubicación	Fecha de Caducidad	Elegir
002305	MALLA POLIPROPILENO 15 CM X 15 CM	UND	3494	0	S/0	S/42	0.00	CIRUGIA PERUANA					SU1	2025-03-01	<input checked="" type="checkbox"/>
7750215007822	AMLODIPINO 10MG X 100 UND	BLIS	1500	0	S/0	S/40	3.50	PORTUGAL					SU1	2022-11-30	<input checked="" type="checkbox"/>
001186	LAMINA CUBRE OJETO 22X22MM X 100 UND	UND	1497	0	S/0	S/7	0.00	SL				<input type="button" value="Info"/>	SU1	2025-03-21	<input checked="" type="checkbox"/>
7750942541880	AMOXICILINA 250MG X 100 TABLETAS	CJA.BLIS	1226	12	S/20	S/2	3.50	INDUQUIMICA	AMOXICILINA		<input checked="" type="checkbox"/>	<input type="button" value="Info"/>	SU1	2023-02-28	<input checked="" type="checkbox"/>
002117	ENFAMIL ENFACARE PREMIUM X 353 GR	UND.CJA	1004	1004	S/20	S/67.9	0.00	SL					SU1	2025-03-21	<input checked="" type="checkbox"/>

Totales

Inafecta	0.00
Gravada	0.00
IGV	0.00
Gratuita	0.00
Descuento	0.00

Figura 45. Búsqueda de medicamentos

Fuente: Elaboración propia

Ventas Nueva

Comprobante:(*) Serie: Numero: Cliente: (*)

Ingrese Nombre del Producto: (*) Ingrese Nombre del Servicio: (*) PUNTOS:

Fecha de Venta: Items: 1 Moneda: Tipo Cambio: Evento: 0 % Dcto.

Nombre	P/U	Stock Unidades	Unid x Caja	Unid x Blister	Tipo Igv	Presentación Venta	Cantidad	Importe	Eliminar
ENFAMIL ENFACARE PREMIUM X 363 GR (SL)	67.9	1004	1	0	Exonerado - Operaciór	PVP1	3	203.70	<input type="button" value="X"/>

Subtotal: Dscto: Total:

Totales

Exonerada	<input type="text" value="203.70"/>
Inafecta	<input type="text" value="0.00"/>
Gravada	<input type="text" value="0.00"/>
IGV	<input type="text" value="0.00"/>
Gratuita	<input type="text" value="0.00"/>
Descuento	<input type="text" value="0.00"/>
Total	<input type="text" value="203.70"/>

Figura 46. Calculo automático de los montos de venta

Fuente: Elaboración propia

Cobrar

Monto Recibido: Monto a Cobrar:

Forma de Pago: Cambio:

Figura 47. Pago de venta

Fuente: Elaboración propia



Figura 48. Factura de venta generada
Fuente: Elaboración propia

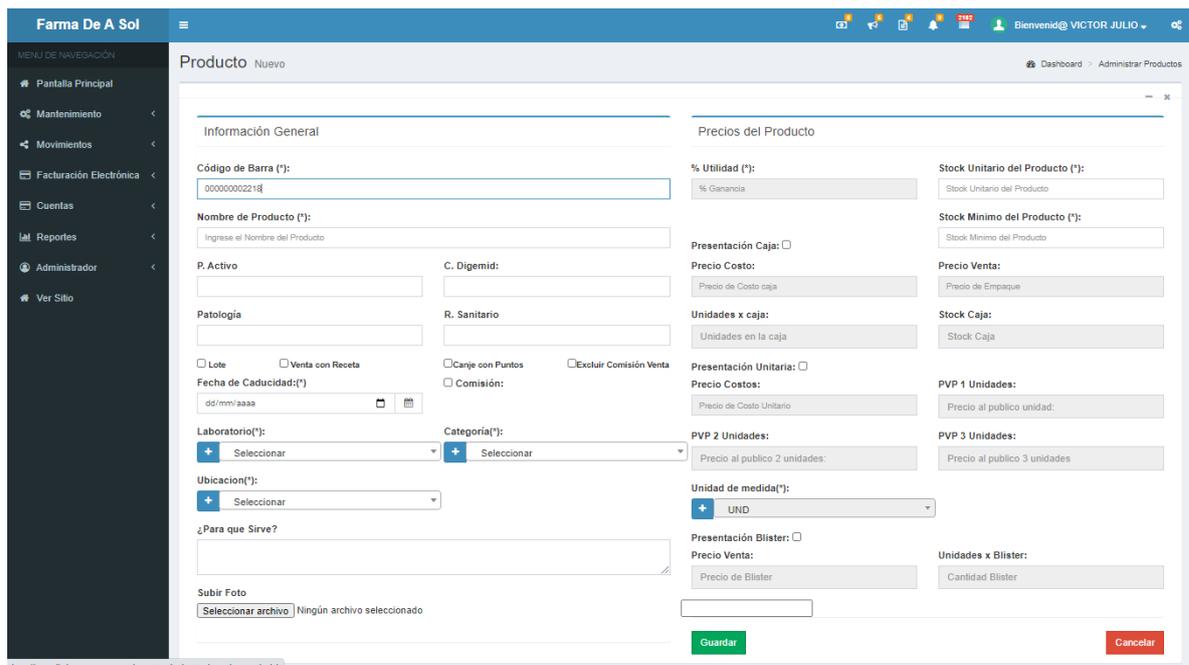


Figura 49. Mantenimiento de productos
Fuente: Elaboración propia

Farma De A Sol

MENÚ DE NAVEGACIÓN

- Pantalla Principal
- Mantenimiento
- Movimientos
- Facturación Electrónica
- Cuentas
- Reportes
- Administrador
 - Rol
 - Usuarios
 - Permisos
 - Configuración de la Empresa
 - Configuración del Sistema
 - Configuración Facturación
 - Configuración Puntos
 - Generador Código de Barra
 - Formas de Pago
 - Tipo de Venta
 - Tipo de Compra
 - Configuración Backup
- Ver Sitio

Información de la Empresa

Información Básica

Razón Social: Farma De A Sol Meliss

RUC: 20606225645

Gerente: Victor Julio Carreño Arone

Porcentaje de IGV: 0

Domicilio Fiscal: Av. Tulumayo 872

Telefono: 994687084

Email: corporacionsocorro@hotmail.com

Celular: 994687084

Moneda: Soles

Descripción: Farma de A Sol Meliss

Misión: Our mission

Visión: Our vision

Codigo Establecimiento Digemid: 0088017

Logo Empresa: Ningún archivo seleccionado

Logo de Facturación: Ningún archivo seleccionado

Guardar **Cancelar**

Figura 50. Configurar información de empresa

Fuente: Elaboración propia

Farma De A Sol

MENÚ DE NAVEGACIÓN

- Pantalla Principal
- Mantenimiento
- Movimientos
- Facturación Electrónica
- Cuentas
- Reportes
- Administrador
- Ver Sitio

Facturación Documentos Electrónicos

Rango de fechas: 01/11/2020 al 22/04/2021 **Cargar Reporte**

Sel. Todo **Enviar**

Mostrar: 10 registros **Exportar**

Nro Doc	Tipo de Comprobante	Tipo Documento	Nro. Documento	Fecha	Cliente	Total	Hora envío a Sunat	Desc XML	Guía de Remisión	Enviar	Respuesta Sunat(CDR)	Ver Ticket	Enviar Correo	Envía What
B001-000001	BOLETA	DNI	00000000	26-03-2021	PUBLICO EN GENERAL	40.00	2021-03-26 00:00:00							
B001-000002	BOLETA	DNI	00000000	26-03-2021	PUBLICO EN GENERAL	7.00	2021-03-26 00:00:00							
B001-000003	BOLETA	DNI	00000000	26-03-2021	PUBLICO EN GENERAL	7.00	2021-03-26 00:00:00							
B001-000004	BOLETA	DNI	00000000	26-03-2021	PUBLICO EN GENERAL	7.00	2021-03-26 00:00:00							
B001-000005	BOLETA	DNI	70498426	26-03-2021	RAUL BARDALES DELAGUILA	7.00	2021-03-26 00:00:00							
B001-000006	BOLETA	DNI	00000000	26-03-2021	PUBLICO EN GENERAL	7.00	2021-03-26 00:00:00							

Figura 51. Ventana de Documentos Electrónicos

Fuente: Elaboración propia

CAPITULO VI: ARQUITECTURA

6.1 Alcance

El alcance de la arquitectura de software planteada considera únicamente el análisis, modelamiento y desarrollo de la solución para el “Proceso de Ventas de la botica el Gran Poder”, así como también la interconexión con los servicios de integración de SUNAT WebService API 2.1 y Reniec WebService.

Este sistema se encarga de automatiza el control de stock de productos o medicamentos, emisión de facturas y boletas electrónicas, llevar el control de las ventas, mostrar reportes, indicadores, alta de usuarios y clientes.

6.2 Definición, acrónimos y abreviaturas

6.2.1 Definición

- Requerimiento/Requisito: Es la descripción de una funcionalidad el sistema.
- Caso de uso: Es la descripción de un flujo del sistema.
- Funcionalidad: Descripción de lo que el software debe hacer.
- Interfaces Externas: Cómo debe interactuar el sistema con las personas, el sistema de hardware, o con otros sistemas (software y hardware).
- Rendimiento: Indicación de la velocidad, disponibilidad, tiempos de respuesta, tiempos de recuperación, tiempos de determinadas funciones.
- Calidad: Cumplir con las especificaciones dadas por los stakeholders

- Stackholder: Es el dueño o interesado del sistema.
- Cliente/Usuarios: Son todas las personas quienes hacen uso de los servicios que ofrece la empresa.
- IpharmaExpress: Sistema de Gestión de Boticas y Farmacias.
- Clase: Abstracción de un objeto para la creación de un modelo de datos.

6.2.2 Acrónimos

- MVC: Patrón de diseño de software.
- UML:
- PHP:

6.2.3 Abreviaturas

- CI: CodeIgniter
- REQ: Requerimiento
- BD: Base de datos

6.3 Referencias

Los documentos o diagramas basados en el estándar UML 2.1 que se realizaron para la hacer referencia a la arquitectura del sistema son:

Nº	Documento/Diagrama
1	Diagrama de Requerimientos Funcionales
2	Diagrama de Requerimientos No Funcionales
3	Diagrama de casos de uso del sistema
4	Diagrama de casos de uso
5	Diagrama de objetos
6	Diagrama de secuencia
7	Diagrama de dominio
8	Diagrama de clases
9	Diagrama de base de datos
10	Diagrama de Actividades ASIS y TOBE
11	Diagrama de reglas de Negocio
12	Diagrama de despliegue

Tabla 21. Documentos de análisis realizados
Fuente: Elaboración propia

6.4 Visión general

Se presenta la visión general del negocio que en esencia se encuentra en su misión, visión, equipo y valores que están descritos en su documento de visión empresarial.

Misión:

Ofrecer una asistencia sanitaria de calidad, para alcanzar y mantener la salud, ofreciendo servicios farmacéuticos y productos, de forma profesional, servicial y cercana.

Visión:

Con respecto a nuestros usuarios: facilitar un rápido acceso a los productos que precisen, primando los servicios profesionales, para que saquen el máximo beneficio a sus medicamentos.

Sobre nuestro equipo: responsabilizarse, respetar y cumplir el trabajo de forma individual, siendo conscientes de la importancia que tiene el trabajo de cada uno para conseguir un equipo eficaz y dinámico.

Valores:

Compromiso, responsabilidad, seguridad

Humanidad, honestidad, respeto

Calidad, resolución, eficiencia

Profesionalidad

6.5 Representación de la arquitectura

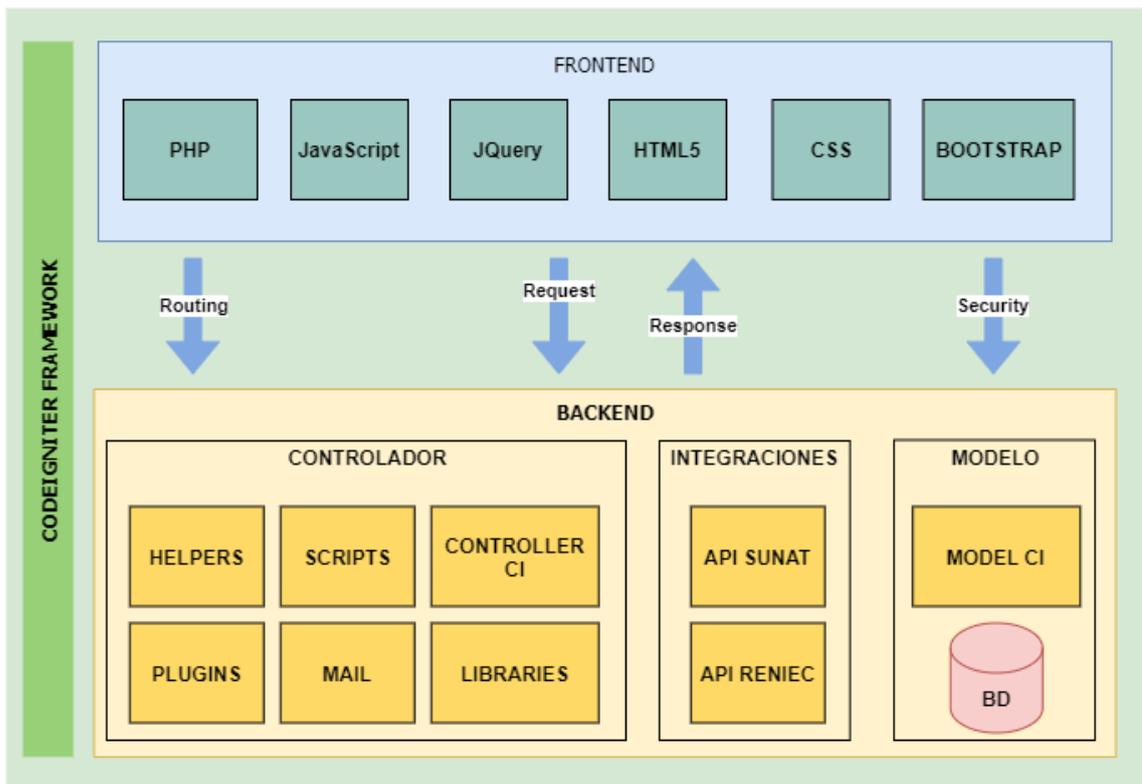


Figura 52. Arquitectura general de la solución.

Fuente: Elaboración propia

6.6 Metas y restricciones de la arquitectura

Se han identificado los siguientes requerimientos no funcionales y funcionales que definen las metas y restricciones arquitectónicas.

A) Separación lógica de capas de componentes Backend y FrontEnd.

La arquitectura se separó estos dos grandes componentes implementados con el patrón de diseño MVC y se implementó con la ayuda de un framework que implementa ese patrón como lo es Codeigniter.

B) Lenguaje de programación a usarse debe ser PHP

La aplicación se programó en el lenguaje PHP se presentó como propuesta, luego fue aprobado por los stakeholders debió a un tema de bajo coste en mantenimiento y hosting.

- C) Integración con los servicios API Web Service de Reniec y Sunat
 La aplicación está integrada a Reniec para obtener los datos de los usuarios y clientes de forma automática, así mismo está integrado a SUNAT para la emisión automática de documentos electrónicos.
- D) Las licencias de los productos de las herramientas de software no deben significar costos elevados para el proyecto.
 Por lo que opto por recurrir a herramientas que tiene licencia GLP o Software libre.
- E) La aplicación debe tener interfaces intuitivas y adaptables a distintos tipos de pantallas.
 Las vistas de la aplicación están construidas con Bootstrap 4 lo cual permite la adaptabilidad de distintas pantallas.

6.7 Vistas o planos

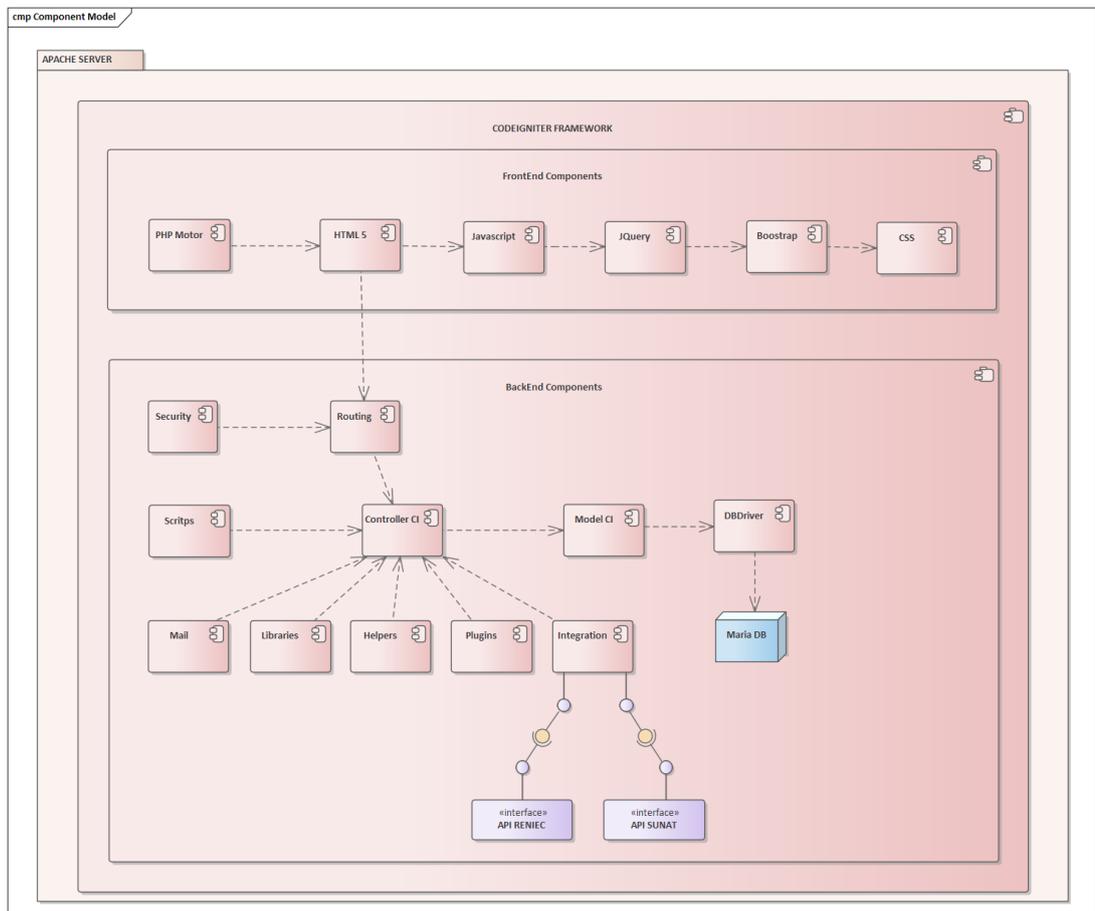


Figura 53. Diagrama de los componente de arquitectura

Fuente: Elaboración propi

6.8 Implementación

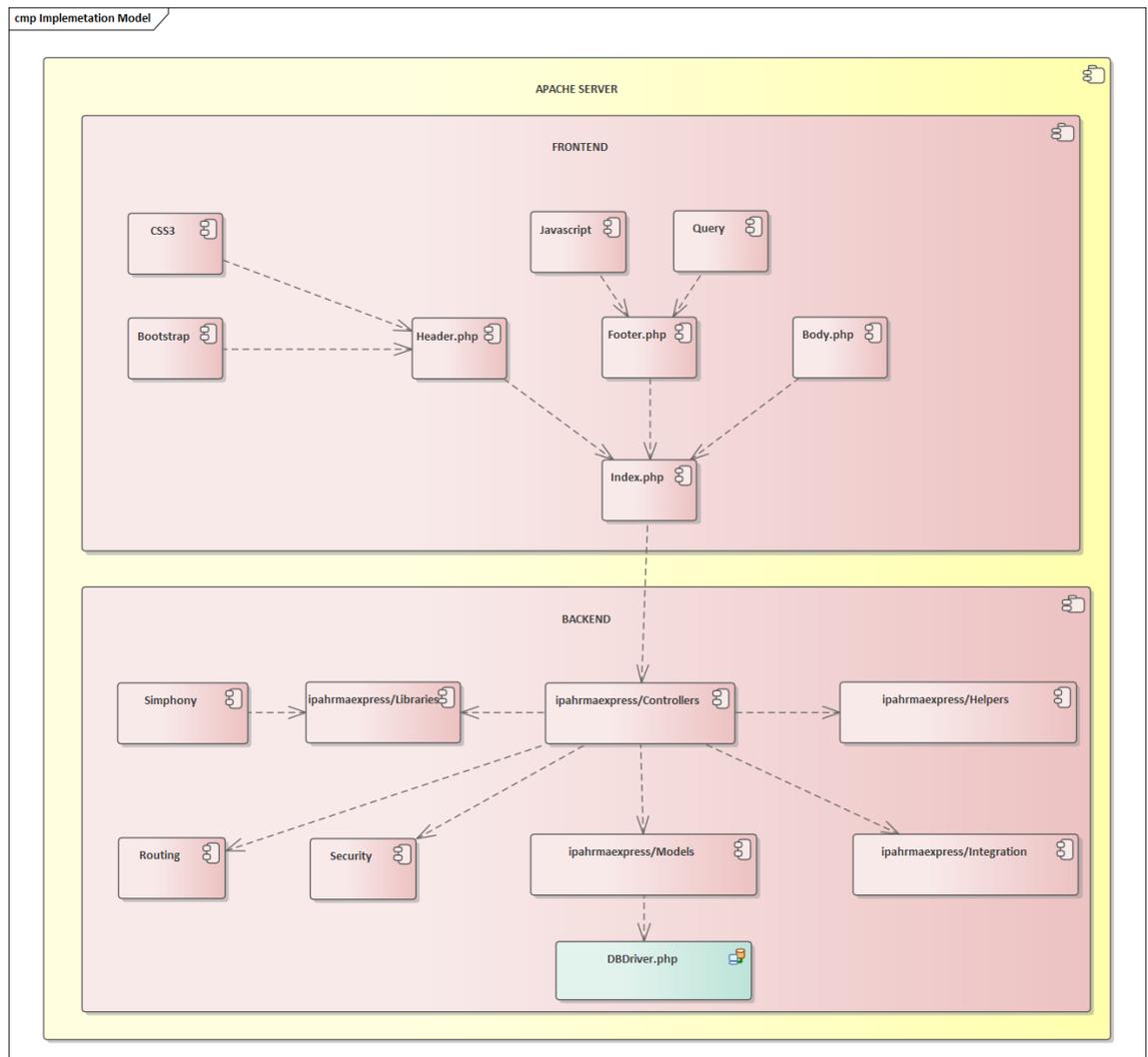


Figura 54. Diagrama de componente de implementación
Fuente: Elaboración propia

6.9 Calidad

Los resultados de la medición de calidad usando el estándar ISO/IEC 25000 SQuaRe (Requerimiento y Evaluación de Calidad de Software), para el software IpharmaExpress, la Tabla 23 indica que la calidad interna y externa tienen un valor de 7,55/10 y 9,08/10 respectivamente, considerándolas como SATISFACTORIO y MUY SATISFACTORIO respectivamente, dando así un resultado final de 8,31/10, considerándolo como un producto satisfactorio, el cual representa un 83% de la calidad total.

CALIDAD	CALIDAD DEL SISTEMA	NIVEL DE PUNTUACION	GRADO DE SATISFACCION
Interna	7,55	ACEPTABLE	SATISFACTORIO
Externa	9,08	CUMPLE CON LOS REQUISITOS	MUY SATISFACTORIO
Total	8,31	ACEPTABLE	SATISFACTORIO

Tabla 22. Resultado final del análisis de calidad de software
Fuente: Vivanco, A, Elaboración propia.

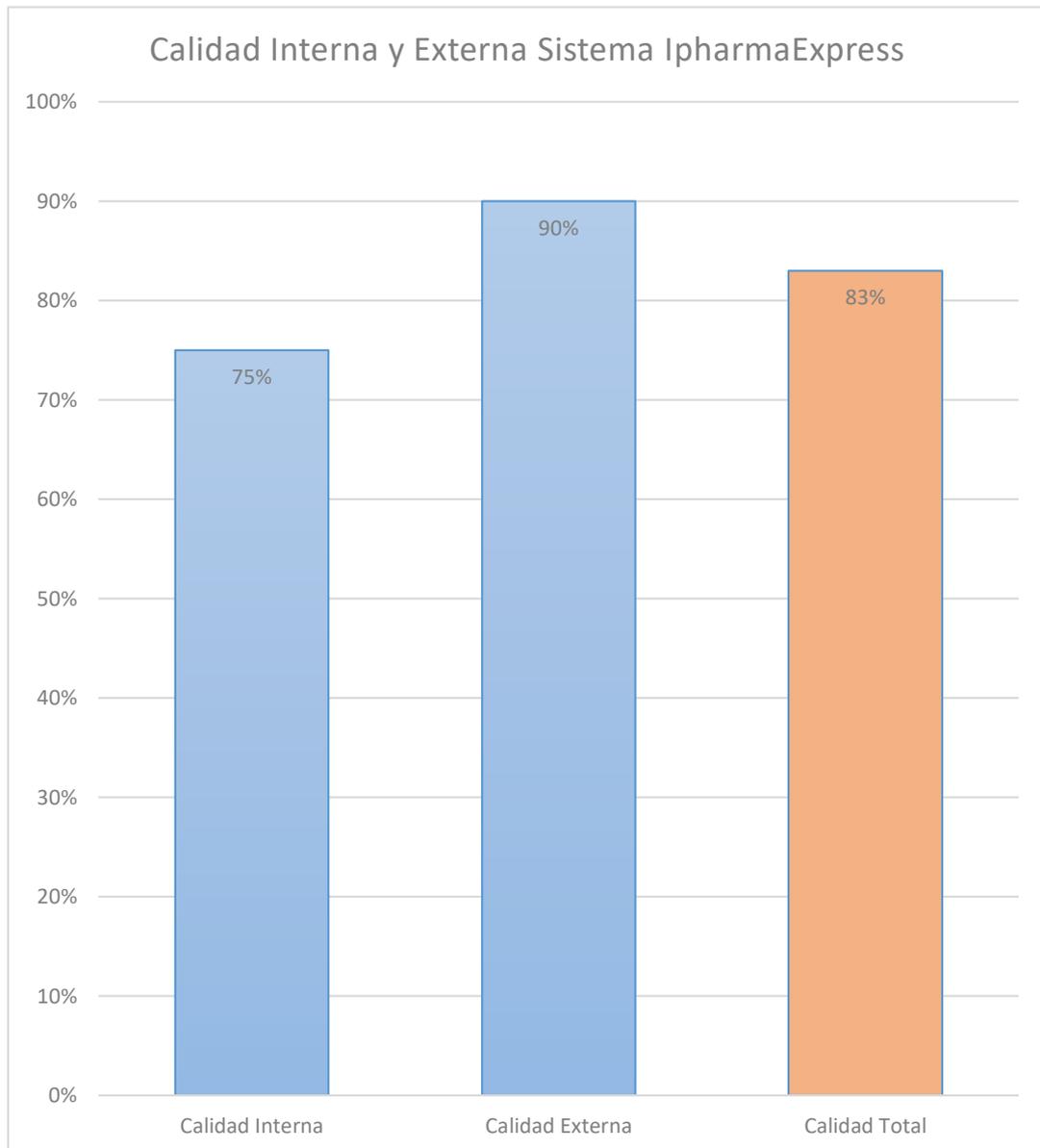


Figura 55. Resultado del porcentaje de calidad
Fuente: Elaboración propia

En la Tabla 24 se muestra el resumen de los valores obtenidos de las características que fueron aplicadas en la evaluación de calidad, en donde se puede apreciar que la calidad externa tiene un valor mayor con respecto a la calidad interna lo que indica que a nivel de código el sistema tiene oportunidades de mejora, asimismo se puede apreciar a detalle cómo se obtuvieron estos resultados en el anexo 2.

VALOR TOTAL OBTENIDO DE CADA CARACTERÍSTICA DE CALIDAD							
	Características	Valor Parcial Total (/10)	Nivel de importancia	Porcentaje de importancia	Valor Final	Subtotal de la calidad del sistema (/10)	Calidad Total del sistema (/10)
CALIDAD INTERNA	Adecuación Funcional	10,00	M	15%	1,50	7,55	8,31
	Fiabilidad	10,00	M	10%	1,00		
	Eficiencia en el desempeño	5,60	M	20%	1,12		
	Usabilidad	7,53	M	20%	1,51		
	Mantenibilidad	6,92	A	35%	2,42		
CALIDAD EXTERNA	Adecuación Funcional	10,00	A	30%	3,00	9,08	
	Fiabilidad	8,75	M	15%	1,31		
	Eficiencia en el desempeño	9,00	M	20%	1,80		
	Usabilidad	6,46	M	15%	0,97		
	Mantenibilidad	10,00	M	20%	2,00		

Tabla 23. Resumen de matriz de calidad
Fuente: Balseca (2014)

6.10 Diseño de base de datos

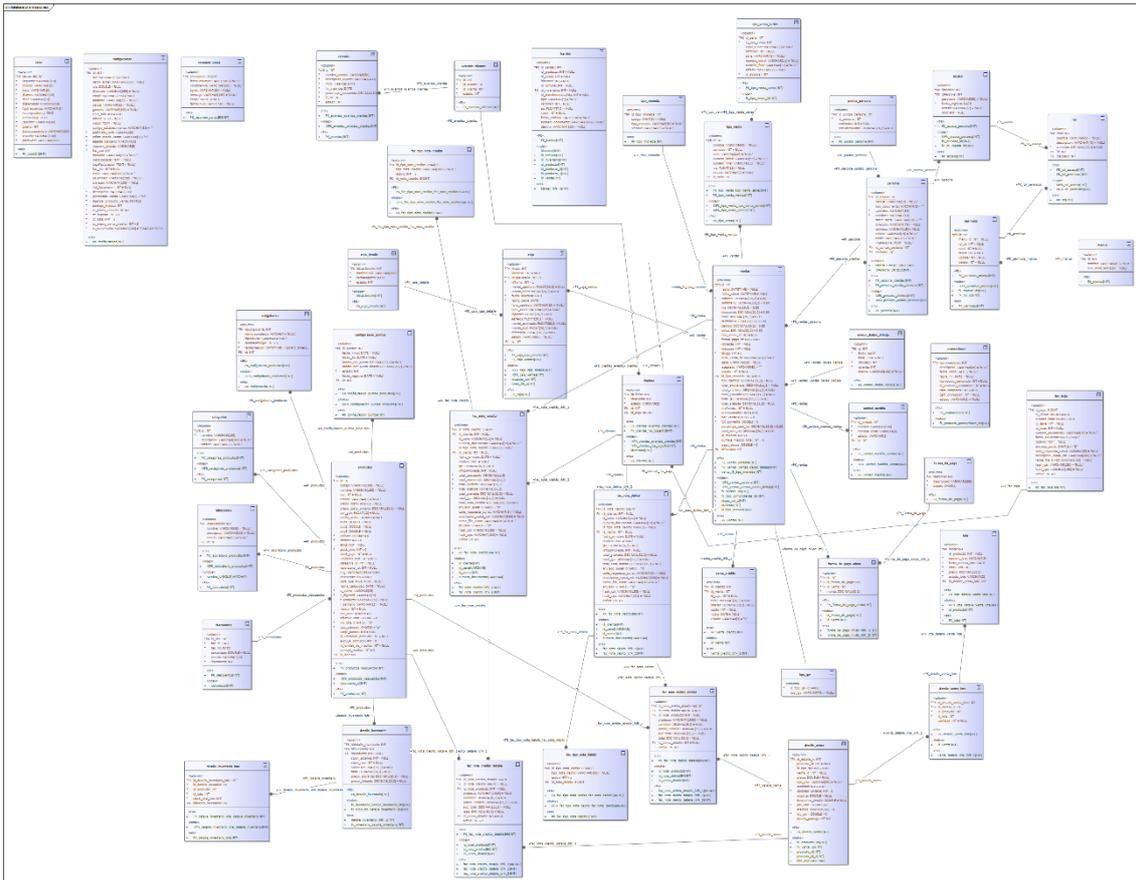


Figura 56. Diagrama de base de datos
Fuente: Elaboración propia.

6.10.1 Script de tablas

```
-- phpMyAdmin SQL Dump
-- version 5.0.4
-- https://www.phpmyadmin.net/
-- Servidor: 127.0.0.1
-- Tiempo de generación: 19-04-2021 a las 02:22:55
-- Versión del servidor: 10.4.17-MariaDB
-- Versión de PHP: 7.3.26
```

```
SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
START TRANSACTION;
SET time_zone = "+00:00";
```

```
/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT
*/;
/*!40101 SET
@OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION
*/;
/*!40101 SET NAMES utf8mb4 */;
```

```
--
-- Base de datos: `ipharmaexpress`
--
```

```
-----
--
-- Estructura de tabla para la tabla `acceso`
--
```

```
CREATE TABLE `acceso` (
  `idacceso` int(11) NOT NULL,
  `idPersona` int(11) NOT NULL,
  `password` varchar(50) DEFAULT NULL,
  `fecha_registro` date NOT NULL,
  `estado` varchar(2) DEFAULT NULL,
  `idrol` int(11) DEFAULT NULL,
  `ip_actual` text DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
-----
--
-- Estructura de tabla para la tabla `caja`
--
```

```
CREATE TABLE `caja` (
  `idcaja` int(11) NOT NULL,
  `idacceso` int(11) DEFAULT NULL,
  `idCajaDetalle` int(11) NOT NULL DEFAULT 1,
```

```

`idTurno` int(11) NOT NULL DEFAULT 1,
`monto_apertura` float(20,2) DEFAULT NULL,
`monto_cierre` float(20,2) DEFAULT NULL,
`fecha_apertura` date NOT NULL,
`fecha_cierre` date NOT NULL,
`hora_apertura` varchar(15) DEFAULT NULL,
`hora_cierre` varchar(15) DEFAULT NULL,
`ingresos` float(20,2) DEFAULT NULL,
`egresos` float(20,2) DEFAULT NULL,
`monto_estimado` float(20,2) DEFAULT NULL,
`monto_real` float(20,2) DEFAULT NULL,
`diferencia` float(20,2) DEFAULT NULL,
`estado` varchar(1) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

```

--
-- Estructura de tabla para la tabla `caja_detalle`
--

```

```

CREATE TABLE `caja_detalle` (
  `idCajaDetalle` int(11) NOT NULL,
  `nombreCaja` varchar(100) NOT NULL,
  `fechaRegistro` date NOT NULL,
  `estado` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

```

--
-- Estructura de tabla para la tabla `categorias`
--

```

```

CREATE TABLE `categorias` (
  `id` int(11) NOT NULL,
  `nombre` varchar(100) NOT NULL,

```

```
`descripcion` varchar(100) DEFAULT NULL,  
`estado` varchar(2) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

--

-- Estructura de tabla para la tabla `clientes`

--

```
CREATE TABLE `clientes` (  
  `idCliente` int(11) NOT NULL,  
  `idPersona` int(11) NOT NULL,  
  `estado` varchar(2) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

--

-- Estructura de tabla para la tabla `codigobarra`

--

```
CREATE TABLE `codigobarra` (  
  `idcodigobarra` int(11) NOT NULL,  
  `nombreproducto` varchar(45) DEFAULT NULL,  
  `idproducto` varchar(45) DEFAULT NULL,  
  `cantidadcodigo` int(11) NOT NULL DEFAULT 0,  
  `fechacreacion` datetime NOT NULL DEFAULT current_timestamp()  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

--

-- Estructura de tabla para la tabla `configuracion`

--

```
CREATE TABLE `configuracion` (  
  `id` int(11) NOT NULL,
```

```

`ruc` varchar(13) DEFAULT NULL,
`razon_social` varchar(255) DEFAULT NULL,
`iva` double DEFAULT NULL,
`direccion` varchar(255) DEFAULT NULL,
`email` varchar(255) DEFAULT NULL,
`telefono` varchar(25) DEFAULT NULL,
`celular` varchar(25) DEFAULT NULL,
`gerente` varchar(100) DEFAULT NULL,
`cant_min` int(5) DEFAULT NULL,
`mision` text DEFAULT NULL,
`vision` text DEFAULT NULL,
`codigo_establecimiento` varchar(50) DEFAULT "",
`publicado_web` varchar(5) NOT NULL,
`editar_precio_venta` varchar(5) NOT NULL DEFAULT '0',
`agente_bancario` varchar(5) NOT NULL,
`imprimir_directo` varchar(5) NOT NULL,
`fac_aut` int(1) NOT NULL,
`moneda` varchar(25) DEFAULT NULL,
`logoEmpresa` text DEFAULT NULL,
`logoFacturacion` text DEFAULT NULL,
`fac_ele` int(1) DEFAULT NULL,
`clave` varchar(100) DEFAULT NULL,
`usuariosol` varchar(100) DEFAULT NULL,
`clavesol` varchar(100) DEFAULT NULL,
`ind_facturacion` int(1) DEFAULT NULL,
`descripcion` varchar(200) NOT NULL,
`porcentaje_venta` varchar(5) NOT NULL DEFAULT 'no',
`duplicar_producto_venta` char(1) NOT NULL,
`puntaje_medico` int(1) NOT NULL,
`ct_precio_unitario` int(1) NOT NULL DEFAULT 1,
`ct_importe` int(1) NOT NULL DEFAULT 1,
`ct_total` int(1) NOT NULL DEFAULT 1,
`ct_precio_venta_credito` int(1) NOT NULL DEFAULT 0,
`ct_comentario` varchar(100) NOT NULL DEFAULT 'GRACIAS POR SU PREFERENCIA'
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

```
--  
-- Estructura de tabla para la tabla `configuracion_puntos`  
--
```

```
CREATE TABLE `configuracion_puntos` (  
  `id_puntos` int(11) NOT NULL,  
  `fecha_inicio` date DEFAULT NULL,  
  `fecha_fin` date DEFAULT NULL,  
  `monto_por_punto` decimal(20,2) DEFAULT NULL,  
  `monto_por_punto_dinero` decimal(20,2) DEFAULT NULL,  
  `estado` int(11) NOT NULL,  
  `fecha_registro` date DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
--  
-- Estructura de tabla para la tabla `descuentos`  
--
```

```
CREATE TABLE `descuentos` (  
  `id_des` int(11) NOT NULL,  
  `fec_ini` date NOT NULL,  
  `fec_fin` date NOT NULL,  
  `porcentaje` double DEFAULT NULL,  
  `estado` varchar(10) NOT NULL,  
  `idproducto` int(11) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
--  
-- Estructura de tabla para la tabla `detalle_venta`  
--
```

```
CREATE TABLE `detalle_venta` (  
  `id_detalle_v` int(11) NOT NULL,  
  `producto_id` int(11) DEFAULT NULL,
```

```

`id_tipo_igv` int(11) DEFAULT NULL,
`venta_id` int(11) DEFAULT NULL,
`precio` double DEFAULT NULL,
`tipo_pvp` varchar(50) DEFAULT NULL,
`cantidad` int(11) DEFAULT NULL,
`cantidad_unitaria` int(11) NOT NULL DEFAULT 0,
`importe` double DEFAULT NULL,
`descuento_detalle` double DEFAULT NULL,
`por_com` double NOT NULL DEFAULT 0,
`efectivo_com` double DEFAULT 0,
`tot_com` double NOT NULL DEFAULT 0,
`detalle_puntaje` int(11) DEFAULT 0
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

--

-- Estructura de tabla para la tabla `detalle_venta_lote`

--

```

CREATE TABLE `detalle_venta_lote` (
  `id_detalle_venta_lote` int(11) NOT NULL,
  `id_detalle_v` int(11) NOT NULL,
  `id_producto` int(11) NOT NULL,
  `id_lote` int(11) NOT NULL,
  `cantidad` int(11) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

--

-- Estructura de tabla para la tabla `eventos`

--

```

CREATE TABLE `eventos` (
  `id` int(11) NOT NULL,
  `nombre_evento` varchar(50) NOT NULL,
  `descripcion_evento` varchar(400) NOT NULL,

```

```

`inicio_vigencia` date NOT NULL,
`fin_vigencia` date NOT NULL,
`porcentaje_descuento` decimal(10,0) NOT NULL,
`id_rol` int(11) NOT NULL,
`estado` int(1) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

```

```

--
-- Estructura de tabla para la tabla `eventos_clientes`
--

```

```

CREATE TABLE `eventos_clientes` (
  `id` int(11) NOT NULL,
  `id_evento` int(11) NOT NULL,
  `id_cliente` int(11) NOT NULL,
  `estado` int(1) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

```

```

--
-- Estructura de tabla para la tabla `fac_baja`
--

```

```

CREATE TABLE `fac_baja` (
  `id_baja` bigint(20) NOT NULL,
  `id_cliente` bigint(20) DEFAULT NULL,
  `numero_baja` bigint(20) DEFAULT NULL,
  `id_serie` int(11) DEFAULT NULL,
  `id_local` int(11) DEFAULT NULL,
  `numero_documento` varchar(15) COLLATE utf8_spanish_ci DEFAULT NULL,
  `fecha_anulacion` date DEFAULT NULL,
  `motivo` text COLLATE utf8_spanish_ci DEFAULT NULL,
  `enviado_sunat` char(1) COLLATE utf8_spanish_ci DEFAULT '0',
  `code_respuesta_sunat` varchar(5) COLLATE utf8_spanish_ci DEFAULT NULL,
  `descripcion_sunat_cdr` varchar(300) COLLATE utf8_spanish_ci DEFAULT NULL,

```

```

`name_file_sunat` varchar(80) COLLATE utf8_spanish_ci DEFAULT NULL,
`hash_cdr` varchar(150) COLLATE utf8_spanish_ci DEFAULT NULL,
`hash_cpe` varchar(150) COLLATE utf8_spanish_ci DEFAULT NULL,
`activo` int(11) DEFAULT 1
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_spanish_ci;

```

```

--
-- Estructura de tabla para la tabla `fac_nota_credito`
--

```

```

CREATE TABLE `fac_nota_credito` (
  `id_nota_credito` bigint(20) NOT NULL,
  `id_cliente` int(11) DEFAULT NULL COMMENT 'cliente que usa el app',
  `id_serie` varchar(11) COLLATE utf8_spanish_ci DEFAULT NULL COMMENT 'serie del comprobante',
  `numero_documento` varchar(15) COLLATE utf8_spanish_ci DEFAULT NULL,
  `id_tipo_nota_credito` char(2) COLLATE utf8_spanish_ci DEFAULT NULL,
  `id_venta` int(11) DEFAULT NULL COMMENT 'documento de venta afectado',
  `fecha_emision` date DEFAULT NULL,
  `motivo` text COLLATE utf8_spanish_ci DEFAULT NULL,
  `igv` decimal(8,2) DEFAULT NULL,
  `idTipoMoneda` int(1) DEFAULT NULL,
  `total_descuento` decimal(11,2) NOT NULL,
  `total_exonerada` decimal(11,2) NOT NULL,
  `total_inafecta` decimal(11,2) NOT NULL,
  `total_gratuita` decimal(11,2) NOT NULL,
  `total_gravada` decimal(11,2) DEFAULT NULL,
  `total_igv` decimal(11,2) DEFAULT NULL,
  `total_nota_credito` decimal(11,2) DEFAULT NULL,
  `enviado_sunat` char(1) COLLATE utf8_spanish_ci DEFAULT '0',
  `code_respuesta_sunat` varchar(5) COLLATE utf8_spanish_ci DEFAULT NULL,
  `descripcion_sunat_cdr` varchar(300) COLLATE utf8_spanish_ci DEFAULT NULL,
  `name_file_sunat` varchar(80) COLLATE utf8_spanish_ci DEFAULT NULL,
  `anulado` char(1) COLLATE utf8_spanish_ci DEFAULT '0' COMMENT 'comunicacion de baja',
  `hash_cdr` varchar(150) COLLATE utf8_spanish_ci DEFAULT NULL,
  `hash_cpe` varchar(150) COLLATE utf8_spanish_ci DEFAULT NULL,
  `activo` int(11) DEFAULT 1
)

```

```

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_spanish_ci;

-----

--
-- Estructura de tabla para la tabla `fac_nota_credito_detalle`
--

CREATE TABLE `fac_nota_credito_detalle` (
  `id_nota_credito_detalle` bigint(20) NOT NULL,
  `id_nota_credito` bigint(20) DEFAULT NULL,
  `id_local_producto` int(11) DEFAULT NULL COMMENT 'producto de un local',
  `producto` varchar(1000) COLLATE utf8_spanish_ci DEFAULT NULL,
  `cantidad` decimal(8,2) DEFAULT NULL,
  `precio_unitario` decimal(11,2) DEFAULT NULL,
  `sub_total` decimal(11,2) DEFAULT NULL,
  `total` decimal(11,2) DEFAULT NULL,
  `id_venta_detalle` int(11) DEFAULT NULL COMMENT 'detalle de la venta relacionada',
  `activo` int(11) DEFAULT 1
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_spanish_ci;

```

```

-----

--
-- Estructura de tabla para la tabla `fac_nota_debito`
--

CREATE TABLE `fac_nota_debito` (
  `id_nota_debito` bigint(20) NOT NULL,
  `id_cliente` int(11) DEFAULT NULL COMMENT 'cliente que usa el app',
  `id_serie` varchar(11) COLLATE utf8_spanish_ci DEFAULT NULL COMMENT 'serie del comprobante',
  `numero_documento` varchar(15) COLLATE utf8_spanish_ci DEFAULT NULL,
  `id_tipo_nota_debito` char(2) COLLATE utf8_spanish_ci DEFAULT NULL,
  `id_venta` int(11) DEFAULT NULL COMMENT 'documento de venta afectado',
  `fecha_emision` date DEFAULT NULL,
  `motivo` text COLLATE utf8_spanish_ci DEFAULT NULL,
  `igv` decimal(8,2) DEFAULT NULL,
  `idTipoMoneda` int(1) DEFAULT NULL,

```

```

`total_gravada` decimal(11,2) DEFAULT NULL,
`total_igv` decimal(11,2) DEFAULT NULL,
`total_nota_debito` decimal(11,2) DEFAULT NULL,
`enviado_sunat` char(1) COLLATE utf8_spanish_ci DEFAULT '0',
`code_respuesta_sunat` varchar(5) COLLATE utf8_spanish_ci DEFAULT NULL,
`descripcion_sunat_cdr` varchar(300) COLLATE utf8_spanish_ci DEFAULT NULL,
`name_file_sunat` varchar(80) COLLATE utf8_spanish_ci DEFAULT NULL,
`anulado` char(1) COLLATE utf8_spanish_ci DEFAULT '0' COMMENT 'comunicacion de baja',
`hash_cdr` varchar(150) COLLATE utf8_spanish_ci DEFAULT NULL,
`hash_cpe` varchar(150) COLLATE utf8_spanish_ci DEFAULT NULL,
`activo` int(11) DEFAULT 1
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_spanish_ci;

```

```

--
-- Estructura de tabla para la tabla `fac_nota_debito_detalle`
--

```

```

CREATE TABLE `fac_nota_debito_detalle` (
  `id_nota_debito_detalle` bigint(20) NOT NULL,
  `id_nota_debito` bigint(20) DEFAULT NULL,
  `id_local_producto` int(11) DEFAULT NULL COMMENT 'producto de un local',
  `producto` varchar(1000) COLLATE utf8_spanish_ci DEFAULT NULL,
  `cantidad` decimal(8,2) DEFAULT NULL,
  `precio_unitario` decimal(11,2) DEFAULT NULL,
  `sub_total` decimal(11,2) DEFAULT NULL,
  `total` decimal(11,2) DEFAULT NULL,
  `id_venta_detalle` int(11) DEFAULT NULL COMMENT 'detalle de la venta relacionada',
  `activo` int(11) DEFAULT 1
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_spanish_ci;

```

```

--
-- Estructura de tabla para la tabla `fac_tipo_nota_credito`
--

```

```

CREATE TABLE `fac_tipo_nota_credito` (
  `id_tipo_nota_credito` char(2) COLLATE utf8_spanish_ci NOT NULL,
  `tipo_nota_credito` varchar(100) COLLATE utf8_spanish_ci DEFAULT NULL,
  `activo` int(11) DEFAULT 1
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_spanish_ci;

```

```

--
-- Estructura de tabla para la tabla `fac_tipo_nota_debito`
--

```

```

CREATE TABLE `fac_tipo_nota_debito` (
  `id_tipo_nota_debito` char(2) COLLATE utf8_spanish_ci NOT NULL,
  `tipo_nota_debito` varchar(100) COLLATE utf8_spanish_ci DEFAULT NULL,
  `activo` int(11) DEFAULT 1
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_spanish_ci;

```

```

--
-- Estructura de tabla para la tabla `forma_de_pago`
--

```

```

CREATE TABLE `forma_de_pago` (
  `ideforma` int(2) NOT NULL,
  `descripcion` varchar(250) NOT NULL,
  `estado` char(1) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

```

--
-- Estructura de tabla para la tabla `forma_de_pago_mixto`
--

```

```

CREATE TABLE `forma_de_pago_mixto` (
  `id` int(11) NOT NULL,

```

```
`id_forma_de_pago` int(11) NOT NULL,  
`id_venta` int(11) NOT NULL,  
`idCaja` int(11) NOT NULL,  
`monto` decimal(10,2) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
--  
-- Estructura de tabla para la tabla `historial_canje_puntos`  
--
```

```
CREATE TABLE `historial_canje_puntos` (  
  `id_canje_puntos` int(11) NOT NULL,  
  `id_persona` int(11) NOT NULL,  
  `id_producto` int(11) NOT NULL,  
  `id_puntos` int(11) NOT NULL,  
  `cant_puntos` int(11) NOT NULL,  
  `a_puntos` int(11) NOT NULL,  
  `d_puntos` int(11) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
--  
-- Estructura de tabla para la tabla `kardex`  
--
```

```
CREATE TABLE `kardex` (  
  `id_kardex` int(11) NOT NULL,  
  `id_producto` int(11) DEFAULT NULL,  
  `id_venta` int(11) DEFAULT NULL,  
  `idacceso` int(11) DEFAULT NULL,  
  `id_compra` int(11) DEFAULT NULL,  
  `id_inventario` int(11) DEFAULT NULL,  
  `id_transferenciasalida` int(11) DEFAULT NULL,  
  `tipo` varchar(1) DEFAULT NULL,  
  `cantidad` int(10) DEFAULT NULL,
```

```

`pu` float(7,2) DEFAULT NULL,
`stock` int(10) DEFAULT NULL,
`fecha_emision` datetime DEFAULT NULL,
`numero_documento` varchar(100) DEFAULT NULL,
`destino` varchar(100) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

```

--
-- Estructura de tabla para la tabla `laboratorio`
--

```

```

CREATE TABLE `laboratorio` (
  `idlaboratorio` int(11) NOT NULL,
  `nombre` varchar(45) DEFAULT NULL,
  `descripcion` varchar(45) DEFAULT NULL,
  `estado` varchar(2) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

```

--
-- Estructura de tabla para la tabla `lote`
--

```

```

CREATE TABLE `lote` (
  `idlote` int(11) NOT NULL,
  `id_producto` int(11) DEFAULT NULL,
  `numero_lote` varchar(8) NOT NULL,
  `fecha_caduca_lote` date NOT NULL,
  `stock_lote` int(11) NOT NULL,
  `precio` decimal(15,2) NOT NULL,
  `estado_lote` varchar(2) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

```

--
-- Estructura de tabla para la tabla `menus`
--

CREATE TABLE `menus` (
  `id` int(11) NOT NULL,
  `nombre` varchar(45) DEFAULT NULL,
  `link` varchar(100) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

```

-----

--
-- Estructura de tabla para la tabla `permisos`
--

```

```

CREATE TABLE `permisos` (
  `id` int(11) NOT NULL,
  `menu_id` int(11) DEFAULT NULL,
  `rol_id` int(11) DEFAULT NULL,
  `read` int(11) DEFAULT NULL,
  `insert` int(11) DEFAULT NULL,
  `update` int(11) DEFAULT NULL,
  `delete` int(11) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

```

-----

--
-- Estructura de tabla para la tabla `persona`
--

```

```

CREATE TABLE `persona` (
  `idPersona` int(11) NOT NULL,
  `cedula` varchar(13) DEFAULT NULL,
  `tipo_documento` varchar(3) DEFAULT "",
  `apellidos` varchar(60) NOT NULL,
  `nombres` varchar(60) NOT NULL,

```

```

`razon_social` varchar(120) DEFAULT "",
`direccion` varchar(70) DEFAULT NULL,
`telefono` varchar(15) DEFAULT NULL,
`celular` varchar(15) DEFAULT NULL,
`email` varchar(70) DEFAULT NULL,
`imgPersona` text DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

```

-----
--
-- Estructura de tabla para la tabla `productos`
--

```

```

CREATE TABLE `productos` (
  `id` int(11) NOT NULL,
  `codigo` varchar(100) DEFAULT NULL,
  `nombre` varchar(100) DEFAULT NULL,
  `iva` int(11) DEFAULT NULL,
  `estado` varchar(2) DEFAULT NULL,
  `precio_costo` double DEFAULT NULL,
  `precio_costo_unitario` decimal(20,2) DEFAULT NULL,
  `por_gan` float DEFAULT NULL,
  `precio_venta` double DEFAULT NULL,
  `pvp1` double DEFAULT NULL,
  `pvp2` double DEFAULT NULL,
  `pvp3` double DEFAULT NULL,
  `pblistier` double NOT NULL DEFAULT 0,
  `cblister` int(11) NOT NULL DEFAULT 0,
  `stock` int(11) DEFAULT NULL,
  `stock_min` int(11) NOT NULL DEFAULT 0,
  `stock_caja` int(11) DEFAULT NULL,
  `unidades_caja` int(11) DEFAULT NULL,
  `categoria_id` int(11) DEFAULT NULL,
  `laboratorio_id` int(11) DEFAULT NULL,
  `img` varchar(255) DEFAULT NULL,
  `descuento` int(11) DEFAULT NULL,
  `para_que_sirve` text DEFAULT NULL,

```

```

`fecha_caducidad` date DEFAULT NULL,
`p_activo` varchar(25) NOT NULL,
`c_digemid` varchar(25) NOT NULL,
`t_producto` varchar(25) DEFAULT NULL,
`r_sanitario` varchar(15) DEFAULT NULL,
`receta` int(2) DEFAULT NULL,
`por_com` double NOT NULL DEFAULT 0,
`efectivo_com` double DEFAULT 0,
`ind_lote` char(5) NOT NULL DEFAULT 'n',
`tipo_atencion` char(5) NOT NULL DEFAULT 'p',
`canje_puntos` int(2) DEFAULT NULL,
`id_ubicacion_producto` int(11) DEFAULT NULL,
`excluye_comision` int(2) DEFAULT 0,
`id_unidad_de_medida` int(11) DEFAULT NULL,
`puntaje_medico` int(11) DEFAULT 0
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

--

-- Estructura de tabla para la tabla `promociones`

--

```

CREATE TABLE `promociones` (
  `idpromociones` int(11) NOT NULL,
  `descripcion` varchar(80) DEFAULT NULL,
  `fecha_inicio` date DEFAULT NULL,
  `fecha_fin` date DEFAULT NULL,
  `idproducto_comprado` int(11) DEFAULT NULL,
  `id_producto_promocion` int(11) DEFAULT NULL,
  `cant_comprada` int(11) DEFAULT NULL,
  `cant_promocion` int(11) DEFAULT NULL,
  `estado` varchar(2) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

--

```
-- Estructura de tabla para la tabla `puntos_persona`  
--
```

```
CREATE TABLE `puntos_persona` (  
  `id_puntos_persona` int(11) NOT NULL,  
  `id_persona` int(11) NOT NULL,  
  `nroPuntos` int(11) DEFAULT NULL,  
  `nroSolesxPunto` decimal(20,2) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
-----
```

```
--  
-- Estructura de tabla para la tabla `resumen_sunat`  
--
```

```
CREATE TABLE `resumen_sunat` (  
  `idResumen` bigint(20) NOT NULL,  
  `fechaResumen` datetime DEFAULT NULL,  
  `nombreXML` varchar(100) DEFAULT NULL,  
  `ticket` varchar(50) DEFAULT NULL,  
  `mensaje` varchar(150) DEFAULT NULL,  
  `items` bigint(20) DEFAULT NULL,  
  `fechaEnvio` datetime DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
-----
```

```
--  
-- Estructura de tabla para la tabla `rol`  
--
```

```
CREATE TABLE `rol` (  
  `idrol` int(11) NOT NULL,  
  `nombre` varchar(45) DEFAULT NULL,  
  `descripcion` varchar(45) DEFAULT NULL,  
  `comision` decimal(20,2) NOT NULL DEFAULT 0.00  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```

-----

--
-- Estructura de tabla para la tabla `tipo_igv`
--

CREATE TABLE `tipo_igv` (
  `id_tipo_igv` char(2) COLLATE utf8_spanish_ci NOT NULL,
  `tipo_igv` varchar(70) COLLATE utf8_spanish_ci DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_spanish_ci;

```

```

-----

--
-- Estructura de tabla para la tabla `tipo_moneda`
--

CREATE TABLE `tipo_moneda` (
  `id_tipo_moneda` int(11) NOT NULL,
  `codigo` char(3) COLLATE utf8_spanish_ci DEFAULT NULL,
  `tipo_moneda` varchar(50) COLLATE utf8_spanish_ci DEFAULT NULL,
  `simbolo` varchar(5) COLLATE utf8_spanish_ci DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_spanish_ci;

```

```

-----

--
-- Estructura de tabla para la tabla `tipo_servicio`
--

CREATE TABLE `tipo_servicio` (
  `idTipoServicio` int(11) NOT NULL,
  `nombre_servicio` varchar(100) NOT NULL,
  `estado` int(1) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

```
--  
-- Estructura de tabla para la tabla `tipo_venta`  
--
```

```
CREATE TABLE `tipo_venta` (  
  `id` int(11) NOT NULL,  
  `nombre` varchar(45) DEFAULT NULL,  
  `cantidad` int(11) DEFAULT NULL,  
  `serie` varchar(45) DEFAULT NULL,  
  `numero_inicial` varchar(11) DEFAULT NULL,  
  `numero_final` varchar(11) DEFAULT NULL,  
  `iva` float DEFAULT NULL,  
  `estado` varchar(2) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
--  
-- Estructura de tabla para la tabla `tipo_venta_series`  
--
```

```
CREATE TABLE `tipo_venta_series` (  
  `id_serie` int(11) NOT NULL,  
  `id_tipo_venta` int(11) NOT NULL,  
  `valor_sunat` varchar(2) DEFAULT NULL,  
  `cantidad` int(11) DEFAULT NULL,  
  `serie` varchar(45) DEFAULT NULL,  
  `numero_inicial` varchar(11) DEFAULT NULL,  
  `numero_final` varchar(11) DEFAULT NULL,  
  `estado` varchar(2) DEFAULT NULL,  
  `id_sucursal` int(11) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
--  
-- Estructura de tabla para la tabla `ventas`
```

--

```
CREATE TABLE `ventas` (  
  `id` int(11) NOT NULL,  
  `fecha` datetime DEFAULT NULL,  
  `fecha_sunat` datetime DEFAULT NULL,  
  `subtotal` decimal(20,2) NOT NULL DEFAULT 0.00,  
  `subtotal12` decimal(20,2) DEFAULT 0.00,  
  `iva` decimal(20,2) DEFAULT 0.00,  
  `descuento` decimal(20,2) DEFAULT 0.00,  
  `total` decimal(20,2) DEFAULT 0.00,  
  `recibido` decimal(20,2) DEFAULT 0.00,  
  `cambio` decimal(20,2) DEFAULT 0.00,  
  `bolsa` decimal(20,2) NOT NULL DEFAULT 0.00,  
  `tipo_venta_id` int(11) DEFAULT NULL,  
  `forma_pago_id` int(11) DEFAULT NULL,  
  `idCliente` int(11) DEFAULT NULL,  
  `idacceso` int(11) DEFAULT NULL,  
  `idCaja` int(11) NOT NULL DEFAULT 1,  
  `num_venta_id` varchar(45) DEFAULT NULL,  
  `serie` varchar(10) DEFAULT NULL,  
  `aceptado` varchar(60) DEFAULT "",  
  `estado` int(2) DEFAULT NULL,  
  `id_tipo_moneda` int(11) DEFAULT NULL,  
  `tipo_cambio` decimal(11,2) DEFAULT NULL,  
  `total_exonerada` decimal(11,2) DEFAULT NULL,  
  `total_inafecta` decimal(11,2) DEFAULT NULL,  
  `total_gravada` decimal(11,2) DEFAULT NULL,  
  `total_igv` decimal(11,2) DEFAULT NULL,  
  `total_gratuita` decimal(11,2) DEFAULT NULL,  
  `nroPuntos` int(11) DEFAULT NULL,  
  `puntoGanados` int(11) DEFAULT NULL,  
  `por_venta` double NOT NULL DEFAULT 0,  
  `tot_porventa` double NOT NULL DEFAULT 0,  
  `porcentaje_com_rol` decimal(20,2) DEFAULT 0.00,  
  `total_com_rol` decimal(20,2) DEFAULT 0.00,  
  `id_medico` int(11) DEFAULT 0,  
  `puntaje_medico_total` int(11) DEFAULT 0,
```

```

`pago_deuda` double DEFAULT 0
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

6.11 Diagrama de clases del sistema

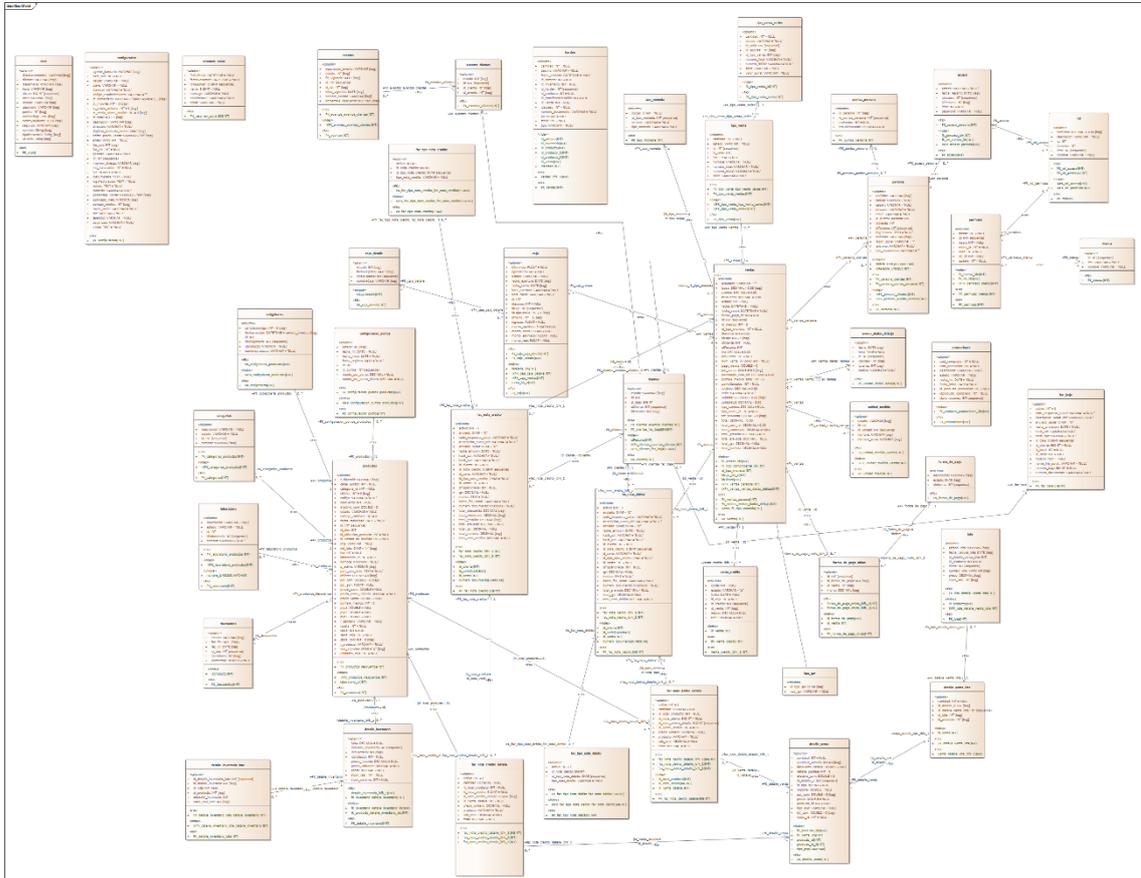


Figura 57. Diagrama de clases
Fuente: Elaboración propia.

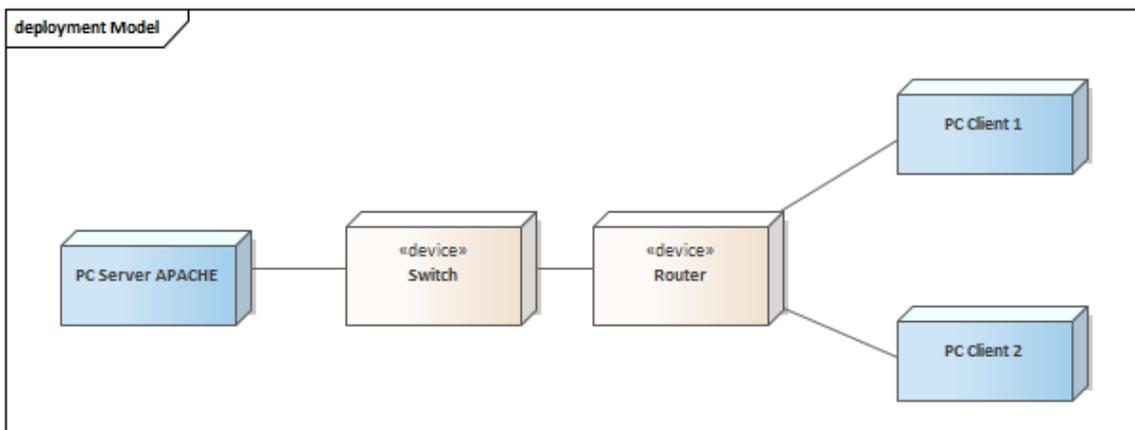


Figura 58. Diagrama de despliegue de la aplicación
Fuente: Elaboración Propia.

Conclusiones

- Se analizó e implementó el sistema “IpharmaExpress” para automatizar el proceso de ventas para la Botica “El Gran Poder”, reduciendo el tiempo de atención según el diagrama de actividades ASIS vs TOBE en un 50%.
- Se realizó el análisis de calidad de software aplicando la norma ISO/IEC 25000 con “**Adecuación Funcional**” enfocado para la calidad interna donde se obtuvo un puntaje 10/10 y para la calidad externa donde se obtuvo 10/10 obteniendo nivel de puntuación: “**cumple con los requisitos**” y grado de satisfacción: “**muy satisfactorio**”, en ambos casos.
- Se realizó el análisis de calidad de software aplicando la norma ISO/IEC 25000 con “**Fiabilidad**” enfocado para la calidad interna donde se obtuvo un puntaje 10/10 y para la calidad externa donde se obtuvo 8,75/10 obteniendo nivel de puntuación: “**cumple con los requisitos**” y grado de satisfacción: “**muy satisfactorio**”, en ambos casos.
- Se realizó el análisis de calidad de software aplicando la norma ISO/IEC 25000 con “**Eficiencia en el desempeño**” enfocado para la calidad interna donde se obtuvo un puntaje 5,60/10 obteniendo nivel de puntuación: “**aceptable**” y un grado de satisfacción: “**satisfactorio**” y para la calidad externa donde se obtuvo 9,00/10 obteniendo nivel de puntuación: “**cumple con los requisitos**” y grado de satisfacción: “**muy satisfactorio**”.
- Se realizó el análisis de calidad de software aplicando la norma ISO/IEC 25000 con “**Usabilidad**” enfocado para la calidad interna donde se obtuvo un puntaje 7,53/10 y para la calidad externa donde se obtuvo 8,75/10 obteniendo nivel de puntuación: “**aceptable**” y un grado de satisfacción de: “**satisfactorio**”, en ambos casos.
- Se realizó el análisis de calidad de software aplicando la norma ISO/IEC 25000 con “**Mantenibilidad**” enfocado para la calidad interna donde se obtuvo un puntaje 6,92/10 obteniendo nivel de puntuación: “**aceptable**” y un grado de satisfacción: “**satisfactorio**” para la calidad externa donde se obtuvo 10/10 obteniendo nivel de puntuación: “**cumple con los requisitos**” y un grado de satisfacción de: “**muy satisfactorio**”.

- Se realizó el análisis de calidad de software aplicando la norma ISO/IEC 25000 a los atributos de **“Adecuación Funcional, Fiabilidad, Eficiencia en el Desempeño, Usabilidad, Mantenibilidad”** enfocados en la calidad interna y externa del sistema web Ipharmaexpress obteniendo un 7,55 y 9,08 de puntaje respectivamente, haciendo un valor total de 8,31/10 lo que representa que el sistema tuvo un nivel de puntuación: **“aceptable”** y un grado de satisfacción: **“satisfactorio”**.

Observaciones

- No se encontraron observaciones en el desarrollo de la tesis.

Recomendaciones y trabajos futuros

- Para contar con un enfoque disciplinado en la asignación de tareas y responsabilidades dentro de una organización del desarrollo, es necesaria la aplicación de una metodología, con la cual se puede mantener una fácil administración de este proceso; como por ejemplo la metodología RUP.
- Optimizar el código fuente para incrementar la puntuación de la calidad interna.
- En la definición de métricas del sistema se pueden considerar otras métricas que no estas determinadas en la norma ISO/IEC 25000, siempre y cuando estas nuevas métricas se adapten al modelo establecido.
- Se recomienda aplicar el modelo de calidad de ISO 25000 en cualquier tipo de software no solo a un software en específico, siempre y cuando se escoja las variables más adecuadas para la evaluación y de acuerdo con las características propias del producto de software.
- El software “IpharmaExpress” tiene oportunidades de mejora respecto a incluir más modulo al sistemas por ejemplo el módulo de gestión de compras.
- Se puede incluir en la medición de calidad del software la parte de ISO/IEC 25010 SQuaRE con respecto a la calidad en uso

REFERENCIAS BIBLIOGRÁFICAS

- Vivanco, A. (2011) *Evaluación de calidad del sistema integrado para casas de valores SICAV de la bolsa de valores de Quito utilizando la norma ISO/IEC 14598*, “pregrado” de la Universidad ESCUELA POLITÉCNICA NACIONAL, Quito – Ecuador.
- Arana, Q. Valeria, J. (2014). *Desarrollo e implementación de un sistema de gestión de venta de repuestos automotrices en el almacén de auto repuestos eléctricos marcos en la parroquia posorja cantón guayaquil, provincia de las guayas*, “pregrado” de la Universidad ESTATAL PENÍNSULA DE SANTA ELENA - Ecuador
- Pacuanca, F. (2014). *Sistema web de venta de boletos para las empresas de transporte de la terminal de buses la paz, caso: soluciones de tecnología en ingeniería de sistemas y telecomunicaciones*, “pregrado” de la Universidad MAYOR DE SAN ANDRÉS - Bolivia.
- Bendezú, C. (2017). *Sistema web para el proceso de ventas en La Botica Helífarma E.I.R.L*, Lima, “pre grado” de la Universidad CESAR VALLEJO – Lima.
- Gamboa J, Guillermo R (2017). *Implementación de un sistema web para las ventas en la empresa one to one contact solutions*, “pre pregado” de la Universidad SAN IGNACIO DEL OYOLA – Lima.
- Valles, S. (2015). *Diseño e implementación de un sistema de venta para la distribuidora Josymar trujillo*, “pre grado” de la Universidad UNIVERSIDAD CATÓLICA LOS ÁNGELES DE CHIMBOTE – Chimbote.
- Balseca, E. (2014). *Evaluación de calidad de productos de software en empresas de desarrollo de software aplicando la norma ISO/IEC 25000*, “pre grado” de la Universidad ESCUELA POLITECNICA NACIONAL, Quito – Ecuador.
- ISO/IEC 25010, (2016) «*Systems and software engineering,*» of *Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models*, Estandar Internacional: <https://www.iso.org/>

ISO/IEC 25023, (2016) «*Required Quality Measures,*» of *Systems and software Quality Requirements and Evaluation (SQuaRE) – Measurement of system and software product quality, acquirers and independent evaluators*, Estandar Internacional: <https://www.iso.org/>.

ISO/IEC 25040, (2016) «*Software Product Quality Evaluation Reference Model,*» of *Software engineering - Software product Quality Requirements and Evaluation (SQuaRE) – Evaluation reference model and guide*, Estandar Internacional: <https://www.iso.org/>.

Moreno, J (2015). *Programación Orientada a Objetos*. Madrid, España: Grupo Editorial RA-MA

Noriega, R (2017), *El proceso de desarrollo de software: 2da Edición*. Madrid, España: IT Campus Academy.

Kruchten, P. (2004), *The rational unified process: an introduction*. University of British Columbia – Vancouver: Editorial Addison-Wesley

Martinez, R. (2014). *ISO 25000 y el software actual*. Recuperado de: ["https://es.slideshare.net/rmartinez582/iso-25000-anexo-calidad-de-producto-v3-39659946"](https://es.slideshare.net/rmartinez582/iso-25000-anexo-calidad-de-producto-v3-39659946)

Gómez, R. (2015), *Modelo Vista Controlador*, Recuperado de: <http://rodrigogr.com/blog/modelo-vista-controlador/>

Slideplayer (2014), *Arquitectura Web PHP*, Recuperado de: https://images.slideplayer.es/42/11517137/slides/slide_2.jpg

Wikipedia.com (2018), *Proceso Unificado de Rational*, Recuperado de: https://es.wikipedia.org/wiki/Proceso_Unificado_de_Rational#/media/Archivo:Rup_espanol.gif

iso25000.com (2019), *ISO 25010*, Recuperado de: <https://iso25000.com/index.php/normas-iso-25000>

ANEXOS

Anexo 1: Matriz de consistencia

Problemas	Objetivos	Variables		Metodología
		Variable Independiente	Dimensiones	
General	General	Desarrollo de un sistema web	Adecuación Funcional Fiabilidad Eficiencia en desempeño Usabilidad Mantenibilidad	La metodología de desarrollo de software a aplicar es el Rational Unified Process (RUP), de acuerdo con lo propuesto en este proyecto. Asimismo, se aplicará la norma ISO/IEC 25000 Square, estándar que permite especificar y evaluar la calidad del software desde diferentes criterios, los mismo que serán evaluados bajo los indicadores de eficiencia de desempeño usabilidad y mantenibilidad.
¿En qué medida del desarrollo del sistema web mejorara el proceso de venta de la Botica el Gran Poder, Carabayllo, Lima 2020?	Determinar la mejora del desarrollo del sistema web en el proceso de venta de la Botica el Gran Poder, Carabayllo, Lima 2020			
Especifico	Especifico	Variable dependiente	Dimensión	
¿En qué medida el estándar de, adecuación funcional permite mejorar el proceso de ventas de la Botica el Gran Poder Carabayllo, Lima 2020?	Determinar la mejora con adecuación funcional el proceso de ventas de la Botica el Gran Poder Carabayllo, Lima 2020.	Proceso de ventas	Clientes Atención	
¿En qué medida el estándar de, fiabilidad permite mejorar el proceso de ventas de la Botica el Gran Poder Carabayllo, Lima 2020?	Determinar la mejora con fiabilidad el proceso de ventas de la Botica el Gran Poder Carabayllo, Lima 2020.			
¿En qué medida el estándar de, eficiencia de desempeño permite mejorar el proceso de ventas de la Botica el Gran Poder Carabayllo, Lima 2020?	Determinar la mejora con eficiencia de desempeño el proceso de ventas de la Botica el Gran Poder Carabayllo, Lima 2020.			
¿En qué medida el estándar de, usabilidad permite mejorar el proceso de ventas de la Botica el Gran Poder Carabayllo, Lima 2020?	Determinar la mejora con usabilidad el proceso de ventas de la Botica el Gran Poder Carabayllo, Lima 2020.			
¿En qué medida el estándar de, mantenibilidad permite mejorar el proceso de ventas de la Botica el Gran Poder Carabayllo, Lima 2020?	Determinar la mejora con mantenibilidad el proceso de ventas de la Botica el Gran Poder Carabayllo, Lima 2020.			

Anexo 2: Instrumentos de aplicación

Anexo 2.1. Métricas para la calidad Interna y Externa características: Adecuación funcional, Fiabilidad, Eficiencia en desempeño, Usabilidad y Mantenibilidad.

Métricas para la característica de calidad Adecuación funcional								
Subcaracterística	Métrica	Fase del ciclo de vida de calidad del producto	Propósito de la métrica de calidad	Método de aplicación	Fórmula	Valor deseado	Tipo de medida	Recursos utilizados
Compleitud funcional	Compleitud de la implementación funcional	Interna/Externa	¿Cuán completa es la implementación de acuerdo a la especificación de requerimientos?	Contar el número de las funciones indicadas en la especificación de requerimientos y el número de funciones que faltan o están incorrectas	$X = A / B$ A = Número de funciones que están incorrectas o que no fueron implementadas B = Número de las funciones establecidas en la especificación de requisitos Dónde: $B > 0$	$0 \leq X \leq 1$ El más cercano a 0 es el mejor	x=contable/contable A = Contable B = Contable	Especificación de requerimientos, Código fuente, Desarrollador, Tester

Exactitud funcional	Exactitud	Interna/Externa	¿Cuánto del estándar requerido de exactitud se cumple?	Contar el número de elementos de datos implementados con el estándar específico de exactitud y el número total de elementos de datos implementados	$X = A / B$ A = Número de elementos de datos implementados con el estándar específico de exactitud B = Número total, de elementos de datos implementados Dónde: $B > 0$	$0 \leq X \leq 1$ El más cercano a 1 es el mejor	$x = \text{contable} / \text{contable}$ A = Contable B = Contable	Especificación de requerimientos, Código fuente, Desarrollador, Tester
	Precisión computacional	Interna/Externa	¿Con qué frecuencia ocurren los resultados inexactos?	Contar el número de cálculos inexactos encontrados y tomar el tiempo de operación	$X = A/T$ A = Numero de cálculos inexactos encontrados T = Tiempo de operación Dónde: $T > 0$	$X = A/T$ El más cercano a 0/t es el mejor. Donde el peor caso es $\geq 10/t$.	$X = \text{Contable} / \text{Tiempo}$ A=Contable B=Tiempo	Código fuente, Desarrollador, Tester

Tabla 24. Métricas para la característica de calidad de Educación funcional

Fuente: Balseca (2014)

Métricas para la característica de calidad Fiabilidad

Subcaracterística	Métrica	Fase del ciclo de vida de calidad del producto	Propósito de la métrica de calidad	Método de aplicación	Fórmula	Valor deseado	Tipo de medida	Recursos utilizados
	Eliminación de errores	Interna/Externa	¿Cuántos errores detectados han sido corregidos?	Contar el número de fallas corregidas en la fase de diseño/codificación/pruebas y el número de fallas detectadas en las pruebas	$X = A / B$ A = Número de fallas corregidas en la fase de diseño/codificación/pruebas B = Número de fallas detectadas en las pruebas Dónde: $B > 0$	$0 \leq X \leq 1$ Cuanto más se acerque a 1 es lo mejor	x=contable/contable A = Contable B = Contable	Especificación De requerimientos, Documento de casos de pruebas, Código fuente, Desarrollador, Tester

Madurez	Cobertura de pruebas	Interna/Externa	¿Cuántos casos de prueba requeridos han sido ejecutados durante la etapa de pruebas?	Contar el número de casos de pruebas realizados en un escenario de operación durante la prueba y el número de casos de prueba a ser realizados para cubrir los requerimientos	$X = A / B$ A = Número de casos de pruebas realizados en un escenario de operación durante la prueba B = Número de casos de prueba a ser realizados para cubrir los requerimientos Dónde: $B > 0$	$0 \leq X \leq 1$ El más cercano a 1 es el mejor	$x = \text{contable} / \text{contable}$ A = Contable B = Contable	Especificación de requerimientos, documento de casos de pruebas, código fuente, Desarrollador, Tester
	Tiempo medio entre fallos	Externa	¿Cuál es la frecuencia en que el sistema falla en la operación?	Tomar el tiempo de operación y contar el número total de fallas detectadas actualmente	$X = A / T$ A = Número total de fallas detectadas actualmente T = Tiempo de operación Donde $T > 0$	$X = A / T$ El más cercano a 0/t es el mejor	$X = \text{Contable} / \text{Tiempo}$ A = Contable B = Contable	Especificación de requerimientos, Desarrollador, Tester

Disponibilidad	Tiempo de servicio	Externa	¿Cuál es el tiempo de servicio del sistema que proporciona realmente?	Tomar el tiempo de servicio del sistema que se proporciona actualmente y tomar el tiempo de servicio del sistema regulado en el cronograma operacional	$X = A / B$ A = Tiempo de servicio del sistema que se proporciona actualmente B = Tiempo de servicio del sistema regulado en el cronograma operacional Dónde: $B > 0$	$0 \leq X \leq 1$ Cuanto más se acerque a 1 es lo mejor	X= Tiempo/ Tiempo A = Tiempo B = Tiempo	Especificación de requerimientos, Desarrollador, Tester
	Tiempo medio de inactividad	Externa	¿Cuál es el tiempo promedio que el sistema está inactivo después de que ocurre un fallo?	Tomar el tiempo total de inactividad y contar el número de fallos observados	$X = A / T$ A = Número de fallos observados T = Tiempo total de inactividad Dónde: $T > 0$	$X = A/T$ El más cercano a 0/t es el mejor	X=Contable/ Tiempo A = Tiempo B = Contable	Especificación de requerimientos, Desarrollador, Tester

Tolerancia a fallos	Prevención de fallas	Externa	¿Cuántas fallas iniciales estuvieron bajo control para evitar fallas serias y críticas?	Contar el número de ocurrencia de fallas serias y críticas evitadas contra los casos de pruebas de fallas iniciales y el número de casos de pruebas de fallas iniciales ejecutados durante las pruebas	$X = A / B$ $A =$ Número de ocurrencia de fallas evitadas contra los casos de pruebas de fallas iniciales $B =$ Número de casos de pruebas de fallas iniciales ejecutados durante las pruebas Dónde: $B > 0$	$0 \leq X \leq 1$ Cuanto más se acerque a 1 es lo mejor	$x = \text{contable} / \text{contable}$ $A = \text{Contable}$ $B = \text{Contable}$	Especificación de requerimientos, Desarrollador, Tester
	Redundancia	Interna/Externa	¿Cuántos tipos de componentes/sistemas del son instalados de forma redundante para evitar un fallo en el sistema?	Contar el número total de tipos de componentes y el número de tipos de componentes instalados de forma redundante	$X = A / B$ $A =$ Número componentes /sistemas instalados de forma redundante $B =$ Número total de componentes /sistemas instalados Dónde: $B > 0$	$0 \leq X \leq 1$ Cuanto más se acerque a 1 es lo mejor	$X = \text{contable} / \text{contable}$ $A = \text{Contable}$ $B = \text{Contable}$	Especificación de requerimientos, Código fuente, Desarrollador, Tester

	Anulación de operación incorrecta	Interna	¿Cuántas funciones son implementadas con capacidad de anular operaciones incorrectas?	Contar el número de funciones implementadas que evitan fallas críticas y serias causadas por operaciones incorrectas y contar el número de operaciones incorrectas presentadas	$X = A / B$ A = Número de operaciones incorrectas presentadas B = Número total de funciones implementadas para anular operaciones incorrectas Dónde: $B > 0$	$0 \leq X \leq 1$ Cuanto más se acerque a 0 es lo mejor	$x = \text{contable} / \text{contable}$ A = Contable B = Contable	Especificación de requerimientos, Código fuente, Desarrollador
Recuperabilidad	Tiempo medio de recuperación	Interna/Externa	¿Cuál es el tiempo promedio que toma el sistema en recuperarse completamente después un fallo?	Tomar el tiempo que le tomó al sistema en recuperarse y contar el número de casos en los cuales se ha observado que el sistema entró en recuperación	$X = A / T$ A = Número de casos en los cuales se ha observado que el sistema entró en recuperación T = Tiempo que le tomó al sistema en recuperarse Dónde: $T > 0$	$X = A/T$ El más cercano a 0/t es el mejor. Donde el peor caso es $\geq 10/t$.	$X = \text{Contable} / \text{Tiempo}$ A = Contable B = Tiempo	Especificación de requerimientos, Código fuente, Desarrollador, Tester

Tabla 25. Métricas para la característica de calidad Fiabilidad

Fuente: Balseca (2014)

Métricas para la característica de calidad Eficiencia en el desempeño

Subcaracterística	Métrica	Fase del ciclo de vida de calidad del producto	Propósito de la métrica de calidad	Método de aplicación	Fórmula	Valor deseado	Tipo de medida	Recursos utilizados
Comportamiento del tiempo	Tiempo de respuesta	Interna/Externa	¿Cuál es el tiempo estimado para completar una tarea?	Tomar el tiempo desde que se envía la petición hasta obtener la respuesta	$X = B - A$ A= Tiempo de envío de petición B = Tiempo en recibir la primera respuesta	$0 \leq X \leq 1$ El más cercano a 0 es el mejor. Donde el peor caso es $\geq 15t$.	$X = \text{Tiempo} / \text{Tiempo}$ A= Tiempo B= Tiempo	Especificación de requerimientos, Código fuente, Desarrollador, Tester
	Tiempo de espera	Interna/Externa	¿Cuál es el tiempo desde que se envía una instrucción, para que inicie un trabajo, hasta que lo completa?	Tomar el tiempo cuando se inicia un trabajo y el tiempo en completar el trabajo	$X = B - A$ A= Tiempo cuando se inicia un trabajo B = Tiempo en completar el trabajo	$0 \leq X \leq 1$ El más cercano a 0 es el mejor. Donde el peor caso es $\geq 15t$.	$X = \text{Tiempo} / \text{Tiempo}$ A=Tiempo B=Tiempo	Especificación de requerimientos, Código fuente, Desarrollador, Tester

	Rendimiento	Interna/Externa	¿Cuántas tareas pueden ser procesadas por unidad de tiempo?	Contar el número de tareas completadas en un intervalo de tiempo	$X = A/T$ A= Número de tareas completadas T = Intervalo de tiempo Dónde: $T > 0$	$X = A/T$ El más lejano a $0/t$ es el mejor. Donde el mejor caso es $\geq 10/t$	X=Contable/ Tiempo A= Contable T= Tiempo	Especificación de requerimientos, Código fuente, Desarrollador, Tester
Utilización de recursos	Líneas de código	Interna	¿Cuántas líneas de código existen por cada función implementada?	Contar el número de líneas de código (sin tomar en cuenta espacios ni comentarios) que existen en una determinada función.	$X = A$ A = Número de líneas de código	$1 \leq X \leq 50$ El más cercano a 1 es el mejor. Donde el peor caso es ≥ 50 líneas de código	X=Contable A=Contable	Código fuente
	Utilización de CPU	Interna/Externa	¿Cuánto tiempo de CPU es usado para realizar una tarea dada?	Tomar el tiempo de operación y la cantidad de tiempo de CPU que se usa para realizar una tarea	$X = B - A$ A = La cantidad de tiempo de CPU que realmente es usado para realizar una tarea B = Tiempo de operación Dónde: $B > 0$	$0 \leq X \leq 1$ Cuanto más se acerque a 0 es lo mejor. Donde el peor caso es $\geq 15t$.	X=Tiempo/Tiempo A=Tiempo B=Tiempo	Especificación de requerimientos, Código fuente, Desarrollador, Tester

	Utilización de la memoria	Interna/Externa	¿Cuánto espacio de memoria es usado para realizar una tarea dada?	Medir la cantidad total de espacios de memoria y la cantidad de espacios de memoria que realmente es usado para realizar una tarea	$X = B - A$ $A =$ Cantidad de espacios de memoria que realmente es usado para realizar una tarea $B =$ Cantidad total de espacios de memoria Dónde: $B > 0$	$0 \leq X \leq 15$ El más cercano a 0 es el mejor	$X =$ Tamaño/Tamaño $A =$ Tamaño $B =$ Tamaño	Especificación de requerimientos, Código fuente, Desarrollador, Tester
	Utilización de los dispositivos de E/S	Interna/Externa	¿Cuánto tiempo los dispositivos de E/S utilizan para realizar una tarea?	Tomar el tiempo de operación y el tiempo que los dispositivos de E/S pasan ocupados para realizar la tarea	$X = B - A$ $A =$ Tiempo que los dispositivos de E/S pasan ocupados para realizar la tarea $B =$ Tiempo de Operación Dónde: $B > 0$	$0 \leq X \leq 15$ El más cercano a 0 es el mejor	$X =$ Tiempo/Tiempo $A =$ Tiempo $B =$ Tiempo	Especificación de requerimientos, Código fuente, Desarrollador, Tester
Capacidad	Número de peticiones online	Interna/Externa	¿Cuántas peticiones online pueden ser procesadas por unidad de tiempo?	Contar el número máximo de peticiones online procesadas y tomar el tiempo de operación	$X = A/T$ $A =$ Número máximo de peticiones online Procesada $T =$ Tiempo de operación Dónde: $T > 0$	$X = A/T$ El más lejano a 0/t es el mejor. Donde el mejor caso es $\geq 10/t$	$X =$ Contable/Tiempo $A =$ Contable $T =$ Tiempo	Especificación de requerimientos, Código fuente, Desarrollador, Tester

	Número de accesos simultáneos	Interna/Externa	¿Cuántos usuarios pueden acceder al sistema simultáneamente en un cierto tiempo?	Contar el número máximo de accesos simultáneos y tomar el tiempo de operación	$X = A / T$ A= Número máximo de accesos simultáneos T = Tiempo de operación Dónde: $T > 0$	$X = A / T$ El más lejano a 0/t es el mejor. Donde el mejor caso es $\geq 10/t$	X= Contable/Tiempo A = Contable B = Tiempo	Especificación de requerimientos, Código fuente, Desarrollador, Tester
	Sistema de transmisión de ancho de banda	Externa	¿Cuánto es el valor límite absoluto de transmisión necesaria para cumplir con las funciones?	Contar la cantidad máxima de transmisión de datos y tomar el tiempo de operación	$X = A / T$ A= Cantidad máxima de transmisión de datos B = Tiempo de operación Dónde: $T > 0$	$X = A / T$ El más lejano a 0/t es el mejor. Donde el mejor caso es $\geq 10/t$	X = Tamaño/Tiempo A = Tamaño B = Tiempo	Especificación de requerimientos, Código fuente, Desarrollador, Tester

Tabla 26. Métricas para la característica de calidad Eficiencia en el desempeño

Fuente: Balseca (2014)

Métricas para la característica de calidad Usabilidad

Subcaracterística	Métrica	Fase del ciclo de vida de calidad del producto	Propósito de la métrica de calidad	Método de aplicación	Fórmula	Valor deseado	Tipo de medida	Recursos utilizados
Capacidad de reconocer su adecuación	Integridad de descripción	Interna/Externa	¿Qué cantidad de funciones (o tipos de funciones) son descritas como entendibles en la descripción del producto?	Contar el número de funciones (o tipos de funciones) descritas como entendibles en la descripción del producto y contar el número total de funciones (o tipos de funciones)	$X = A / B$ A = Número de funciones (o tipos de funciones) descritas como entendibles en la descripción del producto B = Número total de funciones (o tipos de funciones) Dónde: $B > 0$	$0 \leq X \leq 1$ El más cercano a 1 es el mejor	X=contable/contable A = Contable B = Contable	Especificación de requerimientos, Código fuente, Desarrollador, Tester
	Capacidad de demostración	Interna/Externa	¿Qué cantidad de funciones tienen la capacidad de demostración?	Contar el número de funciones implementadas con capacidad de demostración y contar el número total de funciones que	$X = A / B$ A = Número de funciones implementadas con capacidad de demostración B = Número total, de	$0 \leq X \leq 1$ El más cercano a 1 es el mejor	X=contable/contable A = Contable B = Contable	Especificación de requerimientos, Código fuente, Desarrollador, Tester

				requieren capacidad de demostración	funciones que requieren capacidad de demostración Dónde: $B > 0$			
Capacidad para ser entendido	Funciones evidentes	Interna	¿Qué cantidad de funciones del producto son evidentes al usuario?	Contar el número de funciones que son evidentes al usuario y comparar con el número total de funciones.	$X = A / B$ A= Número de funciones (o tipo de funciones) evidentes al usuario B = Número total de funciones (o tipo de funciones) Dónde: $B > 0$	$0 \leq X \leq 1$ El más cercano a 1 es el mejor	$X = \text{contable}/\text{contable}$ A = Contable B = Contable	Código fuente
	Efectividad de La documentación del usuario o ayuda del sistema	Interna/Externa	¿Qué cantidad de funciones está descritas correctamente en la documentación del usuario o ayuda en línea?	Contar el número de funciones descritas correctamente y contar el número total de funciones implementadas	$X = A / B$ A= Número de funciones descritas correctamente B = Número total de funciones implementadas Dónde: $B > 0$	$0 \leq X \leq 1$ El más cercano a 1 es el mejor	$X = \text{contable}/\text{contable}$ A = Contable B = Contable	Especificación de requerimientos, Código fuente, Desarrollador, Tester

Operatividad	Recuperabilidad de error operacional	Interna	¿Qué cantidad de funciones pueden tolerar errores de usuario?	Contar número de funciones implementadas con tolerancia de error de usuarios y el número total de funciones requeridas con capacidad de tolerancia.	$X = A / B$ A= Número de funciones implementadas con tolerancia de error de usuarios B = Número total de funciones requeridas con capacidad de tolerancia. Dónde: $B > 0$	$0 \leq X \leq 1$ El más cercano a 1 es el mejor	$X = \text{contable} / \text{contable}$ A = Contable B = Contable	Código fuente
	Claridad del mensaje	Interna/Externa	¿Qué cantidad de mensajes son auto explicativo?	Contar el número de mensajes implementados con explicaciones claras y el número total de mensajes implementados	$X = A / B$ A= Número de mensajes implementados con explicaciones claras B = Número total de mensajes implementados Dónde: $B > 0$	$0 \leq X \leq 1$ El más cercano a 1 es el mejor	$X = \text{contable} / \text{contable}$ A = Contable B = Contable	Especificación de requerimientos, Código fuente, Desarrollador, Tester

	Consistencia operacional	Interna/Externa	¿Cuánta operaciones similares pueden llevarse a cabo consecuentemente ?	Contar el número de operaciones que se comportan de manera incoherente y el número total de operaciones que se comportan de forma normal	$X = A / B$ A= Número de operaciones que se comportan de manera incoherente B = Número total de operaciones que se comportan de forma normal Dónde: $B > 0$	$0 \leq X \leq 1$ El más cercano a 0 es el mejor	$X = \text{contable} / \text{contable}$ A = Contable B = Contable	Especificación de requerimientos, Código fuente, Desarrollador, Tester
	Posibilidad de personalización	Interna/Externa	¿Cuántas funciones y procedimientos operacionales puede un usuario modificar para su conveniencia?	Contar el número de funciones implementadas que pueden ser personalizadas durante la operación y el número de funciones que requieran la capacidad de personalización	$X = A / B$ A = Número de funciones implementadas que pueden ser personalizadas durante la operación B = Número de funciones que requieran la capacidad de personalización Dónde: $B > 0$	$0 \leq X \leq 1$ El más cercano a 1 es el mejor	$X = \text{contable} / \text{contable}$ A = Contable B = Contable	Especificación de requerimientos, Código fuente, Desarrollador, Tester

Protección contra errores del usuario	Verificación de entradas válidas.	Interna/Externa	¿Qué cantidad de ítems de entrada son validados?	Contar el número de ítems de entrada que son validados y el número de ítems que necesitan ser Validados.	$X = A / B$ A= Número de ítems de entrada que son validados B = Número de ítems que necesitan ser validados Dónde: $B > 0$	$0 \leq X \leq 1$ El más cercano a 1, es el mejor	$X = \text{contable} / \text{contable}$ A = Contable B = Contable	Especificación de requerimientos, Código fuente, Desarrollador, Tester
	Prevención del uso incorrecto	Interna/Externa	¿Cuántas funciones tienen la capacidad de evitar operaciones incorrectas?	Contar el número de Funciones implementadas para evitar fallos de funcionamiento provocados por un uso incorrecto y el número total de operaciones iniciales incorrectas	$X = A / B$ A = Número operaciones iniciales incorrectas B = Número de funciones implementadas para evitar fallos de funcionamiento provocados por un uso Incorrecto Dónde: $B > 0$	$0 \leq X \leq 1$ El más cercano a 1, el mejor	$X = \text{Contable} / \text{contable}$ A = Contable B = Contable	Especificación de requerimientos, Código fuente, Desarrollador, Tester

Estética de la Interfaz del usuario	Personalización de la apariencia de la interfaz del usuario	Interna/Externa	¿Qué cantidad de los elementos de la interfaz de usuario pueden ser personalizados en apariencia?	Contar el número de tipos de elementos de interfaz que pueden ser personalizados y contar el número total de tipos de elementos de interfaz	$X = A / B$ A = Número de elementos de interfaz que pueden ser personalizados B = Número total de elementos de interfaz Dónde: $B > 0$	$0 \leq X \leq 1$ El más cercano a 1, es el mejor	$X = \text{contable} / \text{contable}$ A = Contable B = Contable	Especificación de requerimientos, Código fuente, Desarrollador, Tester
Accesibilidad técnica	Accesibilidad física	Interna/Externa	¿A qué cantidad de funciones puede acceder un usuario con discapacidades físicas?	Contar el número de funciones a las que pueden acceder personas con discapacidad y contar el número total de funciones implementadas	$X = A / B$ A = Número de funciones a las que pueden acceder personas con discapacidad B = Número total de elementos de interfaz Dónde: $B > 0$	$0 \leq X \leq 1$ El más cercano a 1, es el mejor	$X = \text{contable} / \text{contable}$ A = Contable B = Contable	Especificación de requerimientos, Código fuente, Desarrollador, Tester

Tabla 27. Métricas para la característica de calidad Usabilidad

Fuente: Balseca (2014)

Métricas para la característica de calidad Mantenibilidad

Subcaracterística	Métrica	Fase del ciclo de vida de calidad del producto	Propósito de la métrica de calidad	Método de aplicación	Fórmula	Valor deseado	Tipo de medida	Recursos utilizados
Modularidad	Capacidad de condensación	Interna	¿Qué tan fuerte es la relación entre los componentes del sistema?	Contar el número de componentes que no son afectados por cambios de otros componentes y el número total de componentes específicos	$X = A / B$ A = Número de componentes que no son afectados por cambios de otros componentes B = Número total, de componentes específicos Dónde: $B > 0$	$0 \leq X \leq 1$ El más cercano a 0, es el mejor	X=contable/contable A = Contable B = Contable	Código fuente
	Acoplamiento de clases	Interna	¿Qué tan fuerte es la relación entre una función del sistema con otras clases implementadas?	Contar el número de relaciones que tiene una función con respecto a otras clases	$X = A$ A = Número de relaciones que tiene una función con respecto a otras clases	$1 \leq X \leq 4$ El más cercano a 1, es el mejor	X = Contable A = Contable	Código fuente

Reusabilidad	Ejecución de reusabilidad	Interna	¿Cuántos elementos pueden ser reutilizados?	Contar el número de elementos reutilizados y el número total de elementos de la biblioteca reutilizable	$X = A / B$ A = Número de elementos reutilizados B = Número total de elementos de la biblioteca reutilizable Dónde: $B > 0$	$0 \leq X \leq 1$ El más cercano a 1, es el mejor	$X = \text{contable} / \text{contable}$ A = Contable B = Contable	Código fuente
Capacidad de ser analizado	Capacidad de pistas de auditoría	Interna/Externa	¿Los usuarios pueden identificar fácilmente la operación específica que causó el fallo?	Contar el número de datos realmente grabados durante la operación y el número de datos previstos a grabarse para controlar el estado del sistema durante la operación	$X = A / B$ A = Número de datos realmente grabados durante la operación B = Número de datos previstos a grabarse para controlar el estado del sistema durante la operación Dónde: $B > 0$	$0 \leq X \leq 1$ El más cercano a 1, es el mejor	$X = \text{contable} / \text{contable}$ A = Contable B = Contable	Especificación de requerimientos, Código fuente, Desarrollador, Tester

	Diagnóstico de funciones suficientes	Interna/Externa	¿Hasta qué punto las funciones de diagnóstico están preparadas o hasta qué punto funcionan para el análisis causal?	Contar el número de funciones de diagnóstico implementadas y contar el número de funciones de diagnóstico requeridas en la especificación de requerimientos	$X = A / B$ A = Número de funciones de diagnóstico implementadas B = Número de funciones de diagnóstico requeridas en la especificación de requerimientos Dónde: $B > 0$	$0 \leq X \leq 1$ El más cercano a 1, es el mejor	$X = \text{contable} / \text{contable}$ A = Contable B = Contable	Especificación de requerimientos, Código fuente, Desarrollador, Tester
Capacidad de ser modificado	Complejidad ciclomática	Interna	¿Cuál es la complejidad estructural de un código fuente?	Contar las instrucciones condicionales, bucles, salidas de métodos y cláusulas AND y OR dentro de los condicionales.	$X = A + 1$ A = Número de instrucciones condicionales que tiene una función	$1 \leq X < 15$ El más cercano a 1, es el mejor	X = Contable A = Contable	Código fuente
	Profundidad de herencia	Interna	¿Qué tan profunda es la jerarquía de la herencia de las clases involucradas en una determinada función?	Contar las jerarquías empleadas en una determinada función o método.	$X = A$ A = Número de jerarquías empleadas para una determinada función.	$0 \leq X \leq 4$ El más cercano a 0 es el mejor	X = Contable A = Contable	Código fuente

	Grado de localización de corrección de impacto	Interna/Externa	¿Hasta qué punto los problemas causados pueden tener como consecuencia un mantenimiento?	Contar el número de fallas aparecidas después que se ha resuelto un fallo y contar el número de fallas resultas	$X = A / B$ A = Número de fallas aparecidas después que se ha resuelto un fallo B = Número de fallas resueltas Dónde: $B > 0$	$0 \leq X \leq 1$ El más cercano a 0, es el mejor	X = contable/contable A = Contable B = Contable	Especificación de requerimientos, Código fuente, Desarrollador, Tester
	Complejidad de modificación	Externa	¿Con qué facilidad el desarrollador puede modificar el software para resolver problemas?	Tomar el tiempo de trabajo que le toma al desarrollador modificar y contar el número de modificaciones	$X = A / T$ A = Número de modificaciones B = Tiempo de trabajo que le toma al desarrollador modificar Dónde: $T > 0$	$X = A / T$ El más lejano a 0/t es el mejor	X = Contable/ Tiempo A = Contable B = Tiempo	Desarrollador
	Índice de éxito de modificación	Externa	¿Hasta qué punto puede el sistema ser operado sin fallas después del mantenimiento?	Contar el número de problemas dentro de un determinado período antes de mantenimiento y contar el número de problemas en el	$X = A / B$ A = Número de problemas dentro de un determinado período antes de mantenimiento B = Número de problemas en el	$0 \leq X \leq 1$ El más cercano a 0, es el mejor	X = contable/contable A = Contable B = Contable	Desarrollador

				mismo período después del mantenimiento	mismo período después del mantenimiento o Dónde: $B > 0$			
Capacidad de ser probado	Complejidad funcional de funciones de pruebas	Interna	¿Son las funciones de prueba completas y fáciles de implementar?	Contar el número de funciones de prueba implementadas y contar el número de funciones de prueba requeridas	$X = A / B$ A = Número de funciones de prueba implementadas B = Número de funciones de prueba requeridas Dónde: $B > 0$	$0 \leq X \leq 1$ El más cercano a 1, es el mejor	X=contable/contable A = Contable B = Contable	Código fuente, Tester
	Capacidad de prueba autónoma	Interna	¿Qué tan independiente es el software al ser probado?	Contar el número de pruebas que están dependiendo de otros sistemas y contar el número total de pruebas dependientes con otros sistemas	$X = A / B$ A = Número de pruebas que están dependiendo de otros sistemas B = Número total de pruebas dependientes con otros sistemas Dónde: $B > 0$	$0 \leq X \leq 1$ El más cercano a 0, es el mejor	X=contable/contable A = Contable B = Contable	Código fuente, Tester

	Capacidad de reinicio de pruebas	Externa	¿Con qué facilidad se puede llevar a cabo las pruebas nuevamente después del mantenimiento?	Contar el número de casos en los cuales el mantenedor puede pausar y restaurar las pruebas y contar el número de casos de pausa en la ejecución de pruebas	$X = A / B$ A = Número de casos en los cuales el mantenedor puede pausar y restaurar las pruebas. B = Número de casos de pausa en la ejecución de pruebas Dónde: $B > 0$	$0 \leq X \leq 1$ El más cercano a 1, es el mejor	$X = \text{contable} / \text{contable}$ A = Contable B = Contable	Desarrollador, Tester
--	----------------------------------	---------	---	--	--	--	---	-----------------------

Tabla 28. Métricas para la característica de calidad Mantenibilidad

Fuente: Balseca (2014)

Anexo 2.2. Matriz para la evaluación de calidad Interna: Adecuación funcional, Fiabilidad, Eficiencia en desempeño, Usabilidad y Mantenibilidad.

CARACTERISTICAS	SUBCARACTERISTICAS	METRICA	FORMULA	VALOR DESEADO (UMBRA L, ETC)	APLI CA	VALOR OBTEN IDO (X)	PONDERA CION (/10)	VALO R PARC IAL TOTA L (/10)	NIVEL DE IMPORTA NCIA	PORCENT AJE DE IMPORTA NCIA	VAL OR FINA L	CALID AD DEL SISTE MA (/10)
Adecuación funcional	Compleitud funcional	Compleitud de la implementación funcional	$X = A / B$ A = Número de funciones que están incorrectas o que no fueron implementadas. B = Número de las funciones establecidas en la especificación de requisitos. Donde: $B > 0$	0	SI	A = 0 B = 25 X = 0,00	10,00	10	M	15%	1,50	7,55
	Exactitud funcional	Exactitud	$X = A / B$ A = Número de elementos de datos implementados con el estándar específico de exactitud. B = Número de las funciones establecidas en la especificación de requisitos. Donde: $B > 0$	1	No	A = B = X = N/A	N/A					

		Precisión Computacional	$X = A / T$ A = Número de cálculos inexactos encontrados. T = Tiempo de operación Donde: $T > 0$	Deseado: 0/15 Min. Peor caso: $\geq 10/15$ Min.	No	A = T = X = N/A	N/A				
Fiabilidad	Madurez	Eliminación de errores	$X = A / B$ A = Número de fallas corregidas, en la fase de diseño, codificación y pruebas. B = Número de fallas detectadas en las pruebas. Donde: $B > 0$	0	No	A = B = X = N/A	N/A				
		Cobertura de pruebas	$X = A / B$ A = Número de casos de pruebas realizados en un escenario de operación durante la prueba B = Número de casos de prueba a ser realizados para cubrir los requerimientos Donde: $B > 0$	1	No	A = B = X = N/A	N/A	10	M	10%	1,00

		Redundancia	$X = A / B$ A = Número de componentes/sistemas instalados de forma redundante B = Número total de componentes/sistemas instalados Donde: $B > 0$	1	No	A = B = X = N/A	N/A						
	Tolerancia a fallos	Anulación de operación incorrecta	$X = A / B$ A = Número de operaciones incorrectas presentadas B = Número total de funciones implementadas para anular operaciones incorrectas. Donde: $B > 0$	0	Si	A = 0 B = 1 X = 0,00	10,00						
	Capacidad de recuperación	Tiempo medio de recuperación	$X = A / T$ A = Número de casos en los cuales se ha observado que el sistema entro en recuperación. T = Tiempo que tomo el sistema en recuperarse. Donde: $T > 0$	Deseado: 0/Min. Peor caso: ≥ 10 /Min.	No	A = B = X = N/A	N/A						
	Comportamiento del tiempo	Tiempo de respuesta	$X = B - A$ A = Tiempo de envío de la petición B = Tiempo en recibir la primera	Deseado: 15 seg. Peor caso: > 15 seg.	No	A = B = X = N/A	N/A						

Eficiencia en el desempeño			respuesta Donde: $B > 0$																			
		Tiempo de espera	$X = B - A$ A = Tiempo cuando se inicia un trabajo B = Tiempo en completar el trabajo Donde: $B > 0$	Deseado: 20 min. Peor caso: > 20 min.	No	A = B = X = N/A	N/A															
		Rendimiento	$X = A / T$ A = Número de tareas completadas T = Intervalo de tiempo Donde: $T > 0$	Deseado: $\geq 10/20$ min. Peor caso: 0/20 min.	No	A = T = X = N/A	N/A															
	Utilización de recursos	Líneas de código	$X = A$ A = Número de líneas de código	Deseado: 1 Peor caso: ≥ 50	Si	A = 22 X = 22	5,60															
		Utilización de CPU	$X = A$ A = Cantidad de CPU que es usado para realizar una tarea	Deseado: 0% Peor caso: $\geq 10\%$	No	A = X = N/A	N/A	5,60	M	20%	1,12											
		Utilización de memoria	$X = A$ A = Cantidad de memoria que es usado para realizar una tarea	Deseado: 0% Peor caso: $\geq 10\%$	No	A = X = N/A	N/A															
		Utilización de los dispositivos de E/S	$X = B - A$ A = Tiempo que los dispositivos de E/S pasan ocupados para realizar una tarea B = Tiempo de	Deseado: 0 seg. Peor caso: > 15 seg.	No	A = B = X = N/A	N/A															

			Operación Dónde: $B > 0$									
	Capacidad	Número de peticiones online (Max)	$X = A / T$ A= Número máximo de peticiones online procesada T = Tiempo de operación Dónde: $T > 0$	Deseado: $\geq 10/3$ Min. Peor caso: 0/3 Min.	No	A = T = X = N/A	N/A					
		Número de accesos simultáneos (Max)	$X = A / T$ A= Número máximo de accesos simultáneos T = Tiempo de operación Dónde: $T > 0$	Deseado: $\geq 10/3$ Min. Peor caso: 0/3 Min.	No	A = T = X = N/A	N/A					
Usabilidad	Capacidad de reconocer su adecuación	Integración descripción	$X = A / B$ A = Número de funciones (o tipos de funciones) descritas como entendibles en la descripción del producto. B = Número total de funciones (o tipos de funciones) Dónde: $B > 0$	1	Si	A = 25 B = 25 X = 1,00	10,00	7,53	M	20%	1,51	

		Capacidad de demostración	$X = A / B$ A = Número de funciones implementadas con capacidad de demostración B = Número total de funciones que requieren capacidad de demostración Dónde: $B > 0$	1	No	A = B = X = N/A	N/A						
	Capacidad de ser atendido	Funciones evidentes	$X = A / B$ A= Número de funciones (o tipo de funciones) evidentes al usuario B = Número total de funciones (o tipo de funciones) Dónde: $B > 0$	1	No	A = 25 B = 25 X = 1,00	10,00						
		Efectividad de la documentación del usuario o ayuda del sistema	$X = A / B$ A = Número de funciones descritas correctamente B = Número total de funciones implementadas Dónde: $B > 0$	1	No	A = B = X = N/A	N/A						

		Recuperabilidad de error operacional	$X = A / B$ A= Número de funciones implementadas con tolerancia de error de usuarios B = Número total de funciones requeridas con capacidad de tolerancia. Dónde: $B > 0$	1	No	A = B = X = N/A	N/A						
	Operatividad	Claridad de mensajes	$X = A / B$ A= Número de mensajes implementados con explicaciones claras B = Número total de mensajes implementados Dónde: $B > 0$	1	No	A = 90 B = 90 X = 0.94	9.38						
		Consistencia operacional	$X = A / B$ A= Número de operaciones que se comportan de manera incoherente. B = Número total de operaciones que se comportan de forma normal. Dónde: $B > 0$	0	No	A = B = X = N/A	N/A						

		Posibilidad de personalización	$X = A / B$ A = Número de funciones implementadas que pueden ser personalizados durante la operación B = Número de funciones que requieran la capacidad de personalización Dónde: $B > 0$	0	No	A = B = X = N/A	N/A						
	Protección contra errores de usuario	Prevención del uso incorrecto	$X = A / B$ A = Número operaciones iniciales incorrectas B = Número de funciones implementadas para evitar fallos de funcionamiento provocados por un uso Incorrecto Dónde: $B > 0$	1	Si	A = 8 B = 109 X = 0.07	0,73						
	Estética de la interfaz del usuario	Personalización de la apariencia de la interfaz del usuario	$X = A / B$ A = Número de elementos de interfaz que pueden ser personalizados B = Número total de elementos de interfaz Dónde: $B > 0$	1	No	A = B = X = N/A	N/A						

	Accesibilidad técnica	Accesibilidad física	X = A / B A= Número de funciones a las que pueden acceder personas con discapacidad B = Número total de elementos de interfaz Dónde: B > 0	1	No	A = B = X = N/A	N/A					
	Modularidad	Capacidad de condensación	X = A / B A = Número de componentes que no son afectados por cambios de otros componentes B = Número total, de componentes específicos Dónde: B > 0	0	Si	A = 0 B = 4 X = 0,00	10,00	6,92	A	35%	2,42	
		Acoplamiento de clases	X = A / B A = Número de relaciones que tiene una función con respecto a otras clases	Deseado: 1 Peor caso: >= 4	Si	A = 2 X = 2	5,00					
	Reusabilidad	Ejecución de la reusabilidad	X = A / B A = Número de elementos reutilizados B = Número total de elementos de la biblioteca reutilizable Dónde: B > 0	1	SI	A = 7 B = 7 X = 1,00	10,00					

Mantenibilidad	Capacidad de ser analizado	Capacidad de pistas de auditoria	$X = A / B$ A = Número de datos realmente grabadas durante la operación B = Número de datos previstos a grabarse para controlar el estado del sistema durante la operación Dónde: $B > 0$	1	Si	A = 20 B = 20 X = 1,00	10,00						
		Diagnóstico de funciones suficientes	$X = A / B$ A = Número de funciones de diagnóstico implementadas B = Número de funciones de diagnóstico requeridas en la especificación de requerimientos Dónde: $B > 0$	1	No	A = B = X = N/A	N/A						
	Capacidad de ser modificado	Complejidad ciclomática	$X = A + 1$ A = Número de instrucciones condicionales que tiene una función	Deseado: 1 Peor caso: ≥ 15	Si	A = 8 X = 8	4,00						
		Profundidad de herencia	$X = A$ A = Número de jerarquías empleadas para una determinada función.	Deseado: 1 Peor caso: ≥ 4	Si	A = 3 X = 3	2,50						

		Grado de localización de corrección de impacto	$X = A / B$ A = Número de fallas aparecidas después que se ha resuelto un fallo B = Número de fallas resueltas Dónde: $B > 0$	0	No	A = B = X = N/A	N/A						
Capacidad de ser probado		Compleitud funcional de funciones de pruebas	$X = A / B$ A = Número de funciones de prueba implementadas B = Número de funciones de prueba requeridas Dónde: $B > 0$	1	No	A = B = X = N/A	N/A						
		Capacidad de prueba autónoma	$X = A / B$ A = Número de pruebas que están dependiendo de otros sistemas B = Número total de pruebas dependientes con otros sistemas Dónde: $B > 0$	0	No	A = B = X = N/A	N/A						

Tabla 29. Matriz para evaluación de calidad Interna.

Fuente: Balseca (2014)

Anexo 2.3. Matriz para la evaluación de calidad Externa: Adecuación funcional, Fiabilidad, Eficiencia en desempeño, Usabilidad y Mantenibilidad.

CARACTERISTICAS	SUBCARACTERISTICAS	METRICA	FORMULA	VALOR DESEADO (UMBRAL, ETC)	APLICACION	VALOR OBTENIDO (X)	PONDERACION (/10)	VALOR PARCIAL TOTAL (/10)	NIVEL DE IMPORTANCIA	PORCENTAJE DE IMPORTANCIA	VALOR FINAL	CALIDAD DEL SISTEMA (/10)
Adecuación funcional	Compleitud funcional	Compleitud de la implementación funcional	$X = A / B$ A = Número de funciones que están incorrectas o que no fueron implementadas. B = Número de las funciones establecidas en la especificación de requisitos. Donde: $B > 0$	0	Si	A = 0 B = 25 X = 0,00	10,00	10,00	A	30%	3,00	9,08
	Exactitud funcional	Exactitud	$X = A / B$ A = Número de elementos de datos implementados con el estándar específico de exactitud. B = Número de las funciones establecidas en la especificación de requisitos. Donde: $B > 0$	1	No	A = B = X = N/A	N/A					

		Precisión Computacional	$X = A / T$ A = Número de cálculos inexactos encontrados. T = Tiempo de operación Donde: $T > 0$	Deseado: 0/15 Min. Peor caso: $\geq 10/15$ Min.	Si	A = 0 T = 15 X = 0/15 min	10,00					
Fiabilidad	Madurez	Eliminación de errores	$X = A / B$ A = Número de fallas corregidas, en la fase de diseño, codificación y pruebas. B = Número de fallas detectadas en las pruebas. Donde: $B > 0$	0	Si	A = 2 B = 4 X = 0,50	5,00	8,75	M	15%	1,31	
		Cobertura de pruebas	$X = A / B$ A = Número de casos de pruebas realizados en un escenario de operación durante la prueba B = Número de casos de prueba a ser realizados para cubrir los requerimientos Donde: $B > 0$	1	Si	A = 5 B = 5 X = 1,00	10,00					
		Tiempo medio entre fallos	$X = A/T$ A = Número total de fallas detectadas actualmente T = Tiempo de operación Donde $T > 0$	Deseado: 0/15 Min. Peor caso: $\geq 10/15$ Min.	Si	A = 0 T = 15 X = 0	10,00					

Disponibilidad	Tiempo de servicio	$X = A / B$ A = Tiempo de servicio del sistema que se proporciona actualmente B = Tiempo de servicio del sistema regulado en el cronograma operacional Dónde: $B > 0$	1	SI	A = 365 B = 365 X = 1,00	10,00				
	Tiempo medio de inactividad	$X = A / T$ A = Número de fallos observados T = Tiempo total de inactividad Dónde: $T > 0$	Deseado: 0/Min. Peor caso: $\geq 10/\text{Min.}$	No	A = T = X = N/A	N/A				
Tolerancia a fallos	Redundancia	$X = A / B$ A = Número de componentes/ sistemas instalados de forma redundante B = Número total de componentes/ sistemas instalados Dónde: $B > 0$	1	No	A = B = X = N/A	N/A				
Capacidad de recuperación	Tiempo medio de recuperación	$X = A / T$ A = Número de casos en los cuales se ha observado que el sistema entro en recuperación. T = Tiempo que tomo el sistema en	Deseado: 0/Min. Peor caso: $\geq 10/\text{Min.}$	Seleccionar	A = B = X = N/A	N/A				

			recuperarse. Donde: $T > 0$										
Eficiencia en el desempeño	Comportamiento del tiempo	Tiempo de respuesta	$X = B - A$ A = Tiempo de envío de la petición B = Tiempo en recibir la primera respuesta Donde: $B > 0$	Deseado: 15 seg. Peor caso: > 15 seg.	Si	A = 0 B = 14 X = 0	10,00	9,00	M	20%	1,8		
		Tiempo de espera	$X = B - A$ A = Tiempo cuando se inicia un trabajo B = Tiempo en completar el trabajo Donde: $B > 0$	Deseado: 20 min. Peor caso: > 20 min.	Si	A = 0 B = 20 X = 0	10,00						
		Rendimiento	$X = A / T$ A = Número de tareas completadas T = Intervalo de tiempo Donde: $T > 0$	Deseado: $\geq 10/20$ min. Peor caso: $0/20$ min.	Si	A = 21 T = 20 X = $21/20$ Min.	10,00						
	Utilización de recursos	Líneas de código	$X = A$ A = Número de líneas de código	Deseado: 1 Peor caso: ≥ 50	No	A = X = N/A	N/A						
		Utilización de CPU	$X = A$ A = Cantidad de CPU que es usado para realizar una tarea	Deseado: 0% Peor caso: $\geq 10\%$	Si	A = 3 X = 3,00	7,00						
		Utilización de memoria	$X = A$ A = Cantidad de memoria que es usado	Deseado: 0% Peor caso: $\geq 10\%$	Si	A = 2 X = 2,00	8,00						

		para realizar una tarea																	
		Utilización de los dispositivos de E/S	$X = B - A$ $A =$ Tiempo que los dispositivos de E/S pasan ocupados para realizar una tarea $B =$ Tiempo de Operación Dónde: $B > 0$	Deseado: 0 seg. Peor caso: > 15 seg.	No	$A =$ $B =$ $X =$ N/A	N/A												
Capacidad		Número de peticiones online (Max)	$X = A / T$ $A =$ Número máximo de peticiones online procesada $T =$ Tiempo de operación Dónde: $T > 0$	Deseado: $\geq 10/3\text{Min}$ Peor caso: 0/3Min.	No	$A =$ $T =$ $X =$ N/A	N/A												
		Número de accesos simultáneos (Max)	$X = A / T$ $A =$ Número máximo de accesos simultáneos $T =$ Tiempo de operación Dónde: $T > 0$	Deseado: $\geq 10/3\text{Min}$ Peor caso: 0/3Min.	No	$A =$ $T =$ $X =$ N/A	N/A												
		Sistema de transmisión de ancho de banda	$X = A / T$ $A =$ Cantidad máxima de transmisión de datos $B =$ Tiempo de operación Dónde: $T > 0$	Deseado: $\geq 10/\text{Min}$. Peor caso: 0/Min.	No	$A =$ $T =$ $X =$ N/A	N/A												

Usabilidad	Capacidad de reconocer su adecuación	Integración descripción	$X = A / B$ A = Número de funciones (o tipos de funciones) descritas como entendibles en la descripción del producto. B = Número total de funciones (o tipos de funciones) Dónde: $B > 0$	1	No	A = B = X = N/A	N/A	6.46	M	15%	0,97
		Capacidad de demostración	$X = A / B$ A = Número de funciones implementadas con capacidad de demostración B = Número total, de funciones que requieren capacidad de demostración Dónde: $B > 0$	1	No	A = B = X = N/A	N/A				
	Capacidad de ser atendido	Efectividad de la documentación del usuario o ayuda del sistema	$X = A / B$ A= Número de funciones descritas correctamente B = Número total de funciones implementadas Dónde: $B > 0$	1	Si	A = 25 B = 25 X = 1,00	10,00				

Operatividad	Claridad de mensajes	$X = A / B$ A= Número de mensajes implementados con explicaciones claras B = Número total de mensajes implementados Dónde: $B > 0$	1	Si	A = 90 B = 96 X = 0,94	9,38				
	Consistencia operacional	$X = A / B$ A= Número de operaciones que se comportan de manera incoherente. B = Número total de operaciones que se comportan de forma normal. Dónde: $B > 0$	0	No	A = B = X = N/A	N/A				
	Posibilidad de personalización	$X = A / B$ A = Número de funciones implementadas que pueden ser personalizadas durante la operación B = Número de funciones que requieran la capacidad de personalización Dónde: $B > 0$	0	No	A = B = X = N/A	N/A				

Protección contra errores de usuario	Verificación de entradas validas	X = A / B A= Número de ítems de entrada que son validados B = Número de ítems que necesitan ser validados Dónde: B > 0	1	No	A = B = X = N/A	N/A				
	Prevención del uso incorrecto	X = A / B A = Número operaciones iniciales incorrectas B = Número de funciones implementadas para evitar fallos de funcionamiento provocados por un uso incorrecto Dónde: B > 0	1	No	A = B = X = N/A	N/A				
	Estética de la interfaz del usuario	Personalización de la apariencia de la interfaz del usuario	X = A / B A = Número de elementos de interfaz que pueden ser personalizados B = Número total de elementos de interfaz Dónde: B > 0	1	Si	A = 0 B = 76 X = 0,00	0,00			

	Accesibilidad técnica	Accesibilidad física	X = A / B A= Número de funciones a las que pueden acceder personas con discapacidad B = Número total de elementos de interfaz Dónde: B > 0	1	No	A = B = X = N/A	N/A					
Mantenibilidad	Capacidad de ser analizado	Capacidad de pistas de auditoria	X = A / B A = Número de datos realmente grabadas durante la operación B = Número de datos previstos a grabarse para controlar el estado del sistema durante la operación Dónde: B > 0	1	Si	A = 21 B = 21 X = 1,00	10,00	10.00	M	20%	2,00	
		Diagnóstico de funciones suficientes	X = A / B A = Número de funciones de diagnóstico implementadas B = Número de funciones de diagnóstico requeridas en la especificación de requerimientos Dónde: B > 0	1	No	A = B = X = N/A	N/A					

Capacidad de ser modificado	Grado de localización de corrección de impacto	X = A / B A = Número de fallas aparecidas después que se ha resuelto un fallo B = Número de fallas resueltas Dónde: B > 0	0	No	A = B = X = N/A	N/A				
	Complejidad de modificación	X = A / T A = Número de modificaciones B = Tiempo de trabajo que le toma al desarrollador modificar Dónde: T > 0	Deseado: >= 1/60Min Peor caso: 0/60Min	Si	A = 3 T = 60 X = 3/60 min	10,00				
	índice de éxito de modificación	X = A / B A = Número de problemas dentro de un determinado período antes de mantenimiento B = Número de problemas en el mismo período después del mantenimiento Dónde: B > 0	0	No	A = B = X = N/A	N/A				
Capacidad de ser probado	Capacidad de reinicio de pruebas	X = A / B A = Número de casos en los cuales el mantenedor puede pausar y restaurar las pruebas. B = Número de casos de pausa en la ejecución de	1	No	A = B = X = N/A	N/A				

