



**UNIVERSIDAD POLITÉCNICA SALESIANA**  
**SEDE CUENCA**  
**CARRERA DE COMPUTACIÓN**

**DESARROLLO DE UN MÓDULO PARA LA GESTIÓN DE INSUMOS E  
INVENTARIO EN LA VETERINARIA POLIVET FASE II DE LA APLICACIÓN  
MULTIPLATAFORMA UTILIZANDO IONIC Y JEE**

Trabajo de titulación previo a la obtención del  
título de Ingeniero en Ciencias de la Computación

**AUTORES: KEVIN ADRIÁN GODOY MENDÍA**

**WILLIAM EDUARDO CHABLA PALAGUACHI**

**TUTOR: ING. GABRIEL ALEJANDRO LEÓN PAREDES, PHD.**

Cuenca - Ecuador

2022

## **CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN**

Nosotros, Kevin Adrián Godoy Mendía con documento de identidad N° 0105263123 y William Eduardo Chabla Palaguachi con documento de identidad N° 0302306915; manifestamos que:

Somos los autores y responsables del presente trabajo; y, autorizamos a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

Cuenca, 26 de julio del 2022

Atentamente,

---

Kevin Adrián Godoy Mendía

0105263123

---

William Eduardo Chabla Palaguachi

0302306915

## **CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA**

Nosotros, Kevin Adrián Godoy Mendía con documento de identificación N° 0105263123 y William Eduardo Chabla Palaguachi con documento de identificación N° 0302306915, expresamos nuestra voluntad y por medio del presente documento cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud que somos autores del Proyecto Técnico: “Desarrollo de un módulo para la gestión de insumos e inventario en la veterinaria Polivet fase II de la aplicación multiplataforma utilizando Ionic y JEE”, el cual ha sido desarrollado para optar por el título de: Ingeniero en Ciencias de la Computación, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribimos este documento en el momento que hacemos la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Cuenca, 26 de julio del 2022

Atentamente,

---

Kevin Adrián Godoy Mendía

0105263123

---

William Eduardo Chabla Palaguachi

0302306915

## **CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN**

Yo, Gabriel Alejandro León Paredes con documento de identificación N° 0103652186, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: DESARROLLO DE UN MODULO PARA LA GESTIÓN DE INSUMOS E INVENTARIO EN LA VETERINARIA POLIVET FASE II DE LA APLICACIÓN MULTIPLATAFORMA UTILIZANDO IONIC Y JEE, realizado por Kevin Adrián Godoy Mendía con documento de identificación N° 0105263123 y por William Eduardo Chabla Palaguachi con documento de identificación N° 0302306915, obteniendo como resultado final el trabajo de titulación bajo la opción Proyecto Técnico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Cuenca, 26 de julio del 2022

Atentamente,

---

Ing. Gabriel Alejandro León Paredes, PhD

0103652186

## **DEDICATORIA Y AGRADECIMIENTO**

En primer lugar, quiero agradecer a Dios, es el quién guía mis pasos y mis decisiones en todo momento, además de ser quien que me ha permitido llegar a este lugar. Esto se lo quiero dedicar de manera especial a mis padres. Ellos han sido un pilar fundamental durante todos estos años de estudio, gracias por haber confiado en mí, invertido en mí, por no haberse descuidado de mí, por haberme inspirado con su arduo sacrificio y por haberme ayudado a salir adelante. Gracias por inculcarme que todo esfuerzo tiene altos y bajos, pero que lo más importante es la perseverancia, la cual me ha permitido llegar hasta aquí y me permitirá seguir progresando a futuro. Agradezco a mi familia, a mis primos, y de manera especial quiero agradecer a mi hermano Juan Pablo y a mis dos hermanas María Gracia y Daniela, quienes al igual que mis padres han sido mi motivación durante todo este proceso.

Hacer mención a mi amigo y compañero de tesis por haber formado parte de este camino y habernos ayudado mutuamente, por su dedicación y contribución realizada durante todo este proceso. De aquí en adelante espero que su vida este llena de éxitos profesionales y personales.

Finalmente, quiero dar las gracias a todos los profesores que he tenido a lo largo de los ciclos, ya que han sido los responsables de otorgarme las bases durante todos estos años de estudio en las diferentes áreas. Agradezco que me hayan ayudado con las inquietudes tanto en trabajos como proyectos, porque sus consejos han solidificado esas bases, las cuales espero seguirlas perfeccionando a futuro.

*Kevin Adrián Godoy Mendía*

Estoy muy agradecido principalmente a Dios por que gracias a él que me dio vida y salud que es lo primordial para cumplir esta meta muy importante en mi vida. Agradezco a mis padres y mi hermana ya que ellos han sido parte importante en este proceso, ya que su apoyo principalmente emocional me ha ayudado a no rendirme y seguir adelante en los momentos más complicados. También estoy muy agradecido con mis abuelitos y mis tíos que su apoyo a significado mucho para mí y estuvieron minuto a minuto pendiente de mi salud y educación; que Dios les siga dando mucha salud y muchas bendiciones.

Doy las gracias a mi compañero de tesis ya que juntos luchamos por este proceso grande para nuestras vidas y le deseo lo mejor en su vida laboral y que siga cosechando muchos éxitos en un futuro no muy lejano.

Finalmente agradezco a mis profesores que pusieron en mi sus conocimientos y experiencias para que sepa enfrentarme en campo laboral y que siga adelante a superarme y algún día tener los logros que ellos han obtenido con mucho esfuerzo y sacrificio.

*William Eduardo Chabla Palaguachi*

## RESUMEN

Gran parte de las empresas hoy en día eligen implementar varias tecnologías. Su propósito es sistematizar y mejorar sus operaciones diarias con el fin de reducir la pérdida innecesaria de tiempo, recursos y costos. Parte de estas empresas son las clínicas veterinarias en donde se procesa una gran cantidad de información; Los documentos físicos se pueden perder y destruir, razón por la cual el software empresarial es tan esencial.

En las clínicas veterinarias los procesos (gestión de servicios, facturación, etc.) deben estar alineados con las tendencias tecnológicas actuales y la globalización del mercado con el uso de aplicaciones web progresivas y aplicaciones multiplataforma, pues cada día se hace más urgente revolucionar la forma en las que las veterinarias gestionan estos procesos.

En consecuencia, la propuesta de este trabajo es el desarrollo de módulos para la gestión de insumos e inventario mediante el uso del marco de desarrollo en el framework ionic a fin de optimizar el tiempo de los veterinarios de la clínica “Polivet” en la gestión de insumos e inventario.

La aplicación multiplataforma permitirá a los veterinarios administrar y controlar los suministros con los que cuenta la clínica, el registro de nuevos insumos, la gestión de sus laboratorios, los insumos utilizados en el día a día, además de la búsqueda de insumos mediante la aplicación móvil y el sitio web. Todo ellos se realizan para tener un mejor control en la gestión de los insumos médicos con la que se maneja constantemente la veterinaria y lograr que la creación de este modulo sea de gran ayuda para los médicos que utilicen la aplicación multiplataforma.

**Palabras claves:** desarrollo web, desarrollo móvil, multiplataforma, clínica veterinaria, insumos, inventario, libro diario, ajuste.

## **ABSTRACT**

Most companies today choose to implement multiple technologies. Its purpose is to systematize and improve your daily operations to reduce unnecessary waste of time, resources, and costs. Some of these companies are veterinary clinics where a large amount of information is processed; Physical documents can be lost and destroyed, which is why business software is so essential.

In veterinary clinics, processes (service management, billing, etc.) must be aligned with current technological trends and market globalization with the use of progressive web applications and multiplatform applications, since every day it becomes more urgent to revolutionize the way in which veterinarians manage these processes.

Consequently, the proposal of this work is the development of modules for the management of supplies and inventory using the development framework in the ionic framework to optimize the time of the veterinarians of the "Polivet" clinic in the management of supplies and inventory.

The multiplatform application will allow veterinarians to manage and control the supplies that the clinic has, the registration of new supplies, the management of their laboratories, the supplies used on a day-to-day basis, as well as the search for supplies through the mobile application. and the website. All of them are carried out to have a better control in the management of medical supplies with which the veterinary is constantly handled and to make the creation of this module a great help for doctors who use the multiplatform application.

**Key words:** web development, mobile, development, multiplatform, veterinary clinic, supplies, inventory, diary book, adjustment.



# Índice de Contenido

RESUMEN .....	7
ABSTRACT .....	8
1. INTRODUCCIÓN.....	16
2. DESCRIPCIÓN DEL PROBLEMA .....	17
2.1.1. ANTECEDENTES .....	17
2.1.2 IMPORTANCIA Y ALCANCES.....	18
3. OBJETIVOS GENERALES Y ESPECÍFICOS .....	19
3.1 OBJETIVOS GENERALES.....	19
3.2 OBJETIVOS ESPECÍFICOS.....	19
4. FUNDAMENTOS TEÓRICOS .....	20
4.1 ORÍGENES DE LA MEDICINA VETERINARIA .....	20
4.2 GESTIÓN DE INSUMOS DE MASCOTAS .....	21
4.3 GESTIÓN DE INVENTARIO .....	21
4.4 GESTIÓN LABORATORIOS .....	21
4.5 GESTIÓN LIBRO DIARO .....	22
4.6 INGENIERÍA DE SOFTWARE .....	22
4.7 INGENIERÍA DE REQUERIMIENTOS .....	23
4.7.1 <i>Obtención de información</i> .....	23
4.7.2 <i>Especificación de requerimientos</i> .....	24
4.8 APLICACIONES MULTIPLATAFORMA.....	25
4.9 HERRAMIENTAS DE DESARROLLO FRONTEND .....	25
4.9.1 <i>Framework Ionic</i> .....	25
4.9.2 <i>Typescript</i> .....	26
4.10 HERRAMIENTAS DE DESARROLLO BACKEND .....	26
4.10.1 <i>Java</i> .....	26
4.10.2 <i>Seguridad con Angular &amp; JEE (Oracle Platform, Enterprise Edition)</i> .....	27
4.10.3 <i>Web services</i> .....	27
4.11 PWA (APLICACIÓN WEB PROGRESIVA) .....	28
4.12 APLICACIONES MÓVILES HÍBRIDAS .....	28
4.12.1 <i>Ventajas:</i> .....	29

4.12.2 Desventajas:	29
4.13 BASES DE DATOS RELACIONALES	29
4.14 BASE DE DATOS MYSQL	30
4.15 METODOLOGÍA DE DESARROLLO SCRUM	30
5. MARCO METODOLÓGICO	32
5.1 REQUERIMIENTOS FUNCIONALES	32
5.2 REQUERIMIENTOS NO FUNCIONALES	33
5.3 DISEÑO Y ARQUITECTURA	36
5.3.1. Diagrama de procesos	36
5.3.2. Especificaciones de los casos de uso	39
5.3.3. Descripción de los casos de uso	40
5.4 DISEÑO DE LA APLICACIÓN MULTIPLATAFORMA	56
5.4.1. Diagrama entidad relación	57
5.4.2. Interfaces de usuario	59
5.4.3. Arquitectura de la aplicación	70
5.5 METODOLOGÍA DE DESARROLLO	73
5.1.1. Descripción personal scrum	73
5.1.2. Conformación de los Sprints	73
6. RESULTADOS	78
6.1 CODIFICACIÓN WEB SERVICES	78
6.2 CÓDIGO WEB SERVICES	83
6.3. INTERFACES DE USUARIO FINALES	87
6.4 DESPLIEGUE DE LA APLICACIÓN	96
6.5 ENCUESTA	97
6.5.1 Resultado Pregunta 1:	97
6.5.2 Resultado Pregunta 2:	97
6.5.3 Resultado Pregunta 3:	98
6.5.4 Resultado Pregunta 4:	98
6.5.5 Resultado Pregunta 5:	98
6.5.6 Resultado Pregunta 6:	98
6.5.7 Resultado Pregunta 7:	98
7. CRONOGRAMA DE ACTIVIDADES	99

8. PRESUPUESTO.....	104
9. CONCLUSIONES.....	105
10. RECOMENDACIONES .....	107
REFERENCIAS BIBLIOGRÁFICAS .....	108
ANEXOS.....	113
<i>Anexo 1. Prospección de la Encuesta.....</i>	<i>113</i>
<i>Anexo 2. Manual de usuario.....</i>	<i>115</i>
<i>Anexo 2.1 Página de inicio de sesión.....</i>	<i>115</i>
<i>Anexo 2.2 Página error 404.....</i>	<i>115</i>
<i>Anexo 2.3 Página listado de Inventario .....</i>	<i>116</i>
<i>Anexo 2.4 Generación de reportes .....</i>	<i>116</i>
<i>Anexo 2.5 Carga CSV.....</i>	<i>117</i>
<i>Anexo 2.6 Listado Libro Diario.....</i>	<i>119</i>
<i>Anexo 2.7 Listado Insumos Libro Diario .....</i>	<i>119</i>
<i>Anexo 2.8 Listado Historial insumo .....</i>	<i>120</i>
<i>Anexo 2.9 Registro Libro Diario .....</i>	<i>120</i>
<i>Anexo 2.10 Modificar Libro Diario.....</i>	<i>121</i>
<i>Anexo 2.11 Listado Laboratorio.....</i>	<i>121</i>
<i>Anexo 2.12 Registro Laboratorio .....</i>	<i>122</i>
<i>Anexo 2.13 Modificar Laboratorio.....</i>	<i>122</i>
<i>Anexo 2.14 Eliminar Laboratorio .....</i>	<i>122</i>
<i>Anexo 2.15 Listado insumo.....</i>	<i>123</i>
<i>Anexo 2.18 Eliminar insumo .....</i>	<i>124</i>
<i>Anexo 2.19 Registrar Ajuste .....</i>	<i>125</i>
<i>Anexo 2.20 Historial de Ajuste .....</i>	<i>125</i>

## Índice de Tablas

Tabla 1.Requerimientos no funcionales: Interfaz del sistema.....	33
Tabla 2. Requerimientos no funcionales: Simplifica en el uso del sistema .....	33

Tabla 3.Requerimientos no funcionales: Diseño de la interfaz a características multiplataformas.....	34
Tabla 4.Requerimientos no funcionales: Desempeño .....	34
Tabla 5.Requerimientos no funcionales: Seguridad en información.....	35
Tabla 6.Revisión diagrama de procesos .....	38
Tabla 7.Requerimiento Funcional: Registro Insumos .....	40
Tabla 8.Requerimiento Funcional: Registro Libro Diario.....	41
Tabla 9.Requerimiento Funcional: Registro Laboratorio.....	42
Tabla 10.Requerimiento Funcional: Registro Ajuste .....	43
Tabla 11.Requerimiento Funcional: Buscar y Visualizar el insumo .....	44
Tabla 12.Requerimiento Funcional: Búsqueda y Visualización inventario .....	45
Tabla 13.Requerimiento Funcional: Búsqueda y Visualizar Libro diario.....	46
Tabla 14.Requerimiento Funcional: Búsqueda y Visualizar Ajuste de inventario.....	47
Tabla 15.Requerimiento Funcional: Visualizar Inventario por fecha de caducidad o stock.....	48
Tabla 16. Requerimiento Funcional: Buscar y Visualizar laboratorio .....	49
Tabla 17.Requerimiento Funcional: Historial del insumo en el libro diario .....	50
Tabla 18.Requerimiento Funcional: Modificar insumo .....	51
Tabla 19. Requerimiento Funcional: Modificar laboratorio.....	52
Tabla 20. Requerimiento Funcional: Eliminar insumo .....	53
Tabla 21.Requerimiento Funcional: Eliminar laboratorios .....	54
Tabla 22. Requerimiento Funcional: Generar Reportes .....	55
Tabla 23. Requerimiento Funcional: Actualización del Inicio de Sesión .....	56
Tabla 24. Revisión de prototipos.....	70
Tabla 25. Asignación de roles del método SCRUM .....	73
Tabla 26. Actividades del Objetivo Específico 1 .....	74
Tabla 27. Actividades del Objetivo Específico 2 .....	74
Tabla 28. Actividades del Objetivo Específico 3 .....	75
Tabla 29. Actividades del Objetivo Específico 4 .....	75
Tabla 30. Actividades del Objetivo Específico 5 .....	76
Tabla 31. Distribución de Sprint 1 .....	76
Tabla 32. Distribución de Sprint 2 .....	77
Tabla 33. Distribución de Sprint 3 .....	78
Tabla 34. Presentación de los respectivos sprint desarrollados.....	79

Tabla 42. Total, horas de los desarrolladores .....	99
Tabla 43. Cronograma de actividades de cada objetivo específico .....	100
Tabla 44. Presupuesto Fijo .....	104
Tabla 35. Interrogante 1. Factibilidad al momento de usar .....	113
Tabla 36. Interrogante 2. Sistema según la información del inventario .....	113
Tabla 37. Interrogante 3. Sistema según la información del libro diario .....	113
Tabla 38. Interrogante 4. Sistema de gestión de insumos .....	114
Tabla 39. Interrogante 5. Interacción con el formulario del insumo .....	114
Tabla 40. Interrogante 6. Cumple con los requerimientos .....	114
Tabla 41. Interrogante 7. Aplicación en el campo laboral.....	115

## Índice de Figuras

Fig 1. Herramientas comunes para extraer información.....	24
Fig 2. Diagrama de proceso de ingresos de insumos.....	36
Fig 3. Diagrama de proceso de egresos de insumos .....	38
Fig 4. Diagrama CU: Insumo .....	40
Fig 5. Diagrama CU: Libro Diario .....	41
Fig 6. Diagrama CU: Laboratorio .....	42
Fig 7. Diagrama CU: Inventario .....	43
Fig 8. Diagrama CU: Búsqueda y visualización de insumos .....	44
Fig 9. Diagrama CU: Búsqueda y visualización del inventario .....	45
Fig 10. Diagrama CU: Búsqueda y visualización del libro diario.....	46
Fig 11. Diagrama CU: Búsqueda y visualización del ajuste de inventario .....	47
Fig 12. Diagrama CU: Visualización del inventario por fecha de caducidad o stock....	48
Fig 13. Diagrama CU: Búsqueda y visualización de los laboratorios.....	49
Fig 14. Diagrama CU: Historial del insumo en el libro diario .....	50
Fig 15. Diagrama CU: Modificar insumo.....	51
Fig 16. Diagrama CU: Modificar Laboratorio.....	52
Fig 17. Diagrama CU: Eliminar Insumo .....	53
Fig 18. Diagrama CU: Eliminar Laboratorio.....	54
Fig 19. Diagrama CU: Generar Reportes .....	55
Fig 20. Diagrama CU: Actualización del Inicio de Sesión .....	56

Fig 21. Diagrama Entidad Relación .....	58
Fig 22. Interfaz prototipo página inicial .....	59
Fig 23. Interfaz prototipo Listar Insumos .....	60
Fig 24. Interfaz prototipo Registrar Insumo .....	61
Fig 25. Interfaz prototipo Modificar Insumo .....	62
Fig 26. Interfaz prototipo Listar Laboratorios .....	63
Fig 27. Interfaz prototipo Registrar Laboratorio .....	64
Fig 28. Interfaz prototipo Modificar Laboratorio .....	65
Fig 29. Interfaz prototipo Registrar libro diario .....	66
Fig 30. Interfaz prototipo Historial libro diario .....	67
Fig 31. Interfaz prototipo Ajuste de Inventario .....	68
Fig 32. Interfaz prototipo Inventario .....	68
Fig 33. Interfaz prototipo Libro Diario.....	69
Fig 34. Interfaz prototipo Historial de Ajuste.....	69
Fig 35. Planteamiento general app multiplataforma.....	70
Fig 36. Código de la web services registrarInsumo.....	85
Fig 37. Código de la web services registrarAjuste .....	85
Fig 38. Código de la web services registrarLibroDiario .....	86
Fig 39. Código de la web services tokens inicio de sesión.....	87
Fig 40. Interfaz página inicial Administrador .....	87
Fig 41. Interfaz insumos .....	88
Fig 42. Interfaz registrar insumo .....	89
Fig 43. Interfaz modificar insumos .....	89
Fig 44. Interfaz ajuste de inventario .....	90
Fig 45. Interfaz inventario .....	91
Fig 46. Interfaz historial ajuste .....	91
Fig 47. Interfaz registro libro diario .....	92
Fig 48. Interfaz modificar libro diario .....	92
Fig 49. Interfaz historial libro diario .....	93
Fig 50. Interfaz libro diario .....	93
Fig 51. Interfaz laboratorios .....	94
Fig 52. Interfaz registrar laboratorio.....	95
Fig 53. Interfaz modificar laboratorio .....	95
Fig 54. App PWA desplegada en apache a través de una ip pública.....	96

Fig 55. Despliegue de la app en la Google play store ..... 97

## **1. Introducción**

Los servicios y la facilidad de los procesos institucionales en diversos sectores sociales, se han visto afectados por la transformación de la tecnología. La Universidad Politécnica Salesiana de Cuenca cuenta con una amplia gama de servicios que operan a través de la web, haciendo más manejable y sencilla la gestión de la institución para empleados y alumnos. Con esto, es posible tener más procesos en línea y ahorrar tiempo, mejorar la interacción de los empleados y asegurar la calidad. Es por esto que la universidad, como parte del proceso de dar paso a las tecnologías actuales y dejar procedimientos obsoletos para estos tiempos, ha desarrollado una aplicación multiplataforma dentro de la clínica veterinaria “Polivet” la cual, ya cuenta con una primera versión actualmente.

Debido a que el avance tecnológico se desarrolla rápidamente, especialmente en los sistemas de gestión computarizados, hoy en día la tecnología es de gran importancia en las clínicas veterinarias dedicadas a diferentes actividades comerciales. Estas procesan grandes cantidades de información, es por eso que es inevitable la automatización de procesos de gestión, control y búsqueda de los servicios brindados a los médicos de la clínica veterinaria “Polivet”.

Los médicos veterinarios de la clínica se ven perjudicados por la cantidad de documentos físicos almacenados con relación a los insumos que se manejan, lo que ralentiza su inventario y el registro de nuevos insumos. Esto provoca retrasos de tiempo y un mal manejo de inventario en sus informes, por lo que la veterinaria reconoció la necesidad de implementar los módulos de insumos e inventario.

Con la implementación de estos nuevos módulos, se evitará una gestión retrasada, poco óptima en cuanto al registro de los insumos y control de inventario. Por lo que, la actualización del sistema tiene como objetivo agilizar los procesos de control, creación y búsqueda de insumos que actualmente se realiza en la clínica veterinaria, reduciendo los tiempos de espera de los propietarios del establecimiento y evitando problemas de pérdida o duplicación de información. Además, solo se dará acceso al personal autorizado, protegiendo así los datos del médico, del distribuidor de los insumos, y finalmente de los insumos. La información es almacenada en una base de datos y no en papeles físicos ubicados en su lugar de trabajo.



## **2. Descripción del problema**

### **2.1.1. Antecedentes**

En la Universidad Politécnica Salesiana con su matriz en Cuenca, en el año 2011 se creó la clínica veterinaria denominada “Polivet” pensada para los alumnos de la primera promoción de Medicina Veterinaria, la cual también cuenta con atención de pacientes dentro de la jornada diurna. En sus inicios la veterinaria contó con la atención de 3 a 7 pacientes, misma cifra que se está manteniendo actualmente. Este espacio dispone de áreas como:

- Sala de espera
- Consulta externa
- Recepción
- Peluquería
- Quirófano

En la actualidad la clínica veterinaria “Polivet” trabaja con áreas amplias en consulta externa y quirófano. Además, se ha implementado un laboratorio clínico y área de tomografía, para mejorar la resolución de problemas clínicos e identificar patologías por la alta definición de las imágenes. Entre los pacientes atendidos generalmente se encuentran las mascotas como: perros, gatos, canarios, cacaúas y hámster.

La veterinaria cuenta con su propio sistema creado por las estudiantes Helen Companioni Vargas y Cinthia Malena Iza Cajia de la carrera de Computación de la Universidad Politécnica Salesiana. El sistema titulado: “Desarrollo de una aplicación multiplataforma utilizando el marco de desarrollo Ionic para la gestión de la historia clínica de la veterinaria Polivet” (Iza Cajia & Companioni Vargas, 2022), se encuentra desplegado actualmente en los servidores del grupo de investigación GiHP4C (Cloud computing-Smart Cities-High Performance Computing) correspondiente a la Universidad Politécnica Salesiana - Sede Cuenca.

El problema que presenta la versión actual del sistema radica en que la clínica Polivet, no cuenta con módulos de información acerca de la gestión de insumos, el control de inventario de cada insumo, además de que el mismo carece algunos problemas de seguridad dentro de la aplicación. Al presentarse los problemas mencionados con anterioridad, actualmente el sistema se encuentra en una etapa

de desarrollo, lo cual conlleva a que se necesite un tiempo considerable para el registro completo de insumos por parte del doctor y/o doctora que maneja la veterinaria. El control adecuado de los insumos registrados debe ser manejado mediante un inventario, el cual debe informar a los encargados de la veterinaria aspectos importantes de cada insumo, entre estos se tiene su fecha de vencimiento y su stock.

Al contar con un manejo de inventario de manera manual en las veterinarias se pueden identificar varios puntos de análisis que siempre desembocan en consecuencias negativas como: la pérdida de dinero y un mayor incremento en años posteriores, además de no brindar ningún tipo de información confiable sobre los insumos. (Cornejo Gómez, 2016).

### **2.1.2 Importancia y alcances**

Actualmente, la ciencia informática es indispensable para los diferentes campos como la medicina, con el objetivo de trabajar en conjunto con varias ramas del campo y dar a conocer ideas concretas mediante la automatización de procesos, transacciones tanto de equipos médicos como de medicamentos. (Guerrero Coveña, 2017).

En Ecuador los hospitales veterinarios brindan sus servicios a la sociedad, los cuales deben llevar un control de la información de cada procedimiento que se realiza dentro de la veterinaria como, por ejemplo: historia clínica, gestión de inventarios, etc. (Guamán & Patiño, 2016).

Al momento de implementar un sistema como parte de negocio se debe tener en cuenta el poder optar por técnicas y herramientas dentro del campo de la seguridad informática durante el diseño del módulo de la aplicación. Esto nos ayudará a establecer normas y políticas con el fin de obtener un alto rendimiento y a su vez ofrecer una mejor experiencia de usuario. (Segarra Briones & Segarra Briones, 2020).

Por lo tanto, en este trabajo se propone desarrollar módulos dentro de la aplicación multiplataforma para satisfacer las necesidades en cuanto al registro de insumos médicos, manejo de inventario, además de un control de seguridad en los módulos que presenta la clínica veterinaria “Polivet” de la Universidad Politécnica Salesiana, mediante el framework Ionic para que permita el registro

insumos médicos desde una página web y una aplicación móvil. Una vez puesta en marcha la aplicación se desea agilizar los procesos de recepción de información de cada insumo, reducir la inconformidad y pérdida de información al permitir llevar un mejor registro, control, búsqueda, seguimiento y almacenamiento de todo lo relacionado con la gestión de insumos médicos.

Recapitulando, los módulos de gestión, seguridad y manejo de inventario de los insumos de la clínica Veterinaria “Polivet” serán construidos para facilitar el trabajo tanto a los médicos como estudiantes de la clínica. Permitiendo de esta manera optimizar el control de sus insumos, stock y fecha de vencimiento de los mismos, evitando así inconvenientes de cantidad y calidad en los diferentes tipos de suministros.

### **3. Objetivos generales y específicos**

#### **3.1 Objetivos generales**

- Desarrollar un módulo para la Gestión de Insumos e inventario en la veterinaria “Polivet” Fase II de la Aplicación Multiplataforma utilizando ionic y JEE.

#### **3.2 Objetivos específicos**

- **OE1.** Estudiar y conocer los procesos dentro de la clínica veterinaria y las herramientas tecnológicas para la implementación del módulo de gestión del inventario y seguridad de la aplicación.
- **OE2.** Levantar los procesos relacionados a la gestión de insumos de la clínica veterinaria “Polivet”.
- **OE3.** Analizar y diseñar los requerimientos funcionales y no funcionales de la aplicación multiplataforma mediante la recolección, validación y documentación de los mismos.
- **OE4.** Desarrollar un módulo seguro de compra de insumos y manejo de inventario que cumpla con los requisitos recolectados y evidencie los procesos de forma digital.
- **OE5.** Realizar las pruebas funcionales y despliegue de la aplicación con la actualización del sistema.

En la sección siguiente tenemos información sobre conceptos principales para la creación de la veterinaria, como son los conceptos y las herramientas que se utilizó.

## **4. Fundamentos teóricos**

En esta sección se presentan conceptos acerca de las tecnologías y herramientas que fueron usadas durante el proceso de desarrollo del proyecto.

### **4.1 Orígenes de la medicina veterinaria**

La historia de la medicina veterinaria se remonta a la época de Aristóteles con sus obras mostrando la historia natural, incluyendo una amplia gama de enfermedades propias de los animales. A medida que el tiempo pasó se adquirió un mejor conocimiento, dando paso a los primeros medicamentos y tratados de zootécnica, los cuales fueron marcados en el libro “Catón el viejo”. Durante el siglo I d.C fue creado la palabra veterinario según Lucio Julio Columela en su libro “Los doce libros de agricultura” el cual fue usado para la descripción del pastor encargado de sanar las enfermedades de los animales. A pesar de que en sus inicios los pastores solamente sanaban enfermedades en ovinos y bovinos. Los avances en la ciencia permitieron resolver patologías en otros animales cuadrúpedos como los caballos, los cuales fueron utilizados por soldados durante las guerras (García, 2009).

En el continente americano, el ingreso de animales como bovinos y equinos durante las conquistas de los países europeos causó una gran alteración en los modelos productivos como: el transporte, la alimentación y el sector agrícola. Los colonizadores fueron los pioneros en el conocimiento de técnicas curativas hacia animales domésticos y el uso de herramientas para tratar a los mismos. Con el pasar del tiempo se logró adquirir nuevos conocimientos, logrando que se comenzaran a fundar los primeros establecimientos veterinarios en territorios como: Boston en 1883, Nueva York en 1846, Colombia en 1884 y Cuba en 1907. (García, 2009).

El territorio ecuatoriano dio a conocer el oficio de veterinaria a partir del año 1934. Sus principales actividades de cuidado animal y zootécnica partieron de la importación de ganado para la producción de leche. La demanda de ganado conllevó a la apertura de establecimientos en el país como, por ejemplo, el de la

isla Silva. Entre las actividades sanitarias a destacar, se tiene la construcción de espacios de limpieza garrapaticidas tanto en la región costa como región sierra (García Lóor, 2019).

Con el paso de los años, en Ecuador las familias han optado por incorporar en su hogar una o varias mascotas, convirtiéndolas en un miembro más del grupo familiar. En la actualidad la ley Código Orgánico Integral Penal Art 249 protege a las mascotas de algún tipo de maltrato o muerte repentina. Dado que el animal pudiera pasar por alguno de los casos mencionados, la responsabilidad recaerá sobre el dueño de la mascota (García Lóor, 2019).

#### **4.2 Gestión de insumos de mascotas**

Los medicamentos para mascotas están compuestos por la combinación de varias sustancias, las cuales tienen la finalidad de mejorar la función fisiológica de los animales. En el día a día, los veterinarios tienen la necesidad de recomendar y prescribir medicamentos para tratar las enfermedades de las mascotas (Ramírez Vásquez, 2009).

Las Veterinarias deben registrar sus insumos con el propósito de controlar la calidad de sus productos, verificando la metodología de elaboración y su registro de sanidad antes de comercializarlos en las diferentes provincias del Ecuador. (DEL AGRO-AGROCALIDAD, 2016).

#### **4.3 Gestión de inventario**

El manejo de inventario es importante en cualquier emprendimiento, ya que es la única manera en la que se puede conocer los productos que ofrece la veterinaria. La gestión de la logística de los insumos médicos debe ser controlada por la persona responsable de llevar la contabilidad, permitiendo de esta manera asegurar una mejor eficiencia de inventario al controlar la entrada y salida de cada suministro médico (Lagos Rugama, López Ruiz, & Vásquez M., 2015).

#### **4.4 Gestión Laboratorios**

En la gestión de laboratorios es importante para una empresa que lleva insumos médicos, ya que se desea conocer el nombre de las empresas que fabrican

los respectivos insumos, a fin de llevar un mayor y mejor control en el inventario de la empresa.

#### **4.5 Gestión Libro Diaro**

El libro diario se realiza los respectivos asientos contables con la que opera la empresa, pero si la empresa ya viene trabajando desde tiempo atrás se le colocará los saldos con la que se ha venido manejando la empresa anteriormente, como es su balance y la situación final con la que se ha venido manejando la empresa. Cabe recalcar que el libro diario se maneja según el sistema con la que se emplee en la empresa (HABER).

#### **4.6 Ingeniería de Software**

La ingeniería de software tuvo sus inicios entre los años 70 y 80. La Guía SWEBOOK - ISO 19759 entiende la definición de ingeniería de software como “la aplicación de un enfoque sistemático, disciplinado y cuantificable para el desarrollo, operación y mantenimiento de software; es decir, la aplicación de la ingeniería al software” (Al-Sarayreh, K.T & Abran, 2021).

La encargada de estudiar los periodos de vida del desarrollo de software desde sus comienzos hasta los mantenimientos que recibe en la actualidad es la disciplina informática (Wong & Li, 2021).

Dentro de la ingeniería de software se encuentra el “ciclo de vida de software” está constituido por fases:

- **Planteamiento:** En esta fase se debe determinar lo que necesita el software, el modelo de negocio, y el instrumental necesario para su desarrollo.
- **Diseño:** En esta fase se implementan las ideas con base en las características que poseerá durante la estructura de un software.
- **Elaboración:** En esta fase se puede apreciar el desarrollo del software.
- **Transición:** En esta fase se realiza la implementación y se adjudica el software al usuario.

## **4.7 Ingeniería de requerimientos**

En palabras de (Zave, 1997):

“La ingeniería de requisitos es la rama de la ingeniería de software preocupada por los objetivos del mundo real para las funciones y limitaciones en los sistemas de software. Está también preocupado por la relación de estos factores a especificaciones precisas de comportamiento del software y su evolución a lo largo del tiempo y entre familias de software”.

Determina los procesos necesarios en el desarrollo de un software (Chaves, 2005)

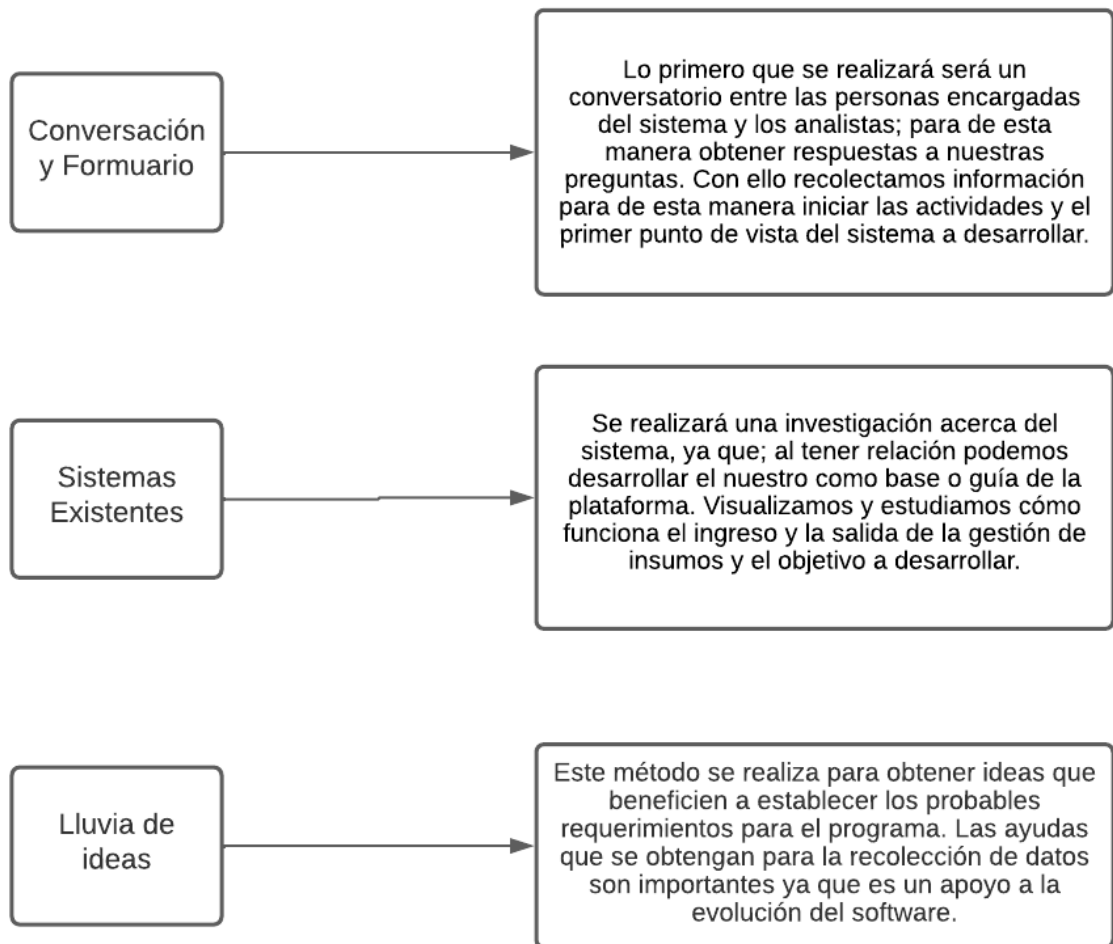
En palabras de (Dick & Jackson, 2017):

Ingeniería de requerimientos es “el subconjunto de ingeniería de sistemas en cuestión con descubrir, desarrollar, rastrear, analizar, calificar, comunicar y gestionar los requisitos que definen el sistema en sucesivos niveles de abstracción”.

### **4.7.1 Obtención de información**

En esta sección se extrae la información general del problema, se determina el alcance, los motivos de la aplicación y stakeholders a futuras versiones del sistema. Todos los datos son obtenidos a través de entrevistas e informes con el fin de lograr determinar requisitos y procesos.

- Herramientas y técnicas (Oliveros & Antonelli, 2015):



*Fig 1. Herramientas comunes para extraer información*

#### **4.7.2 Especificación de requerimientos**

En esta actividad se busca entender mejor lo que desean los stakeholders y plasmarlo en requisitos funcionales y no funcionales. Los requisitos no funcionales van enlazado a un carácter técnico de la aplicación como: ¿qué lenguaje de programación se emplea?, ¿qué base de datos usar?, ¿cuál es el lenguaje de programación a utilizar?, disponibilidad, aplicabilidad de estándares, ambiente de desarrollo y transportabilidad. Mientras que los requisitos funcionales son los que establecen el comportamiento del sistema, con un enfoque en lo funcional.

Esta actividad tiene como objetivo entender las necesidades de los stakeholders y representarlos en requisitos funcionales y no funcionales.



- **Validación y priorización de requerimientos**

Esta tarea se lleva a cabo una vez definidos los requisitos para garantizar que lo propuesto es lo que se quiere lograr y que los requisitos resultantes sean priorizados en bases con temas referidos a necesidad y expectativa del cliente.

- **Definir la arquitectura del sistema (Diagramas UML (lógico))**

Esta tarea explica los componentes principales de la aplicación con el fin de encaminar su desarrollo. Por lo tanto, aún garantiza la estabilidad en el tiempo, permitiendo cambios, adiciones y modificaciones de requisitos resguardando las tareas iniciales. Establecer la forma en la que se construya el sistema a partir de diagramación UML, obteniendo de esta manera una perspectiva más óptima de las entidades con sus respectivos atributos.

## **4.8 Aplicaciones multiplataforma**

Las aplicaciones multiplataformas tienen como objetivo la construcción de una sola aplicación que pueda ser ejecutada en varias plataformas. La elaboración multiplataforma es necesaria en la actualidad debido a que las aplicaciones son independientes al poder ejecutarse en diversas plataformas, sean estas web o móviles. Este tipo de aplicaciones comúnmente son programadas en un lenguaje de programación (Al Abid & Karim, 2017).

## **4.9 Herramientas de Desarrollo Frontend**

### **4.9.1 Framework Ionic**

Dentro de los frameworks para el desarrollo de aplicaciones multiplataformas Ionic se enfoca en la experiencia e interacción del usuario con la interfaz, lo realiza mediante gestos, animaciones y tipografía. Sus componentes son de fácil uso, además de tener una buena adaptabilidad a las diversas plataformas. El framework se complementa bien con lenguajes como: JavaScript, CSS (hoja de estilos en cascadas) y HTML (Lenguaje de etiquetas de hipertexto). En la actualidad, Ionic es considerado una buena opción para la construcción de aplicaciones multiplataforma (Ionic Docs, 2022).

## **4.9.2 Typescript**

Microsoft lanzó el lenguaje de programación de código abierto llamado Typescript en 2012. Al ser robustamente tipado, le permite construir específicos tipos de datos. También es una extensión de JavaScript para ayudar a crear aplicaciones a gran escala al proporcionar módulos, clases e interfaces (Bierman & Torgersen, 2014).

El enfoque de Typescript está basado en mejorar la experiencia del desarrollador mediante la escritura de código, lo que facilita su uso (Freeman, 2021).

Typescript proporciona múltiples ventajas entre las áreas más importantes se encuentran:

- Seguridad: dentro de esta área typescript permite la detección temprana de muchos errores previo a la ejecución del código.
- Legibilidad: es de vital importancia durante la fase de diseño, ya que ayuda a entender de mejor manera la diagramación UML.
- Herramientas: los estilos claros facilitan la comprensión del código por parte de los usuarios.
- Rendimiento: typescript ayuda a reducir el tiempo de ejecución del código.

Las ventajas que ofrece typescript supera por mucho las desventajas que pueda tener, esto se debe a que conforme un proyecto vaya adquiriendo mayor complejidad, aumentarán los beneficios de uso (Jansen & H., 2015).

## **4.10 Herramientas de Desarrollo Backend**

### **4.10.1 Java**

Java es un lenguaje de programación orientado a objetos creado en 1991 y desarrollado originalmente por el equipo de James Gosling en conjunto con él. Nació con la idea de programar electrodomésticos. Con el pasar del tiempo se lanzaron nuevas versiones para el diseño de aplicaciones web y con la peculiaridad de que no dependía de internet. (Ogala & Ojie, 2020).

Java se define como un lenguaje robusto, seguro y simple al momento de crear aplicaciones tradicionales y rápidas tanto en aplicativos móviles como los de la web.

#### **4.10.2 Seguridad con Angular & JEE (Oracle Platform, Enterprise Edition)**

La seguridad dentro de un sistema o aplicación es de vital importancia durante la etapa de desarrollo. La manera en la cual se debe proteger los datos es analizando el código fuente de estos, ya sea para analizarlos en manera de revisión o realizar correcciones en caso de ser necesario; a este proceso se le denomina análisis estático. Dentro de la seguridad existe también el proceso dinámico, el cual se da una vez ejecutada la aplicación mediante el uso de credenciales. Todos estos procesos ayudan a optimizar el rendimiento de un sitio web o aplicación móvil (Munoz & Villalba, 2012).

La seguridad en Angular es de suma importancia, ya que nos evitamos a que nuestra información quede expuesta y además, que corra el riesgo de ser inyectada por algún código malicioso, la cual se lo denomina como Cross Site Scripting (XSS), por ello se debe tomar en cuenta aspectos como; tener la última versión, no utilizar APIs que contengan un riesgo de seguridad, tome en cuenta las plantillas que utiliza, estas y otras muchas recomendaciones se deben tomar en cuenta y trabajar en conjunto con técnicas de TypeScript y JavaScript (Danielecki, 2019).

#### **4.10.3 Web services**

Este servicio interactúa con los clientes por medio de protocolos y componentes que son llamados por medio de la red. Este tipo de aplicación está orientado a servicios (SOA) brindando a los servicios y clientes navegar por un sin número de plataformas y varios escenarios de desarrollo. También su uso se lo realiza mediante HTML y la integración a la base de datos a través de XML (Lenguaje de Mercado Extensible) permitiendo su correcta integración y la adecuada comunicación (Hernández, 2007). Los componentes claves con las que cuentan son:

WSDL (lenguaje de descripción de servicios web): Este lenguaje explica lo que realiza un servicio web basado en XML, esto nos permite saber como utilizar los mensajes y como poder contestarlo (Morales, 2010).

SOAP (Simple Object Access Protocol): Este es un tipo de protocolo; su comunicación se basa mediante XML; esto lo realiza mediante varios procesos (Castro, 2013).

JAX-WS (Java API para XML- basado en Servicios Web): Esta API es un modelo de java el cual es usado para llamar a los servicios web por medio de SOAP (. Hidalgo Macas & Jiménez Acaro, 2016).

JAX-RS: Este tipo API que permite la creación de servicios web mediante la arquitectura REST permitiendo la simplificación y facilidad del desarrollo; una arquitectura ideal para los desarrolladores (Galarza Acosta, 2017).

#### **4.11 PWA (aplicación web progresiva)**

Las aplicaciones PWA son prioritarias durante el desarrollo de sistemas de información. La diferencia con el resto de aplicaciones habituales radica en que estas pueden ser ejecutadas desde cualquier sistema o navegador web. Este tipo de aplicaciones pueden trabajar en conjunto con APIs (application programming interface), como por ejemplo los Service Worker ejecutados por los navegadores en un segundo plano, ya que no se necesita de un sitio web o de la interacción por parte del usuario para la implementación de sus funcionalidades solicitadas.

Esto da inicio a la posibilidad de comportamientos similares al de la app nativa con actividad fuera de línea, las notificaciones automáticas y los iconos en las aplicaciones. (Aguirre, et al., 2019).

#### **4.12 Aplicaciones móviles híbridas**

Las aplicaciones móviles con modelos híbridos se producen al combinar diferentes aplicaciones, en donde se implementan dentro de una sola app móvil componentes tanto web como nativos. La experiencia de usuario como la ejecución en múltiples plataformas son posibles dentro de un modelo híbrido.

Para la creación de este tipo de aplicaciones se necesita de tecnología web estándar como: JavaScript, CSS y HTML (Tiwari, Prakash, Groß, & Hammer,

2020). Por otro lado, también provee un contenedor de código web y una API de JavaScript que une las peticiones a partir del servicio levantado en la web, empaquetando el mismo para una posterior implementación y distribución de aplicaciones móviles híbridas. El framework a usar en las aplicaciones híbridas puede variar según las preferencias del desarrollador. Entre los principales frameworks desde el punto de vista nativos tenemos: Ionic, Córdova y Apache.

#### **4.12.1 Ventajas:**

- Reutilización de código fuente en diferentes SO (Sistemas Operativos)
- Web services portables en diversas plataformas.
- Procedimientos en etapa de desarrollo simples y de bajo costo.

#### **4.12.2 Desventajas:**

- Mal empleo de las tecnologías web y nativas causando problemas de privacidad y seguridad.
- La implementación con las funciones nativas del dispositivo es más lenta y poco fluida.

### **4.13 Bases de datos relacionales**

Es una base de datos que consta de tablas relacionadas con un conjunto de campos y registros. También colabora junto con ACID (atomicity, consistency, isolation, durability) en la inclusión de términos, atomicidad indica si la transacción debe ser ejecutada totalmente o parcialmente y en algunos casos no llevar a cabo la transacción, la consistencia es la encargada del paso de movimientos válidos hacia otros movimientos respetando las reglas establecidas, aislamiento la cual puede ejecutar muchas transacciones dentro de un registro resguardando su independencia. Finalmente, la durabilidad es la que conserva la transacción con los cambios efectuados (Valverde, Portalanza, & Mora, 2019).

#### 4.14 Base de datos MySQL

MySQL fue desarrollado en 1980 por el sueco David Axmark, Michael Widenius y Allan Larsson. A día de hoy es la base de datos más conocida de open source debido a su credibilidad, rapidez y fácil uso. Su licencia dentro del sistema de gestión funciona de manera pública y comercial. En cuanto a la arquitectura es Cliente-Servidor lo cual ofrece un mejor autonomía y rendimiento. Su compatibilidad con SQL posibilita que el gestor de la base de datos permita procesos de administración y recuperación de datos. (Setyawati, Wijoyo, & Soeharmoko, 2020).

#### 4.15 Metodología de desarrollo Scrum

Metodología flexible de desarrollo de software para un grupo de desarrolladores con la facultad de introducir nuevos cambios y entregas en menor tiempo. Scrum al ser un marco incremental e iterativo, es usado actualmente para desarrollar software (Cano, 2021).

Entre los componentes más importantes de la metodología Scrum se encuentran los Sprints, los cuales hacen referencia a un ciclo de trabajo al definir periodos de tiempo en los cuales se realizarán entregas o se mostrarán avances. Las etapas de los Sprint son:

- **Preparación del Sprint:** Juntas planificadas con el propósito de decretar tareas o eventos a realizar donde su finalidad es encontrar el objetivo de satisfacción del Sprint.
- **Scrum cotidiano:** Juntas en cortos lapsos de tiempo junto al equipo de trabajo. En la junta se dialoga acerca de lo realizado en la jornada pasada con el fin de dar a conocer las dificultades encontradas y también se define el cronograma del día actual (Cano, García-Camús, Garzás, Moguerza, & Sánchez, 2021).
- **Inspección del Sprint:** Una vez finalizado cada Sprint se procede a realizar una reunión, donde el equipo se retroalimenta y posteriormente se dedica a la co-creación (Cano, García-Camús, Garzás, Moguerza, & Sánchez, 2021).

- **Retrospectivación del Sprint:** Juntas de retroalimentación en donde se realiza un análisis minucioso sobre los problemas encontrados durante el proceso y donde posteriormente se plantearán posibles soluciones.

Las herramientas más importantes para la gestión dentro del marco de trabajo Scrum son:

- **Product Backlog:** Lista con los requisitos y trabajos a realizar en el proyecto.
- **Sprint Backlog:** Lista conformada tareas y objetivos específicos a desarrollarse dentro del Sprint.

Además, existen roles en Scrum, cada uno de los cuales desarrolla una labor en específico dentro del grupo de trabajo. Los roles son:

**Scrum Master:** Persona responsable de implementar métodos, lineamientos y técnicas establecidas en el proyecto. Además, de ser un guía para el equipo de trabajo, ya que debe dar solución a los problemas presentados por el equipo durante el desarrollo del Sprint.

**Product Owner:** Persona responsable de establecer una comunicación entre cliente y el equipo de trabajo.

**Equipo de trabajo:** Personas encargadas de cumplir con el Sprint mediante los objetivos planteados, y responsables de llevar a cabo el desarrollo del software.

## 5. Marco metodológico

Esta sección detalla los requerimientos, diseños claves para el desarrollo de la aplicación multiplataforma.

### 5.1 Requerimientos funcionales

La presente aplicación multiplataforma ayuda con la gestión de insumos, la gestión de laboratorio, la gestión del inventario, la gestión del libro diario y la manera en como se visualizará el insumo, registro de insumos, registro del laboratorio. Todo ello en razón al control de los insumos con los siguientes requisitos detallado a continuación:

- Controlar usuarios y perfiles de acceso
- Registrar Laboratorio
- Registrar Insumos
- Registrar Libro Diario
- Registrar Ajuste
- Buscar y Visualizar insumo
- Buscar y Visualizar inventario
- Buscar y Visualizar Libro diario
- Buscar y Visualizar Ajuste del inventario
- Visualizar Inventario por fecha de caducidad o stock
- Buscar y Visualizar laboratorio
- Historial del insumo en el libro diario
- Modificar insumo
- Modificar laboratorio
- Eliminar insumo
- Eliminar laboratorios
- Generar Reportes

Como se puede visualizar en esta sección se colocó los requisitos funcionales, dichos requisitos se puede visualizar de manera más detallada en la sección *5.3.3 de diseño y arquitectura*; en la que se encuentra junto a cada caso de uso.



## 5.2 Requerimientos no funcionales

De manera general vamos a colocar las características que se aplicaron en la creación de nuestro módulo de insumos y la seguridad.

- Interfaz de la aplicación
- Simplificar en la utilización del aplicativo
- Modelo de la interfaz y sus características multiplataformas
- Realización
- Solidez de la información

A continuación, se muestra de manera más detallada en tablas cada uno de los requerimientos no funcionales.

*Tabla 1. Requerimientos no funcionales: Interfaz del sistema*

<b>Denominación del Requerimiento:</b>	Interfaz de la aplicación.
<b>Particularidades:</b>	En la aplicación se puede observar una interfaz médica de fácil uso para que se pueda utilizar.
<b>Denominación de requerimiento:</b>	El aplicativo debe ser de fácil manejo a fin de ser intuitiva y sencilla.
<b>Precedencia del requerimiento:</b> Alta	

*Tabla 2. Requerimientos no funcionales: Simplifica en el uso del sistema*

<b>Denominación del Requerimiento:</b>	Simplifica en la utilización del aplicativo.
<b>Particularidades:</b>	Mediante la interfaz de usuario nos muestra una modelo de solución de esta manera los usuarios del programa tendrán una mejor facilidad al momento de manejar la aplicación.
<b>Definición del</b>	

<b>requerimiento:</b>	El sistema de ayuda debe estar acorde a la interfaz (Esto permite que los usuarios que tengan poco conocimiento sean de ayuda al manejo del programa).
<b>Precedencia del requerimiento:</b>	
Alta	

*Tabla 3. Requerimientos no funcionales: Diseño de la interfaz a características multiplataformas.*

<b>Denominación del Requerimiento:</b>	Modelo de la interfaz y sus características multiplataformas.
<b>Particularidades:</b>	La aplicación debe contar con una interfaz practica para el manejo por parte de los usuarios, se debe tener particularidades relacionadas a la web y aplicativo móvil de la veterinaria.
<b>Definición del requerimiento:</b>	Según las particularidades de la web debe tener concordancia con respecto a la veterinaria, mediante la cual se encuentra la gestión de insumos de la veterinaria.
<b>Precedencia del requerimiento:</b>	
Alta	

*Tabla 4. Requerimientos no funcionales: Desempeño*

<b>Denominación del Requerimiento:</b>	Realización
<b>Particularidades:</b>	El aplicativo asegura a los usuarios el correcto funcionamiento y almacenamiento de los datos de esta manera se garantiza el resguardo de los mismos.

<b>Definición del requerimiento:</b>	Cuando se tiene una garantía acerca de un correcto funcionamiento del sistema, el usuario podrá realizar las gestiones pertinentes como son las de registrar, actualizar, modificar, etc.
<b>Precedencia del requerimiento:</b> Alta	

*Tabla 5. Requerimientos no funcionales: Seguridad en información*

<b>Denominación del Requerimiento</b>	Solidez de la información
<b>Particularidades:</b>	La solidez en cuanto a la información es de suma importancia para los usuarios.
<b>Definición del requerimiento:</b>	De acuerdo a la confiabilidad con respecto a la seguridad de la información, de los distintos datos que se manejan tales como los módulos, paths, etc.
<b>Precedencia del requerimiento:</b> Alta	

En la siguiente sección se hablará acerca del diseño y arquitectura en donde se especificará los casos de uso, junto a sus respectivos requerimientos funcionales, las cuales se mencionó en esta sección; además, se mostrará su diagrama entidad relación, su arquitectura multiplataforma y sus interfaces.

### 5.3 Diseño y arquitectura

Esta sección describe en detalle el análisis y el diseño utilizados en la aplicación para definir los límites que tiene el sistema:

#### 5.3.1. Diagrama de procesos

El diagrama de procesos nos presenta el orden secuencial y la relación de cada uno de los procesos que maneja la veterinaria.

- **Ingreso de Insumos**

El proceso se maneja con la llegada del proveedor a la veterinaria con los respectivos insumos. El médico veterinario los receta y los busca en el inventario para verificar si el producto existe en su inventario; de existir simplemente se realiza un ajuste de inventario actualizando su stock, caso contrario si el insumo no se visualiza en el inventario, el médico veterinario es el encargado de registrarlo y hacer el respectivo ajuste de inventario.

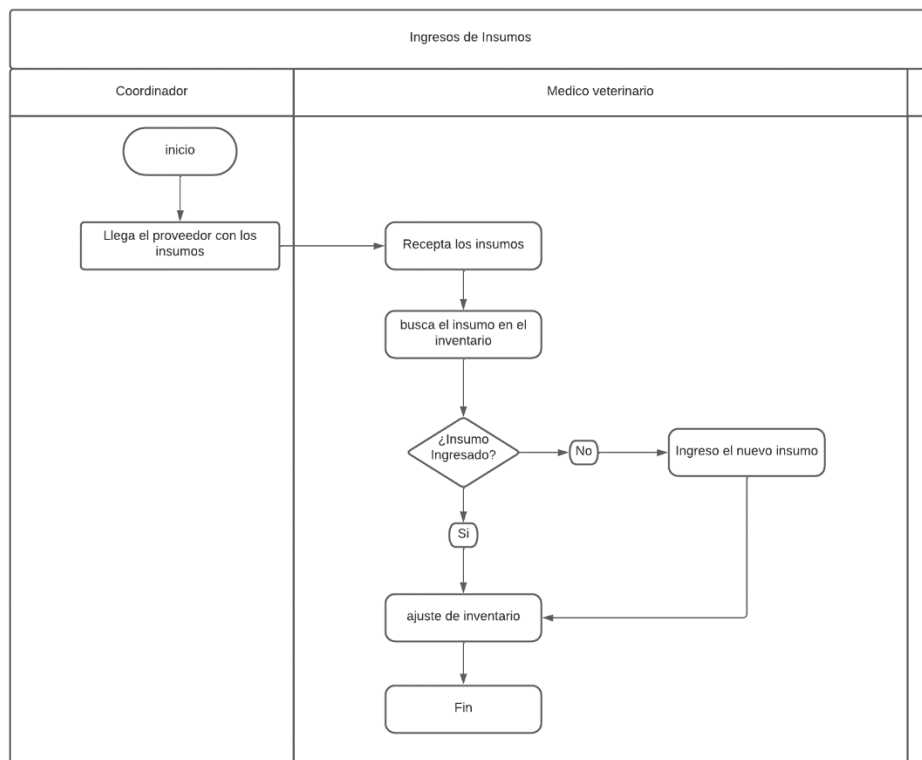


Fig 2. Diagrama de proceso de ingresos de insumos

- **Egreso de Insumos**

El proceso se maneja con la búsqueda de insumos para revisar el stock de cada uno de ellos. En caso de que un insumo esté por agotarse, el médico veterinario es el encargado de notificar al coordinador, quien procederá a realizar el pedido. Cuando la coordinación ejecute la entrega de los insumos, el veterinario verificará que los insumos sean los solicitados y posteriormente procederá a ingresarlos. Los insumos utilizados son ingresados con fecha y cantidad exacta a fin de tener un control de insumos al día. Los insumos pueden ser registrados por 2 motivos: el primer motivo se da debido a prácticas estudiantiles en donde se ocupan insumos o el segundo motivo puede ser personas externas quienes deben pagar por sus insumos; en este caso el médico veterinario receipta el dinero y posteriormente será llevado a tesorería.

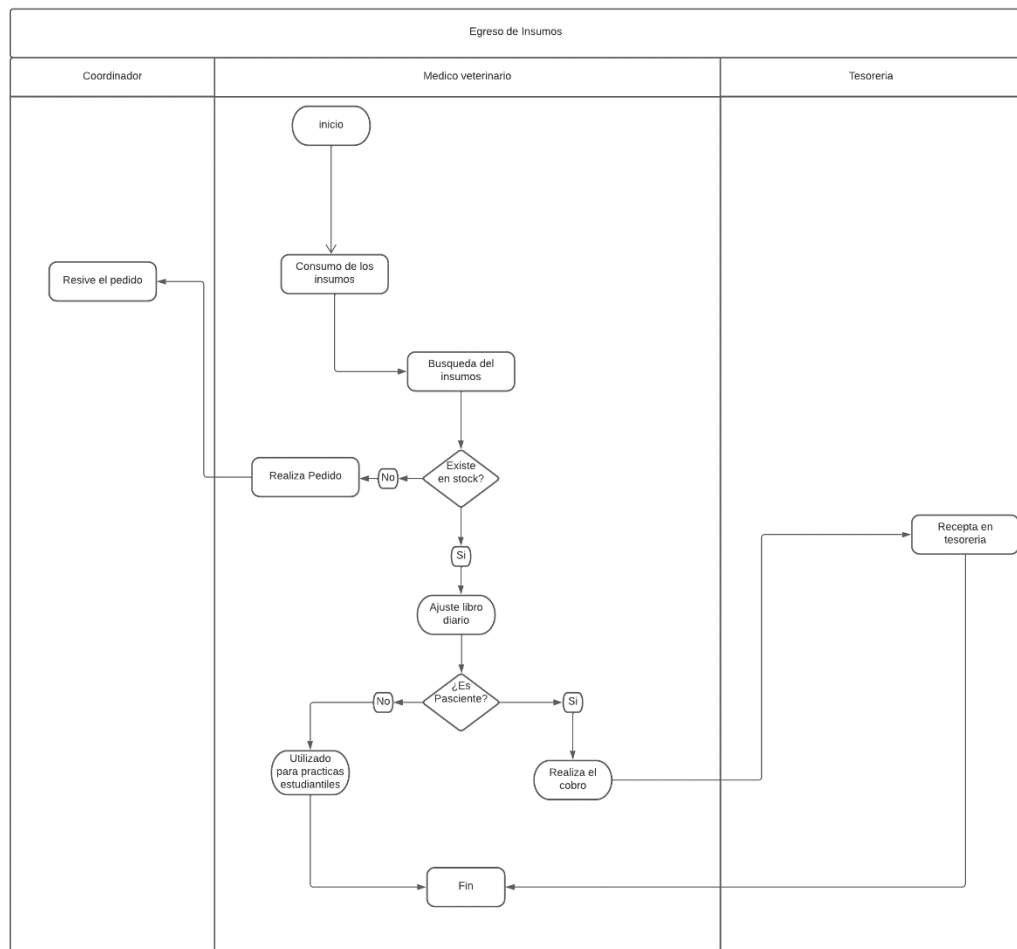


Fig 3. Diagrama de proceso de egresos de insumos

Tabla 6.Revisión diagrama de procesos

Revisado por:	Firma
Dra. Carmen Angélica Yanzaguano Romero.	

### 5.3.2. Especificaciones de los casos de uso

- Insumo

En el módulo insumo se tienen los siguientes casos de uso: registrar Insumo, buscar y visualizar el insumo, modificar insumo y finalmente eliminar insumo.

- Libro Diario

Este módulo contiene los siguientes casos de uso: registro libro diario, buscar y visualizar libro diario y el historial del insumo en el libro diario.

- Inventario

El módulo inventario cuenta con los siguientes casos de uso: Registro Ajuste, Buscar y visualizar inventario, Buscar y visualizar Ajuste de inventario, Visualizar inventario por fecha de caducidad y stock y Generar reportes (inventario, libro diario, historial de ajuste).

- Laboratorio

En la gestión de laboratorios se tienen los siguientes casos de uso: Registro Laboratorio, Buscar y visualizar laboratorios, Modificar laboratorios, Eliminar laboratorios.

- Control de usuarios y perfiles de acceso

Dentro de este caso de uso existente tenemos únicamente el caso de uso actualizar inicio de sesión.

### 5.3.3. Descripción de los casos de uso

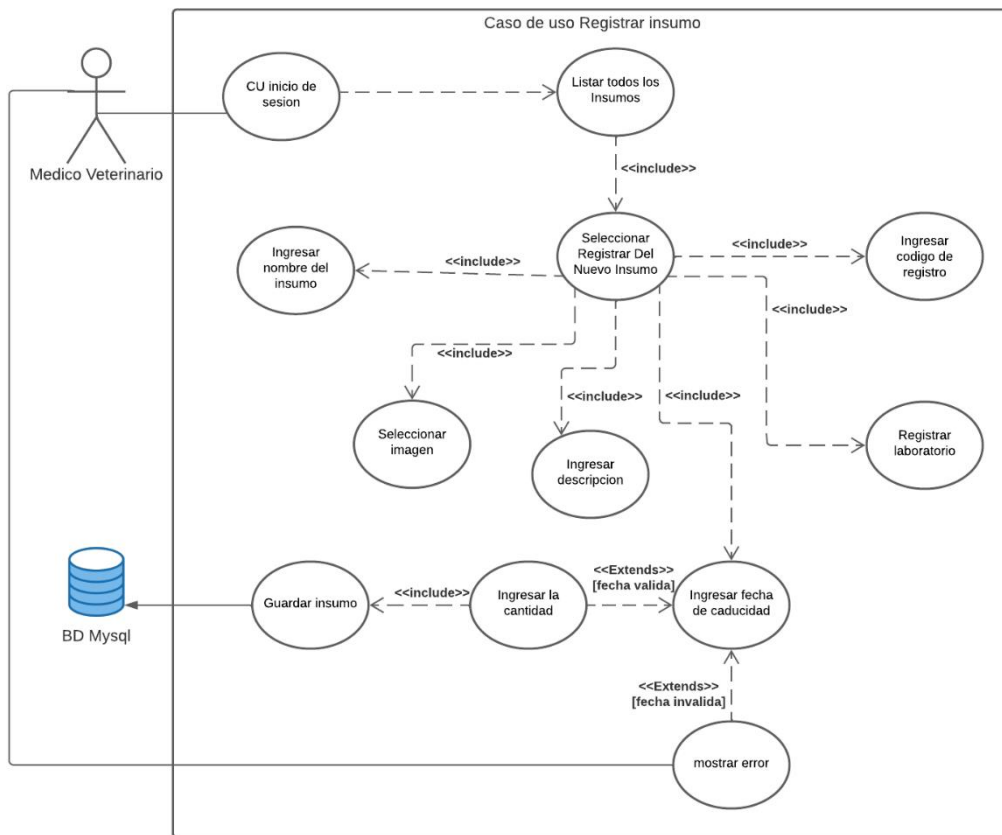


Fig 4. Diagrama CU: Insumo

Tabla 7.Requerimiento Funcional: Registro Insumos

Desarrollo: Registro de insumos	
<b>Precondición</b>	Registro de insumos médicos de la veterinaria por parte del médico veterinario.
<b>Entrada</b>	Nombre insumo, imagen, descripción, laboratorio, fecha de caducidad, cantidad
<b>Proceso</b>	El sistema solicita información de los insumos médicos para la creación de un control de los implementos y medicamentos.
<b>Salida</b>	Se manda un aviso de fallo en caso de no llenar correctamente los campos



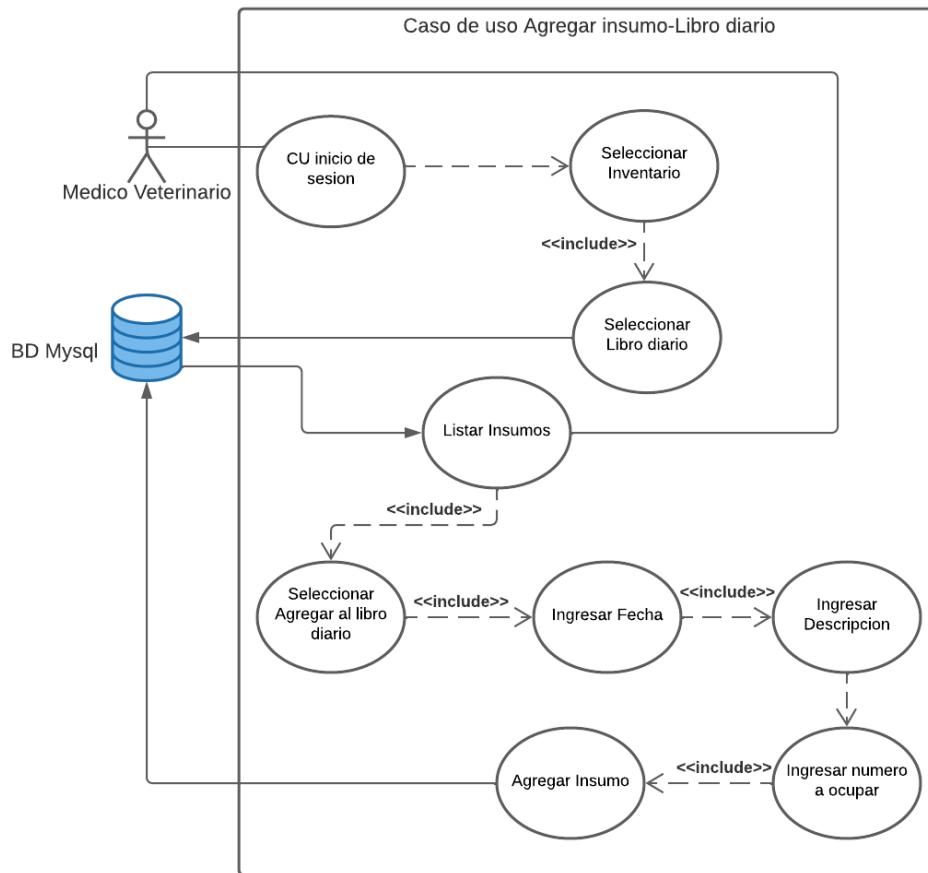


Fig 5. Diagrama CU: Libro Diario

Tabla 8.Requerimiento Funcional: Registro Libro Diario

Desarrollo: Registro de Libro Diario	
<b>Precondición</b>	El medico ingresara los datos en el libro diario
<b>Entrada</b>	Fecha libro diario, descripción, numero a ocupar, insumo
<b>Proceso</b>	El sistema contabiliza mediante los asientos contables cada transacción (DEBE Y HABER)
<b>Salida</b>	Se mostrará un error en caso de no cumplir de manera correcta el llenado de los campos sugeridos.

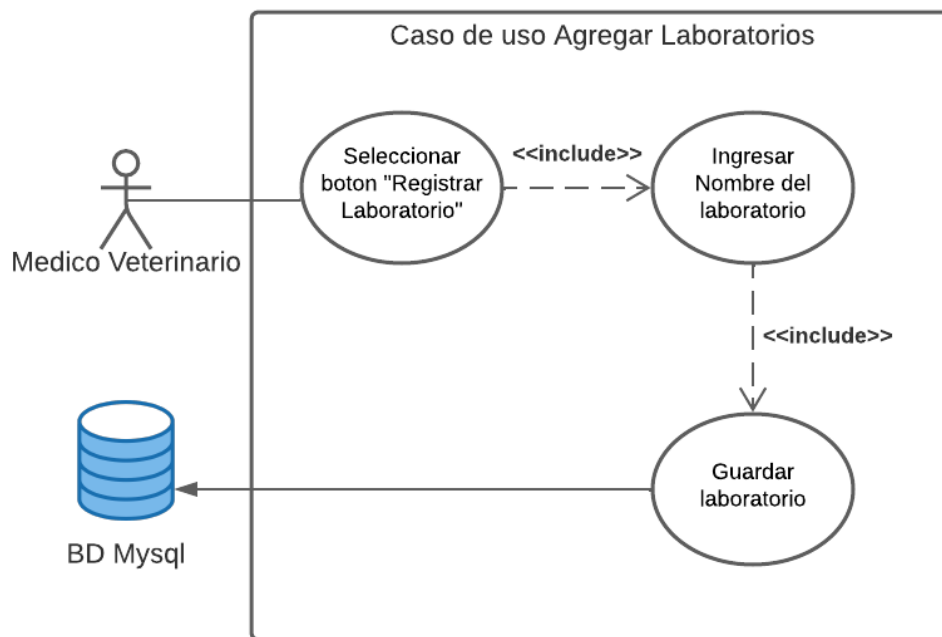


Fig 6. Diagrama CU: Laboratorio

Tabla 9.Requerimiento Funcional: Registro Laboratorio

Desarrollo: Registro de Laboratorio	
<b>Precondición</b>	El usuario registrara el nombre de laboratorio
<b>Entrada</b>	Nombre
<b>Proceso</b>	El sistema permite ingresar el nombre del laboratorio para un mejor control en los insumos médicos
<b>Salida</b>	Se debe llenar el campo requerido para que no muestre un error.

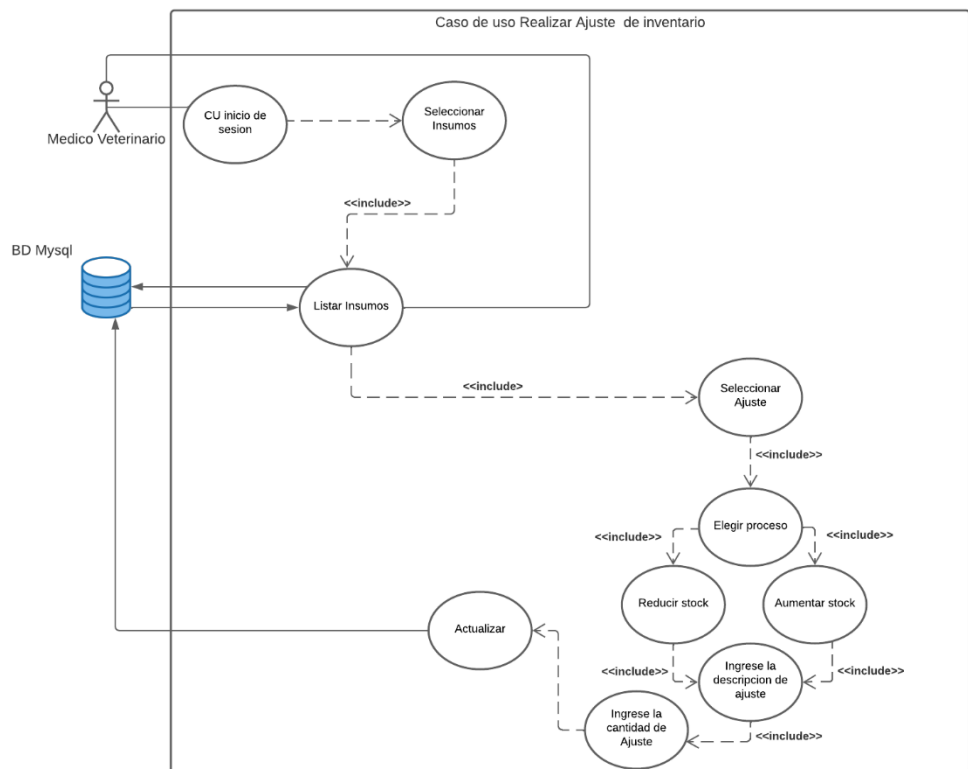


Fig 7. Diagrama CU: Inventario

Tabla 10. Requerimiento Funcional: Registro Ajuste

Desarrollo: Registro del Ajuste	
<b>Precondición</b>	El usuario o medico realizara ajuste respectivo en el inventario
<b>Entrada</b>	Proceso, descripción, cantidad, inventario
<b>Proceso</b>	Esta parte permite que se realice algún ajuste en el libro diario ya sea en el caso de algún fallo a la hora de ingresar u ocupar el insumo.
<b>Salida</b>	El error ocurrirá cuando los campos no se encuentren llenados de manera indicada en las secciones.

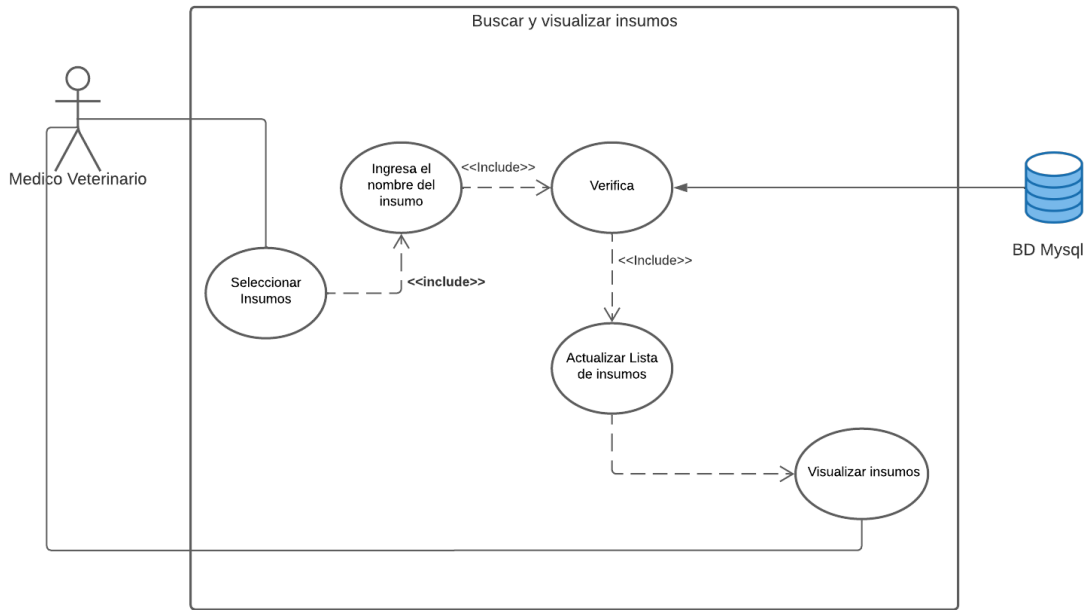


Fig 8. Diagrama CU: Búsqueda y visualización de insumos

Tabla 11. Requerimiento Funcional: Buscar y Visualizar el insumo

Desarrollo: Buscar y visualizar el insumo	
<b>Precondición</b>	El médico podrá buscar por nombre el insumo requerido.
<b>Entrada</b>	Consulta insumo medico
<b>Proceso</b>	Al colocar el nombre del insumo permitirá visualizar ese insumo en específico.
<b>Salida</b>	En requerimiento funcional se podrá visualizar los datos de manera general en la tabla por el nombre.

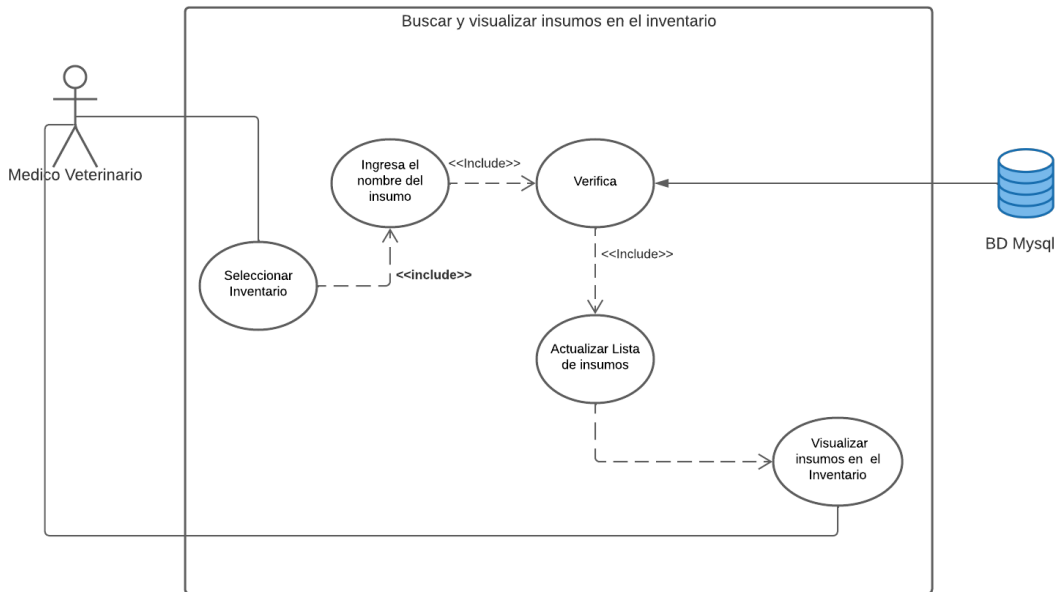


Fig 9. Diagrama CU: Búsqueda y visualización del inventario

Tabla 12.Requerimiento Funcional: Búsqueda y Visualización inventario

Desarrollo: Búsqueda y Visualizar insumo	
<b>Precondición</b>	Ayuda al usuario a encontrar el insumo dentro del inventario
<b>Entrada</b>	Buscar en el inventario
<b>Proceso</b>	Al colocar el nombre del insumo permitirá visualizar ese insumo en específico en la tabla
<b>Salida</b>	En esta sección se podrá visualizar los datos de manera general o ya sea al buscar por nombre de insumo. Además, puede realizar dicho ajuste si lo requiere.

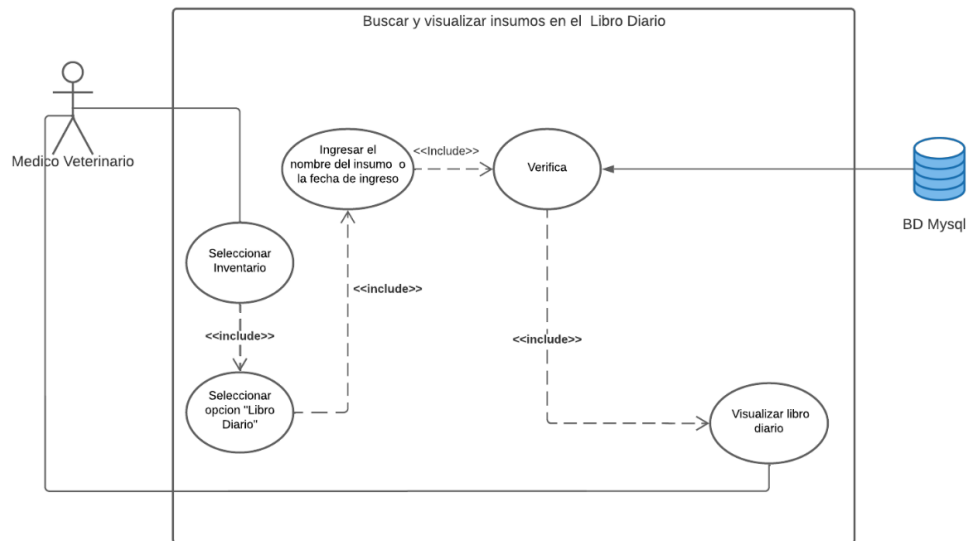


Fig 10. Diagrama CU: Búsqueda y visualización del libro diario

Tabla 13.Requerimiento Funcional: Búsqueda y Visualizar Libro diario

Desarrollo: Búsqueda y Visualizar el libro diario	
<b>Precondición</b>	La vista del libro diario es el registro de este.
<b>Entrada</b>	Nombre, fecha de ingreso
<b>Proceso</b>	Se podrá buscar por nombre y fecha en el que se haya realizado el uso del insumo en el libro diario, a más de ello se puede imprimir ya que el archivo se puede descargar en un documento csv.
<b>Salida</b>	Mostrar los datos ya sea por nombre o fecha de registro del libro diario.

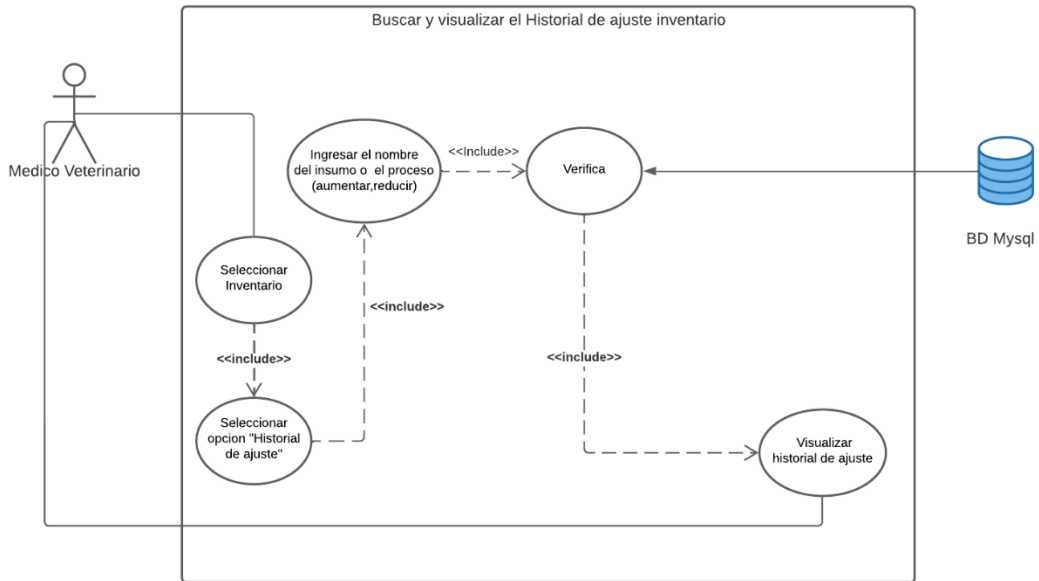


Fig 11. Diagrama CU: Búsqueda y visualización del ajuste de inventario

Tabla 14. Requerimiento Funcional: Búsqueda y Visualizar Ajuste de inventario

Desarrollo: Búsqueda y Visualizar Ajuste de inventario	
<b>Precondición</b>	Se visualiza los datos del ajuste del inventario por parte del médico encargado del mismo.
<b>Entrada</b>	Insumo y proceso
<b>Proceso</b>	Al ingresar el nombre del insumo médico se podrá realizar el ajuste ya sea aumentar o reducir, se podrá visualizar en una tabla, de igual manera se podrá guardar o imprimir, ya que permite descargar en un archivo de tipo csv.
<b>Salida</b>	En esta sección se podrá visualizar los datos de manera general o ya sea al buscar por nombre de insumo o fecha al que se realizó el registro en el libro diario.

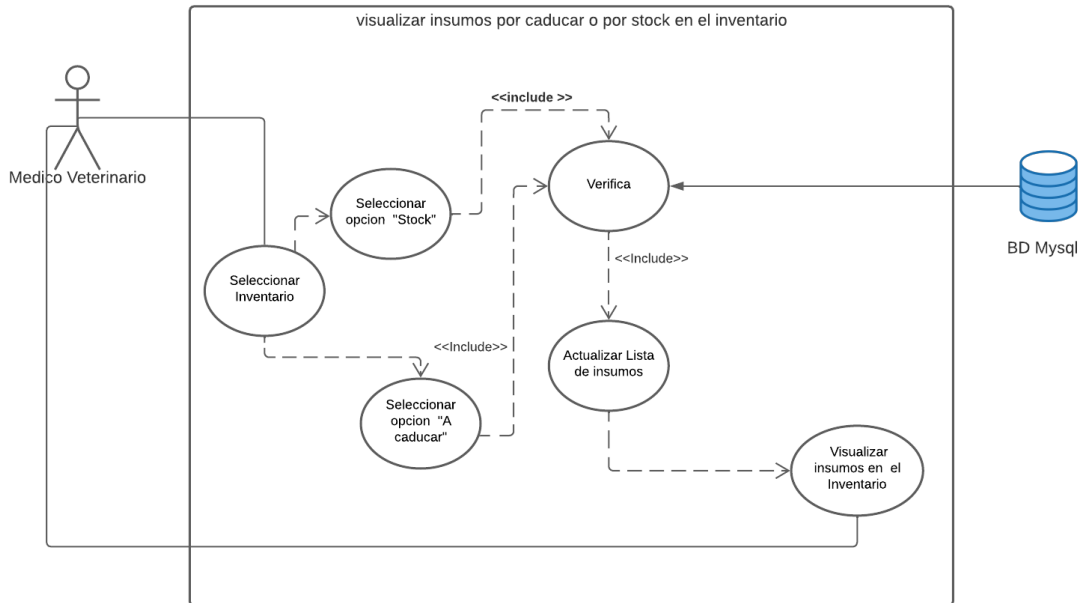


Fig 12. Diagrama CU: Visualización del inventario por fecha de caducidad o stock

**Tabla 15. Requerimiento Funcional: Visualizar Inventario por fecha de caducidad o stock**

Desarrollo: Visualizar Inventario por fecha de caducidad o stock	
<b>Precondición</b>	Vista de datos en el inventario por parte del médico veterinario
<b>Entrada</b>	Stock e inventario
<b>Proceso</b>	El sistema puede mostrar los insumos médicos de esta manera saber los insumos disponibles y faltantes. Adicional a ello el sistema nos permite imprimir un documento tipo csv con el reporte de los mismos.
<b>Salida</b>	Stock modificado en caso de ingresos o egresos de insumos



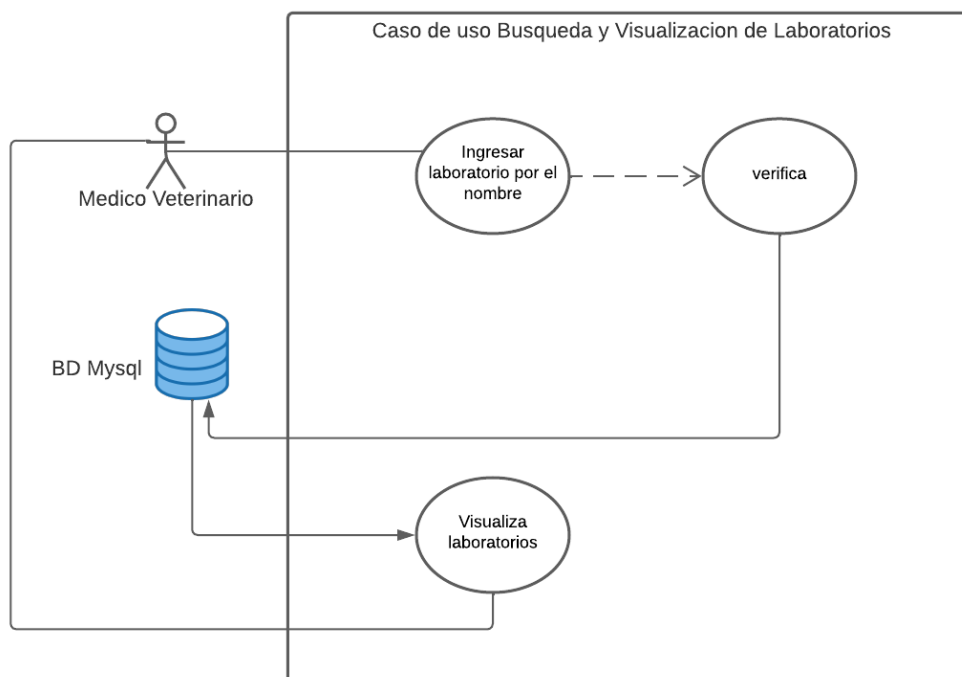


Fig 13. Diagrama CU: Búsqueda y visualización de los laboratorios

Tabla 16. Requerimiento Funcional: Buscar y Visualizar laboratorio

Desarrollo: Buscar y Visualizar laboratorio	
<b>Precondición</b>	Buscar el laboratorio por nombre.
<b>Entrada</b>	Consulta en el laboratorio
<b>Proceso</b>	Cuando se coloca en nombre del laboratorio permitirá visualizar ese laboratorio en específico. Cuando lo modifiquemos o eliminemos se lo realizara por el nombre del laboratorio. Al eliminar solo inhabilitaremos al laboratorio para que no podamos usarlo, pero sin embargo seguirá existiendo en nuestra base de datos
<b>Salida</b>	En esta sección se podrá visualizar los datos de manera general o ya sea al buscar por nombre del laboratorio. Adicionalmente lo podemos eliminar o modificar.

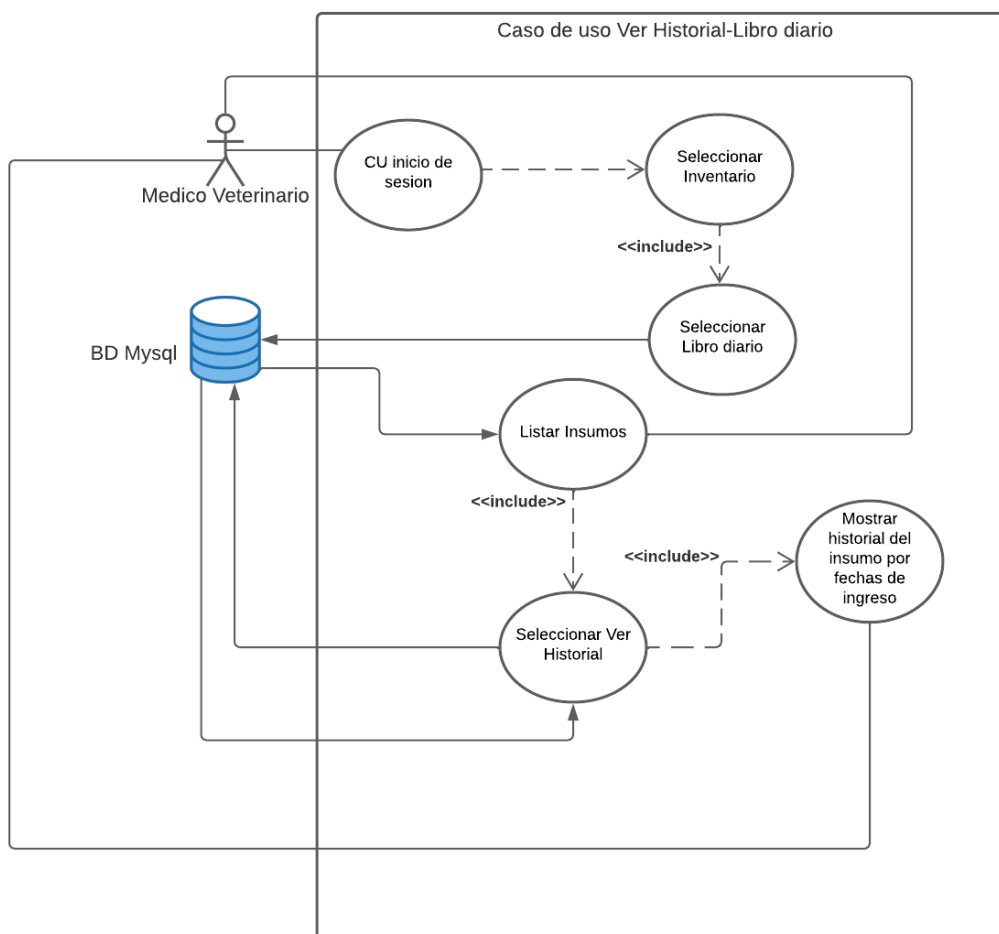


Fig 14. Diagrama CU: Historial del insumo en el libro diario

Tabla 17.Requerimiento Funcional: Historial del insumo en el libro diario

Desarrollo: Historial del insumo en el libro diario	
<b>Precondición</b>	Se visualiza el historial del insumo en el libro diario
<b>Entrada</b>	Listar Insumos
<b>Proceso</b>	Permite listar el historial de un insumo en específico dentro del libro diario en base a las fechas de ingreso, a través de una búsqueda previa en la BD MySQL.
<b>Salida</b>	La información ingresados se visualizará por la fecha en

la que se ingreso

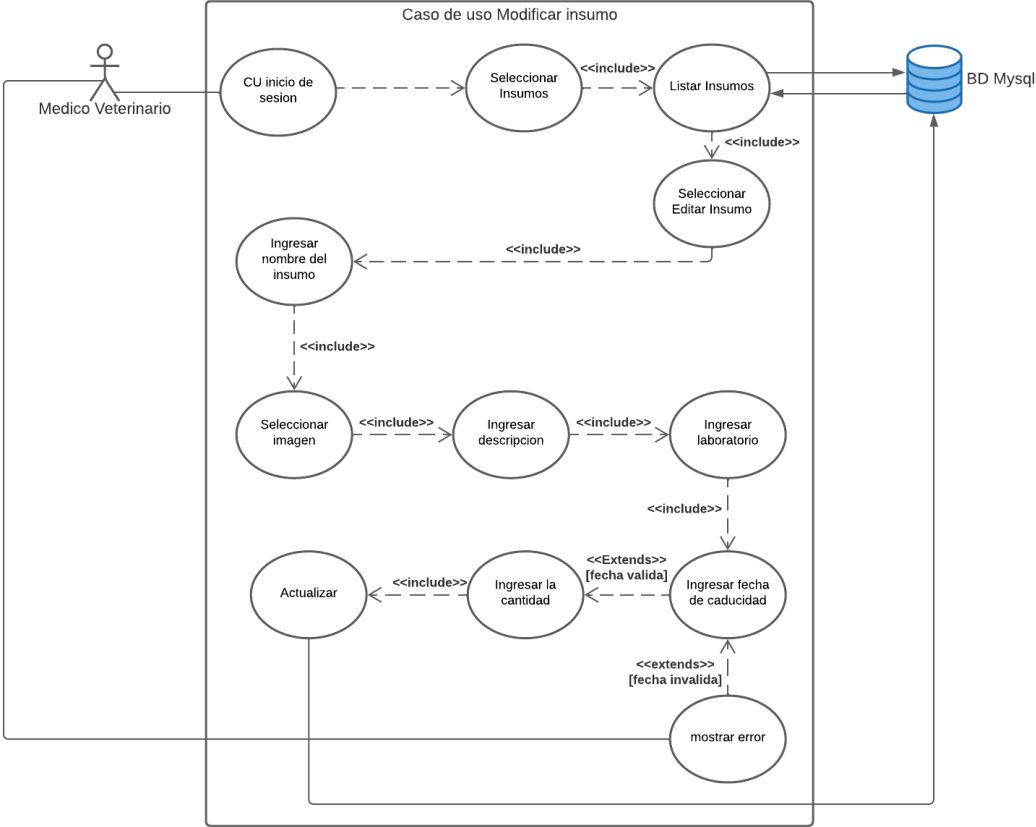


Fig 15. Diagrama CU: Modificar insumo

Tabla 18.Requerimiento Funcional: Modificar insumo

Desarrollo: Modificar insumo	
<b>Precondición</b>	Si existe un mal ingreso de insumos podemos modificar insumo
<b>Entrada</b>	Listar Insumos
<b>Proceso</b>	Permite editar los campos de un insumo para

	posteriormente guárdalos en una BD MySQL.
<b>Salida</b>	Se mandará un error al no modificar de manera adecuada la fecha

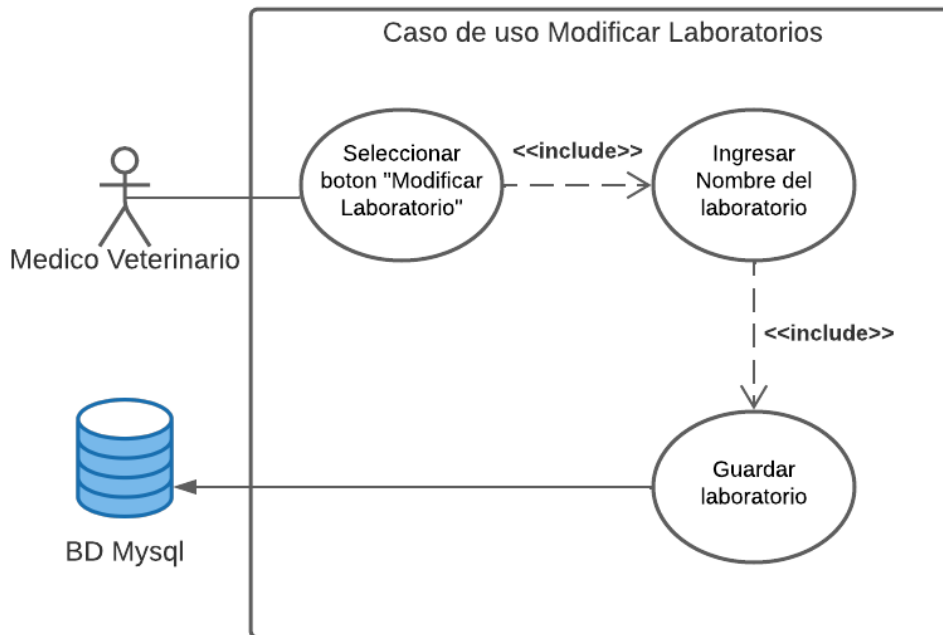


Fig 16. Diagrama CU: Modificar Laboratorio

Tabla 19. Requerimiento Funcional: Modificar laboratorio

<b>Desarrollo: Modificar laboratorio</b>	
<b>Precondición</b>	Si existe un mal ingreso en los laboratorios el médico puede modificar
<b>Entrada</b>	Listar laboratorios
<b>Proceso</b>	Permite editar los campos de un laboratorio para posteriormente guárdalos en una BD MySQL.
<b>Salida</b>	Se podrá visualizar en nuestro listado de laboratorios los

	cambios realizados.
--	---------------------

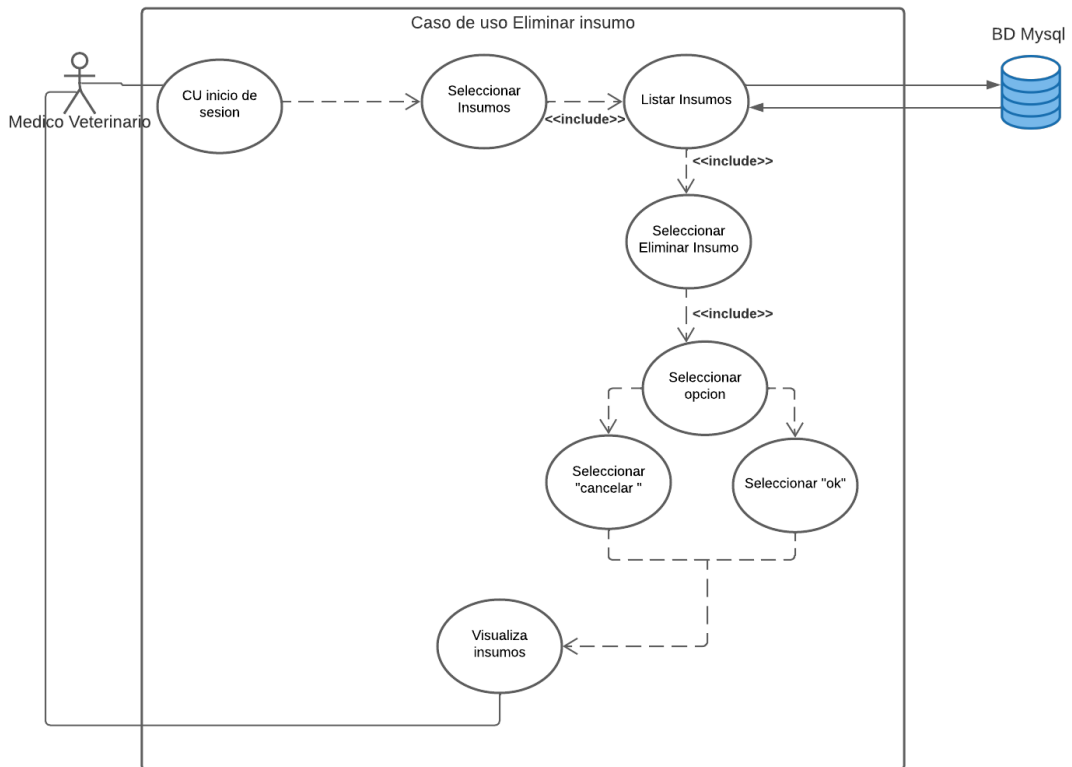


Fig 17. Diagrama CU: Eliminar Insumo

Tabla 20. Requerimiento Funcional: Eliminar insumo

Desarrollo: Eliminar Insumo	
<b>Precondición</b>	Si existe un mal ingreso en el insumo el médico puede eliminarlo
<b>Entrada</b>	Los insumos se eliminan seleccionando listado de los insumos
<b>Proceso</b>	Permite eliminar insumos para posteriormente pasarlos a un estado negativo en la visibilidad dentro de la BD MySQL.

<b>Salida</b>	Se visualizará ya en nuestra lista de insumos con los cambios ya realizados
---------------	---

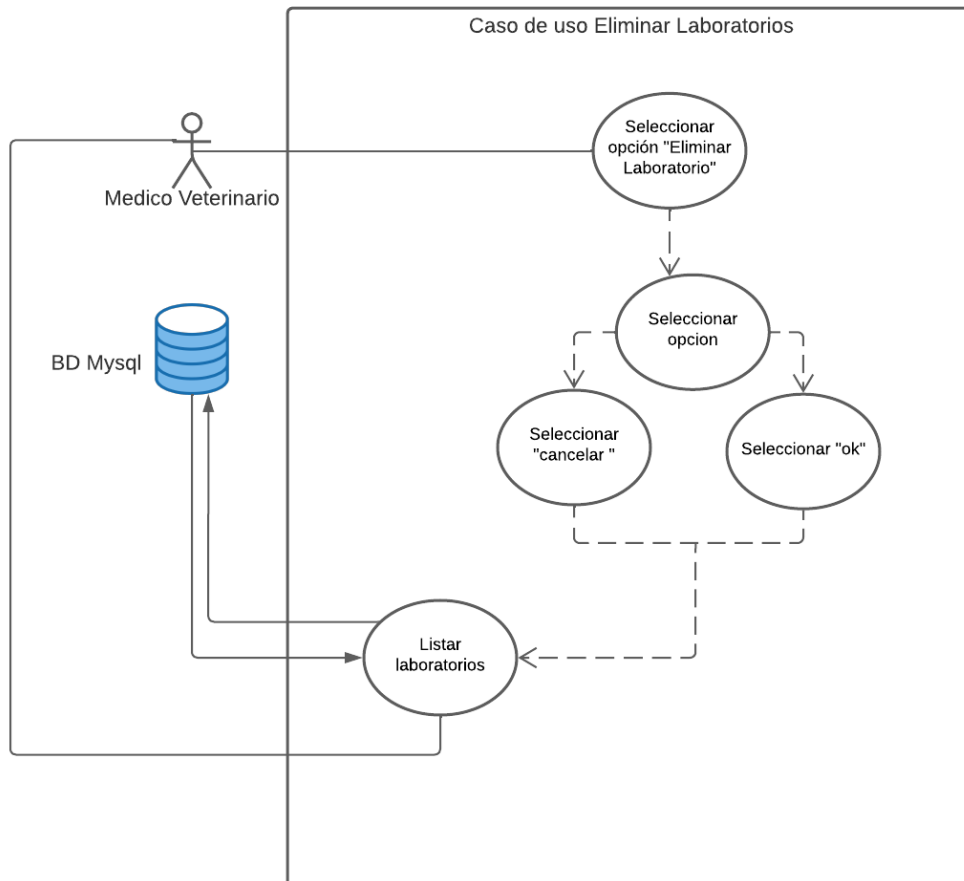


Fig 18. Diagrama CU: Eliminar Laboratorio

Tabla 21.Requerimiento Funcional: Eliminar laboratorios

<b>Desarrollo: Eliminar laboratorio</b>	
<b>Precondición</b>	Si existe un mal ingreso en el laboratorio el médico puede eliminarlo
<b>Entrada</b>	El laboratorio se elimina seleccionando listado del laboratorio.

<b>Proceso</b>	Permite eliminar el laboratorio para posteriormente pasarlos a un estado negativo en la visibilidad dentro de la BD Mysql.
<b>Salida</b>	Se visualizará ya en nuestra lista de laboratorio con los cambios ya realizados

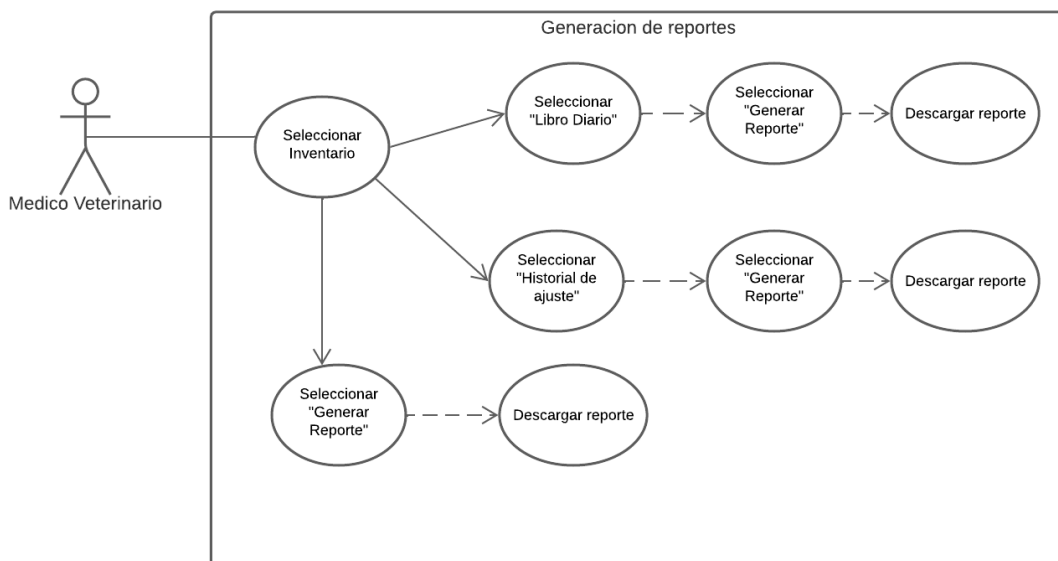


Fig 19. Diagrama CU: Generar Reportes

Tabla 22. Requerimiento Funcional: Generar Reportes

<b>Desarrollo: Generar Reportes</b>	
<b>Precondición</b>	Los médicos podrán generar repostes de tipo csv.
<b>Entrada</b>	Listados de inventario, historial de ajuste y libro diario
<b>Proceso</b>	Permite generar reportes en formato csv sobre los insumos que se encuentran listados en los módulos de: inventario, historial de ajuste y libro diario.

<b>Salida</b>	Se visualiza los datos en un documento csv.
---------------	---

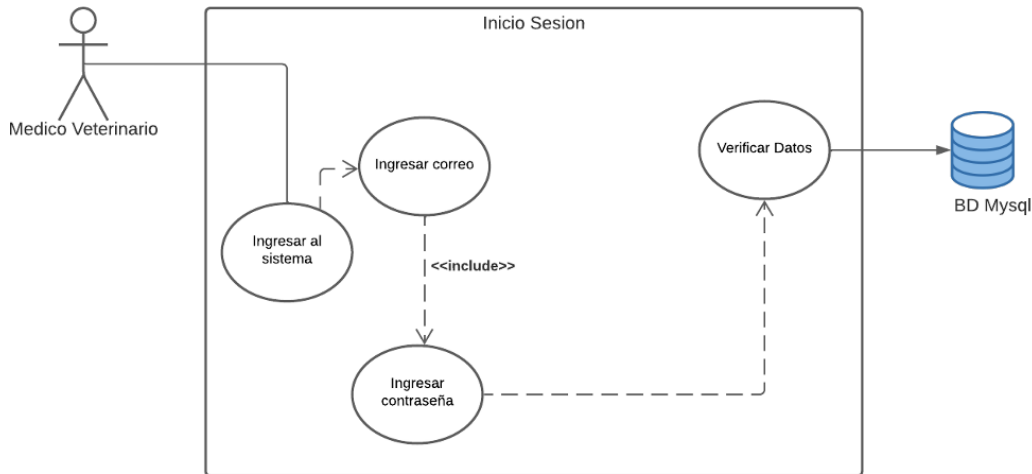


Fig 20. Diagrama CU: Actualización del Inicio de Sesión

Tabla 23. Requerimiento Funcional: Actualización del Inicio de Sesión

<b>Desarrollo: Actualización del Inicio de Sesión</b>	
<b>Precondición</b>	Llenar las credenciales para ingreso del sistema.
<b>Entrada</b>	Correo, contraseña
<b>Proceso</b>	Se actualizo el inicio de sesión para que se pueda ingresar mediante tokens.
<b>Salida</b>	En caso de no ingresar las credenciales correctas se mandará un error

## 5.4 Diseño de la aplicación multiplataforma



### **5.4.1. Diagrama entidad relación**

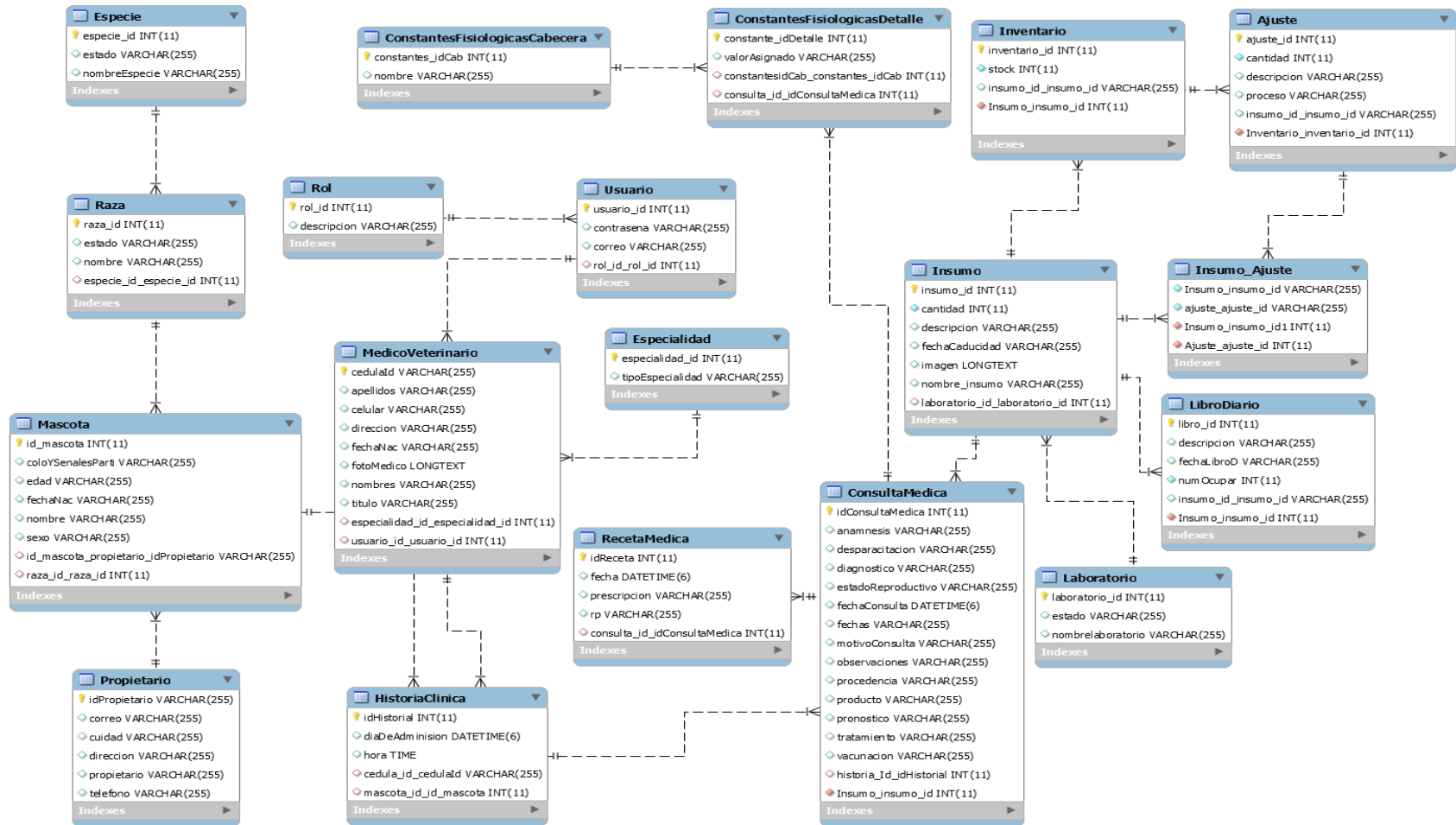


Fig 21. Diagrama Entidad Relación

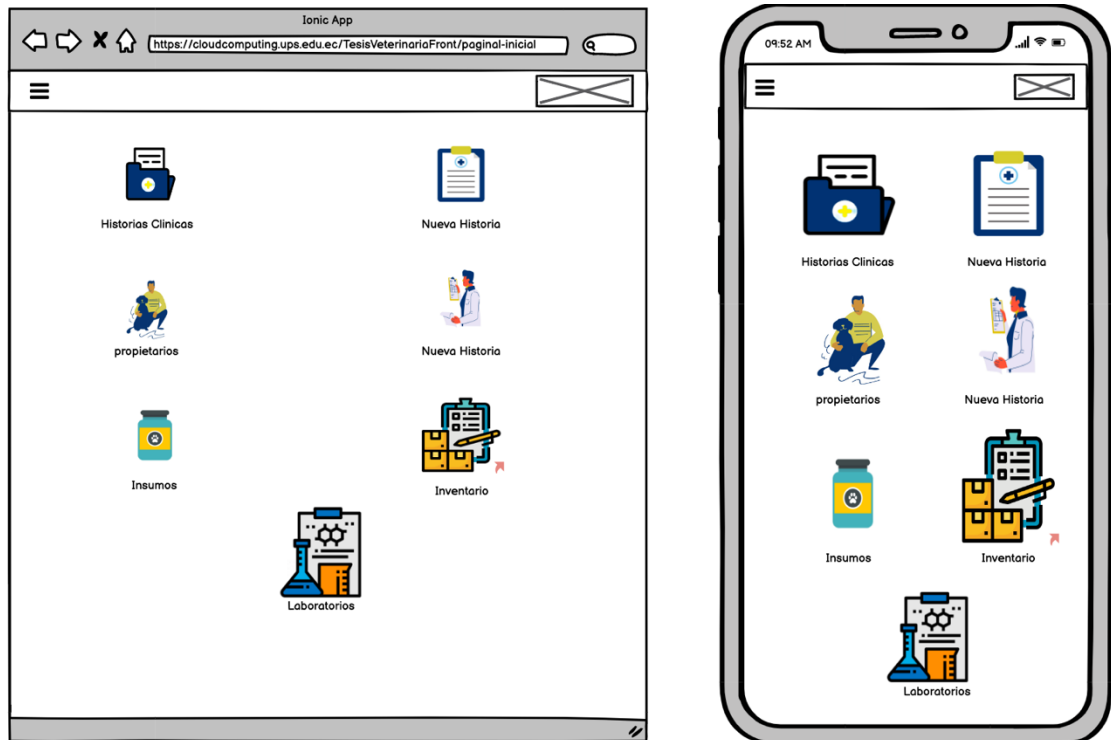
La Figura 21 presenta el diagrama E-R, donde se representa la estructuración de la base de datos junto con sus respectivas relaciones entre las diferentes entidades de la aplicación.

#### 5.4.2. Interfaces de usuario

Esta sección presenta los prototipos del diseño de la interfaz de usuario, los cuales fueron desarrollados en Balsamiq Wireframes Cloud.

#### Ventana Web y Móvil

En la figura 22 se puede observar el prototipo de la interfaz página inicial, en el cual se presentan todos los módulos disponibles de la aplicación.



*Fig 22. Interfaz prototipo página inicial*

En la figura 23 se muestra el prototipo de la interfaz listar insumos, en el cual se presentan todos los insumos disponibles con los que trabaja la veterinaria.

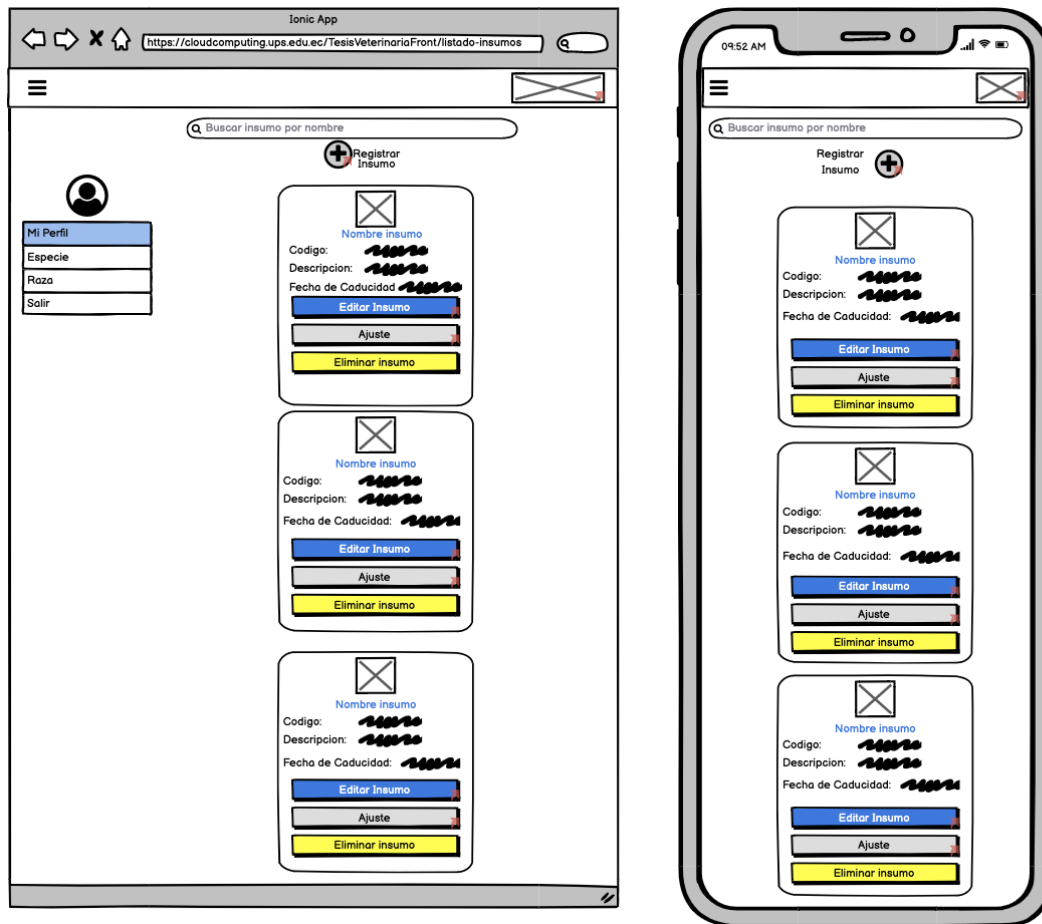


Fig 23. Interfaz prototipo Listar Insumos

En la figura 24 se puede observar el prototipo de la interfaz Registrar insumo, en la cual se presentan todos los campos requeridos para ingresar un insumo.



Fig 24. Interfaz prototipo Registrar Insumo

En la figura 25 se muestra el prototipo de la interfaz Modificar insumo, en la cual se presentan todos los campos modificables del insumo.

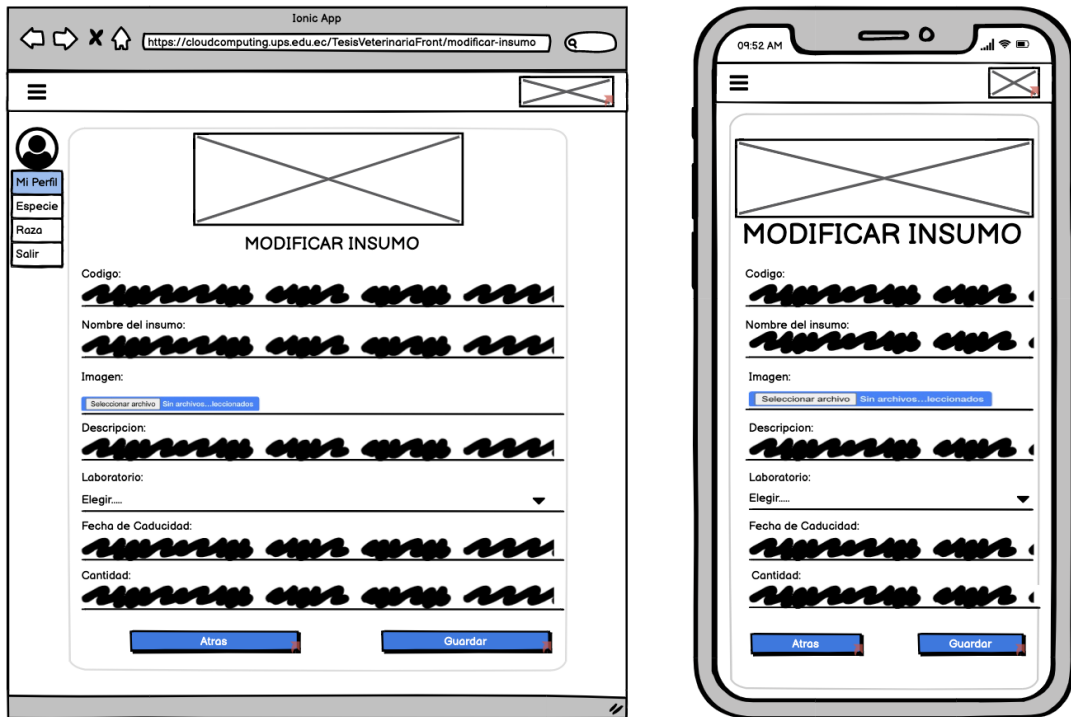


Fig 25. Interfaz prototipo Modificar Insumo

En la figura 26 se puede observar el prototipo de la interfaz listar laboratorios, en el cual se presentan todos los laboratorios disponibles con los que trabaja la veterinaria.

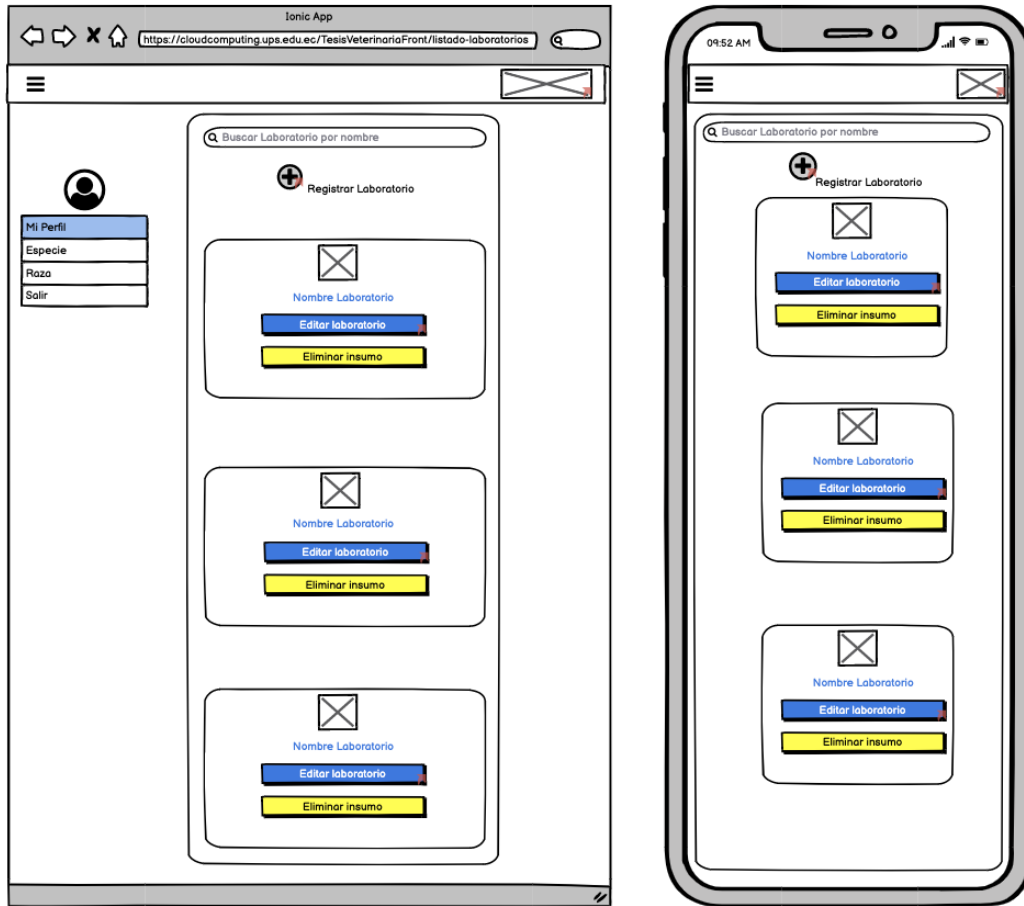
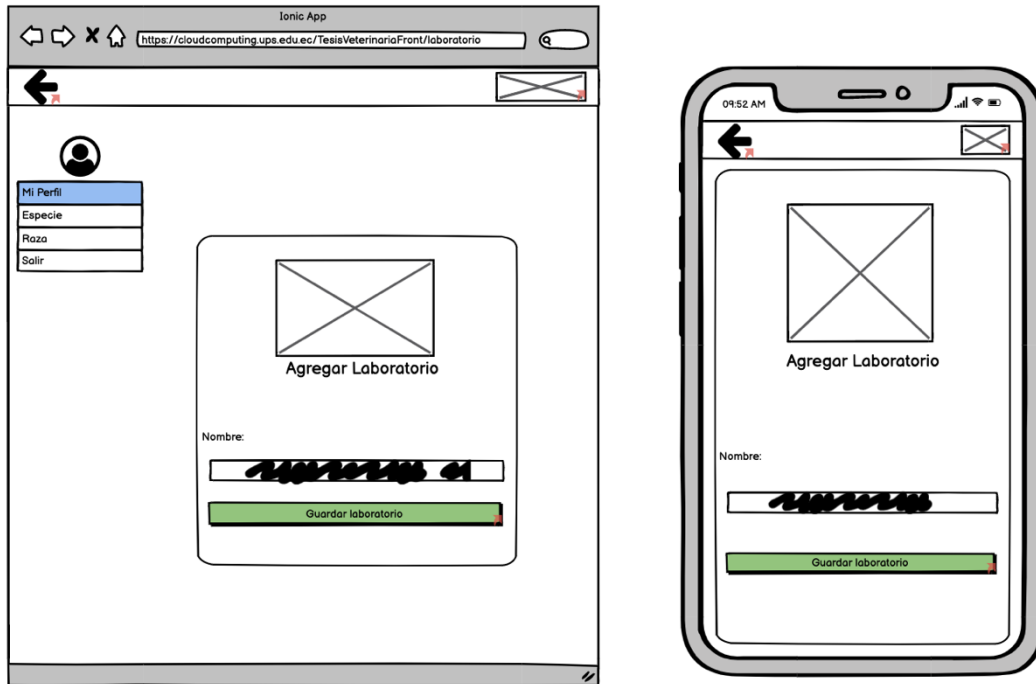


Fig 26. Interfaz prototipo Listar Laboratorios

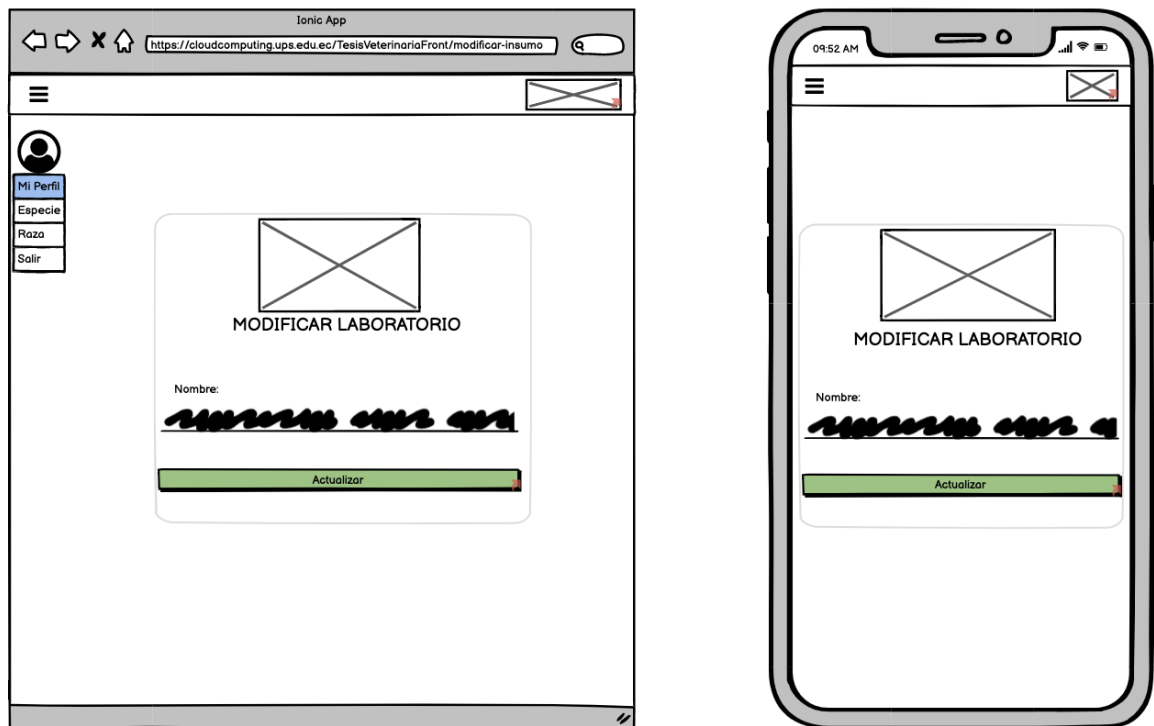
En la figura 27 se muestra el prototipo de la interfaz Registrar laboratorio, en la cual se presentan todos los campos requeridos para ingresar un laboratorio.



*Fig 27. Interfaz prototipo Registrar Laboratorio*

En la figura 28 se muestra el prototipo de la interfaz Modificar laboratorio, en la cual se presentan todos los campos modificables del laboratorio.





*Fig 28. Interfaz prototipo Modificar Laboratorio*

En la figura 29 se puede observar el prototipo de la interfaz Registrar libro diario, en la cual se presentan todos los campos requeridos para ingresar un insumo dentro del libro diario.

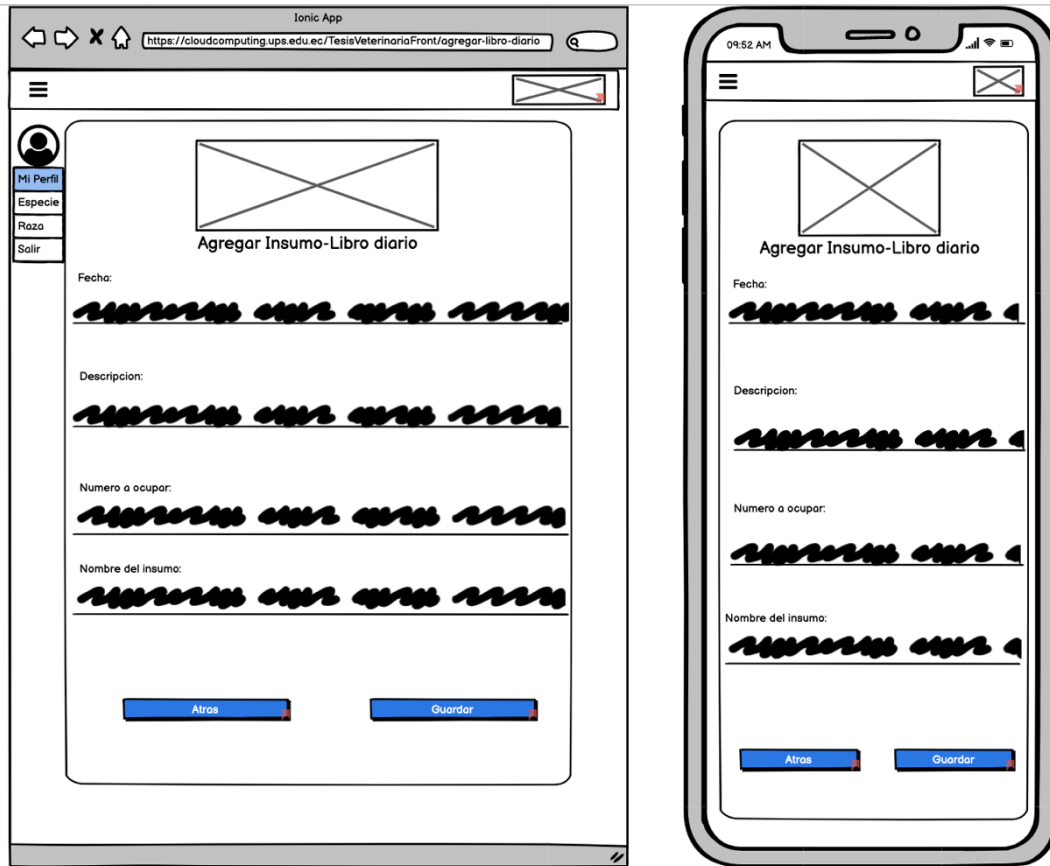


Fig 29. Interfaz prototipo Registrar libro diario

En la figura 30 se muestra el prototipo de la interfaz historial libro diario, en la cual se presenta el historial dicho insumo dentro del libro diario, es decir desde la primera vez que fue ingresado en el libro, hasta la última vez que fue ingresado.



Fig 30. Interfaz prototipo Historial libro diario

En la figura 31 se muestra el prototipo de la interfaz Ajuste de Inventario, en la cual se presentan los procesos con los que cuenta el ajuste, en este caso 2(Aumentar, Reducir).

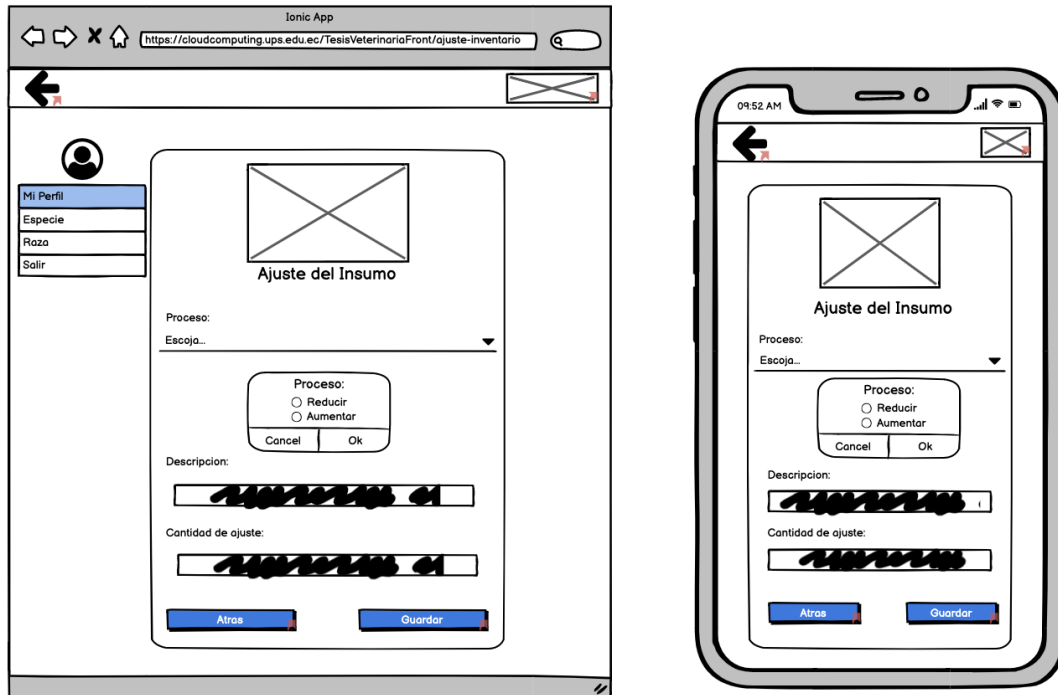


Fig 31. Interfaz prototipo Ajuste de Inventario

En la figura 32 se muestra el prototipo de la interfaz Inventario, en la cual se presentan los insumos con los que cuenta el inventario.

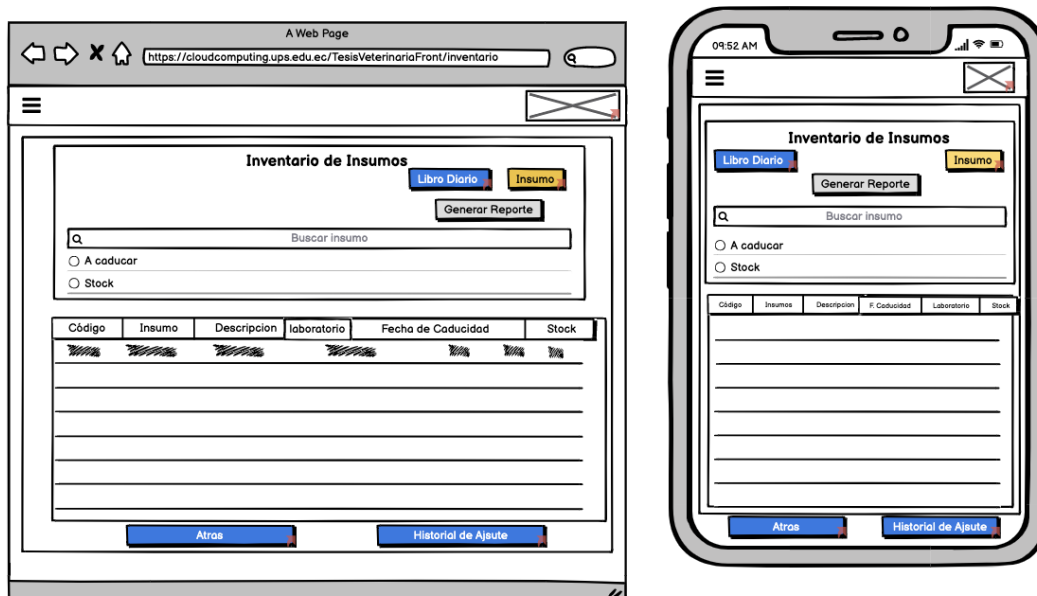


Fig 32. Interfaz prototipo Inventario

En la figura 33 se muestra el prototipo de la interfaz Libro Diario, en la cual se presentan los insumos con los que cuenta el libro diario.

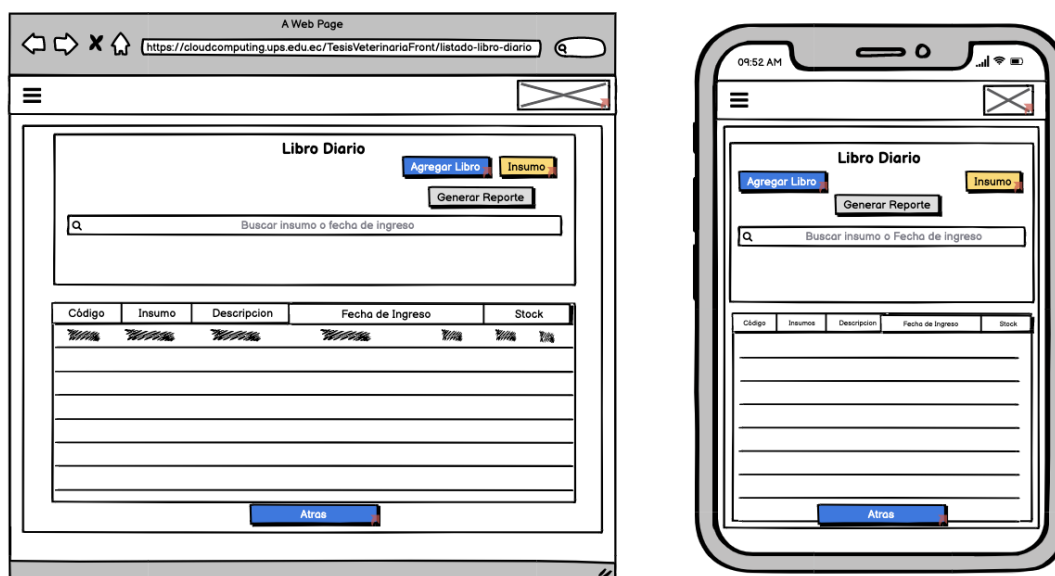


Fig 33. Interfaz prototipo Libro Diario

En la figura 34 se muestra el prototipo de la interfaz Historial de Ajuste, en la cual se presenta el historial de ajuste de inventario.

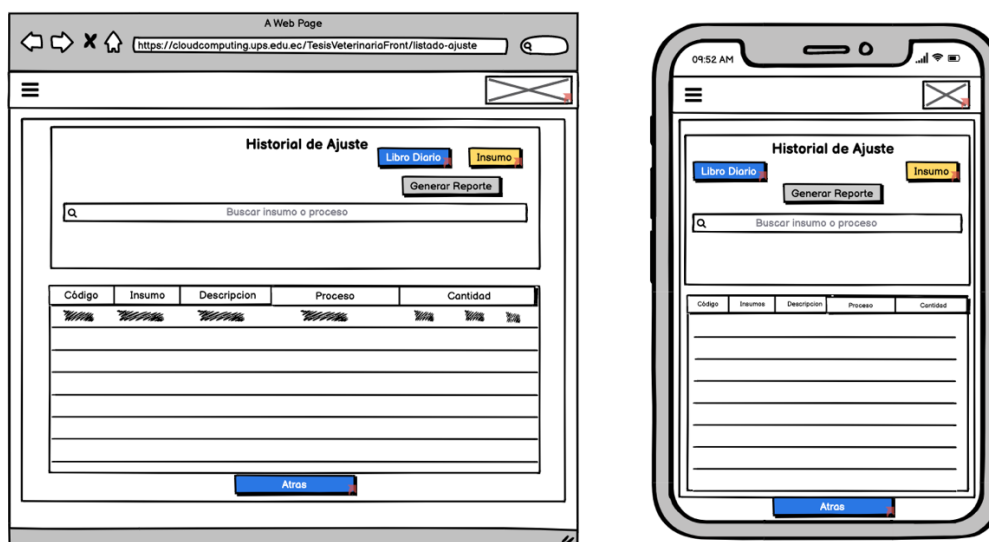
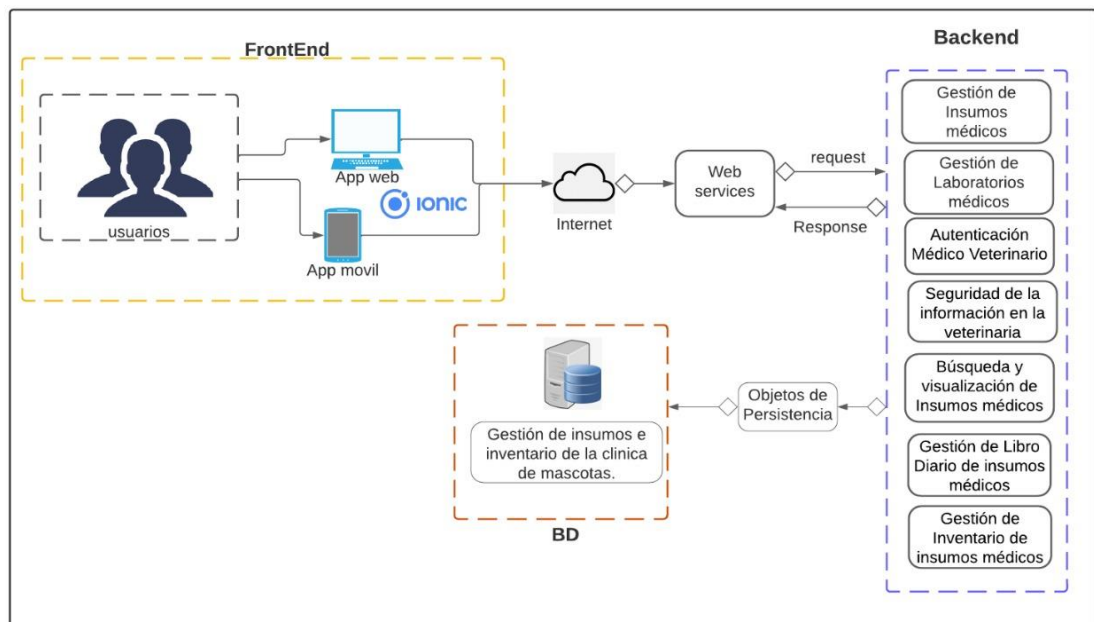


Fig 34. Interfaz prototipo Historial de Ajuste

**Tabla 24. Revisión de prototipos**

Revisado por:	Firma
Dra. Carmen Angélica Yanzaguano Romero.	

### 5.4.3. Arquitectura de la aplicación



*Fig 35. Planteamiento general app multiplataforma*

El planteamiento general de la aplicación multiplataforma perteneciente a la Fig. 35 presenta la interacción entre los usuarios y la aplicación. La app será desplegada para ser usada en plataformas web como móviles. El aplicativo obtendrá datos del usuario con el fin de mostrar sus salidas de diferentes maneras de exposición. En cuanto al backend, será el encargado de administrar los módulos presentados en el planteamiento y dar sus respectivas soluciones, las cuales serán mostradas según las necesidades del usuario. Finalmente está la base de datos, encargada de administrar registros y consultas en los módulos, proporcionando así la información dependiendo de las peticiones realizadas por el usuario.

La arquitectura de aplicaciones multiplataforma se compone de tres componentes básicos:

- Servidor Wildfly 24.0.1
- Base de datos MySQL 5.7.34
- Ionic Framework (Ionic 5)
- Angular 14.17.0
- Node js 6.14.13

La lógica la maneja el backend del servidor de aplicaciones, la cual cuenta con web services, los cuales se conectan con la base de datos a través de la persistencia de datos. Finalmente, el uso de Ionic junto con Angular y Node js permitieron la creación del Frontend.

La realización del proyecto está constituida por cuatro fases:

- La **primera** es tener un conocimiento acerca de los insumos médicos que son adquiridos y posteriormente utilizados o recetados a las mascotas por parte de la clínica veterinaria. Durante la fase de levantamiento serán incluidos diagramas de procesos a fin de reconocer la problemática presentada al momento de receptar información y cuidado de los insumos. Asimismo, investigar y profundizar en patrones de buenas prácticas para la gestión de insumos.
- La **segunda** fase consiste en agendar un primer encuentro con los stakeholders para el levantamiento de requisitos funcionales y no funcionales. En esta parte se debe detallar: obstáculos, la necesidad y sus elementos más importantes, los cuales encaminan a un buen funcionamiento del sistema. Una vez realizado el boceto inicial juntamente con el médico se procede a estudiar los requisitos previos con el objetivo de diseñar un prototipo el cual presentará el funcionamiento del sistema.

Tan pronto como se haya concluido el análisis se debe estructurar el documento final de esta fase con los requisitos establecidos y previamente refinados, en donde se detallará correctamente cada mejora, proceso y necesidades acordadas.

- En la **tercera** fase se realizará el backend, el cual implementará Web services abarcando sus componentes referenciados a continuación y expuestos en la Fig. 19:
  - **Gestión Insumos médicos:** Documento en línea donde se recopila información acerca de un insumo médico. Este documento permite: registrar, actualizar, eliminar y listar los insumos con los cuales trabaja la veterinaria para el tratamiento de las mascotas. Este módulo está compuesto por los siguientes parámetros: código, nombre, descripción, imagen, laboratorio, fecha de caducidad, cantidad.
  - **Autenticación Médico Veterinario:** Este componente define el acceso del veterinario a la app multiplataforma determinado, por el nombre de usuario y contraseña que se encuentren ingresados en la BD.
  - **Seguridad de la información en la veterinaria:** Este módulo permitirá una navegación segura sin pérdida de datos y visibilidad autorizada de los mismos entre los diferentes módulos de la aplicación.
  - **Búsqueda y visualización de Insumos médicos:** Permite a los veterinarios buscar y obtener rápidamente insumos médicos para las mascotas.
  - **Gestión de Inventario de insumos médicos:** Este módulo posibilita a los veterinarios llevar un control preciso de cada insumo médico que dispone la veterinaria. Este módulo este compuesto por los siguientes parámetros: código, nombre, descripción, fabricante, imagen, unidad de medida, fecha de caducidad, cantidad, stock, alarma.
  
- La **cuarta** fase es diseñar e implementar una interfaz de usuario utilizando Ionic como framework y tecnologías como: Angular, JavaScript, HTML y CSS. La ventaja de trabajar con ionic es que al ser multiplataforma se acopla



eficazmente a otros dispositivos, logrando así que la aplicación sea progresiva.

## 5.5 Metodología de desarrollo

Esta sección presenta la metodología SCRUM la cual será usada durante el desarrollo de la aplicación y se compone de 3 Sprints.

### 5.1.1. Descripción personal scrum

La tabla 25, propone 3 roles principales, los cuales fueron asignados para el desarrollo del proyecto. Estos roles son los siguientes:

*Tabla 25. Asignación de roles del método SCRUM*

<b>Rol</b>	<b>Actores</b>
<b>Scrum Master</b>	Ing. Gabriel Alejandro León, Ph.D
<b>Product Owner</b>	Dr. Carmen Angélica Yanzaguano Romero
<b>Equipo de desarrollo</b>	Est. Kevin Adrián Godoy Mendía Est. William Eduardo Chabla Palaguachi

### 5.1.2. Conformación de los Sprints

En la estructura del Sprint es fundamental detallar la lista de actividades que se realizarán en el proyecto, para posteriormente presentar detalladamente la lista de actividades a realizar en cada objetivo específico.

- **Lista de Actividades**

**OE1.** Estudiar y conocer los procesos dentro de la clínica veterinaria y las herramientas tecnológicas para la implementación del módulo de gestión del inventario y seguridad de la aplicación.

*Tabla 26. Actividades del Objetivo Específico 1*

No.	Actividad
1	Determinar y adentrarse en las tecnologías más comunes para la implementación de sistemas multiplataforma (JavaScript, CSS, Ionic, PWA, HTML)
2	Bocetos en el campo de la medicina veterinaria en: área de contabilidad, requerimientos del mercado y demanda de herramientas tecnológicas.

**OE2.** Levantar los procesos relacionados a la gestión de insumos de la clínica veterinaria “Polivet”.

*Tabla 27. Actividades del Objetivo Específico 2*

No.	Actividad
1	Determinar procesos de administración de la veterinaria con relación a la gestión de insumos.
2	Realizar diseño y modelo de cada proceso de administración en la veterinaria relacionados con la gestión de insumos.

**OE3.** Analizar y diseñar los requerimientos funcionales y no funcionales de la aplicación multiplataforma mediante la recolección, validación y documentación de los mismos.

**Tabla 28. Actividades del Objetivo Específico 3**

No.	Actividad
1	Junta con los stakeholders.
2	Recopilación de requisitos.
3	Estudio de requisitos con los stakeholders.
4	Diseño de modelo ER (Entidad Relación).
5	Elaboración del prototipo multiplataforma para la aplicación.
6	Documentación de requisitos.

**OE4.** Desarrollar un módulo seguro de compra de insumos y manejo de inventario que cumpla con los requisitos recolectados y evidencie los procesos de forma digital.

**Tabla 29. Actividades del Objetivo Específico 4**

No.	Actividad
1	Elaboración de los servicios web.
2	Diseño de backend e implementación de los servicios web.
3	Diseño de frontend.
4	Implementación del backend y frontend.
5	Documentación del desarrollo de la aplicación multiplataforma y correcto funcionamiento de la misma.

**OE5.** Realizar las pruebas funcionales y despliegue de la aplicación con la actualización del sistema.

*Tabla 30. Actividades del Objetivo Específico 5*

No.	Actividad
1	Despliegue de la aplicación en el servidor del grupo de investigación GiHP4C.
2	Ejecución del plan de pruebas.
3	Recolección de resultados del plan de pruebas.

Con base en los objetivos y actividades planteadas se han dividido los objetivos específicos en 3 sprints. Al final de cada sprint se procederá a revisarlos a fin de que se pueda verificar que se hayan cumplido todas las actividades. Las actividades serán desarrolladas en los sprints presentados a continuación:

*Tabla 31. Distribución de Sprint 1*

**SPRINT 1**

<b>OE.1</b>
<b>ACT 1.</b> Determinar y adentrarse en las tecnologías más comunes para la implementación de sistemas multiplataforma (JavaScript, CSS, Ionic, PWA, HTML)

<b>ACT 2.</b> Bocetos en el campo de la medicina veterinaria en: área de contabilidad, requerimientos del mercado y demanda de herramientas tecnológicas.
<b>OE.2</b>
<b>ACT 1.</b> Determinar procesos de administración de la veterinaria con relación a la gestión de insumos.
<b>ACT 2.</b> Realizar diseño y modelo de cada proceso de administración en la veterinaria relacionados con la gestión de insumos.

*Tabla 32. Distribución de Sprint 2*

### SPRINT 2

<b>OE.3</b>
<b>ACT 1.</b> Junta con los stakeholders.
<b>ACT 2.</b> Recopilación de requisitos.
<b>ACT 3.</b> Estudio de requisitos con los stakeholders.
<b>ACT 4.</b> Diseño de modelo ER (Entidad Relación).
<b>ACT 5.</b> Elaboración del prototipo multiplataforma para la aplicación.
<b>OE. 4</b>
<b>ACT 1.</b> Elaboración de los servicios web.
<b>ACT 2.</b> Diseño de backend e implementación de los servicios web.
<b>ACT 3.</b> Diseño de frontend.
<b>ACT 4.</b> Implementación del backend y frontend.

*Tabla 33. Distribución de Sprint 3*

**SPRINT 3**

<b>OE. 5</b>
<b>ACT 1.</b> Despliegue de la aplicación en el servidor del grupo de investigación GiHP4C.
<b>ACT 2.</b> Ejecución del plan de pruebas.
<b>ACT 3.</b> Recolección de resultados del plan de pruebas.

Una vez ya con el conocimiento de las herramientas y la tecnología scrum en la que nos basamos, ahora vamos a ver sobre las metodologías como son los requerimientos funcionales y los no funcionales.

## **6. Resultados**

### **6.1 Codificación web services**

A continuación, observaremos el desarrollo de los webs services las cuales están en la parte del backend:

*Tabla 34. Presentación de los respectivos sprint desarrollados*

<b>Descripción</b>	<b>Explicación</b>	
<b>Inicio</b>	Para poder ingresar al sistema se necesita un correo y una contraseña. Una vez que se compruebe que los datos ingresados se encuentren en la base de datos; permitirá el acceso.	
	<b>Datos de entrada</b>	correo, contraseña
	<b>Datos de salida</b>	Estado del sesión creada o no creada
<b>registrar- Laboratorio</b>	Este Web services nos permite registrar el laboratorio según la información que se encuentra almacenada en el formulario.	
	<b>Datos de entrada</b>	Nombre del laboratorio
	<b>Datos de salida</b>	Estado del laboratorio creado o no creado
<b>editar- Laboratorio</b>	Esta Web services nos ayuda a editar los datos del laboratorio, según la información del nombre que se le asigno.	
	<b>Datos de entrada</b>	Id del laboratorio, nombre del laboratorio
	<b>Datos de salida</b>	Estado del laboratorio actualizado
<b>eliminar- Laboratorio</b>	Para poder eliminar los datos del laboratorio, por medio de esta Web services lo realizamos por el código del laboratorio.	
	<b>Datos de entrada</b>	Id del laboratorio
	<b>Datos de salida</b>	Mensaje error o estado eliminado
<b>obtener- Laboratorio</b>	Esta Web services permite listar los datos del laboratorio que se encuentran almacenados en el respectivo registro.	
	<b>Datos de entrada</b>	No recibe parámetros de entrada
	<b>Datos de salida</b>	Estado de la salida del laboratorio, listado de laboratorios
<b>registrar- Insumo</b>	Este Web services nos permite registrar el Insumo según la información que se encuentra almacenada en el formulario.	

	<b>Datos de entrada</b>	Id del insumo, nombre del insumo, imagen, descripción, id del laboratorio, fecha de caducidad, cantidad
	<b>Datos de salida</b>	Estado de salida será creado.
<b>borrar- Insumo</b>	Para poder eliminar los datos del laboratorio, por medio de esta Web services lo realizamos por el código del laboratorio.	
	<b>Datos de entrada</b>	Id del insumo
	<b>Datos de salida</b>	Estado de salida eliminado o mensaje de error
<b>listar- Insumo</b>	Esta Web services permite listar los datos del insumo que se encuentran almacenados en el respectivo registro.	
	<b>Datos de entrada</b>	No recibe ningún parámetro de entrada
	<b>Datos de salida</b>	Recibe la lista de los insumos, en formato JSON o mensaje de error.
<b>editar- Insumo</b>	Esta Web services nos ayuda a editar los datos del insumo, según la información del nombre que se le asigno.	
	<b>Datos de entrada</b>	Id del insumo, nombre del insumo, imagen, descripción, fecha de caducidad, cantidad, id del laboratorio.
	<b>Datos de salida</b>	Estado actualizado.
<b>registrar- LibroDiario</b>	Este Web services nos permite registrar el libro diario según la información que se encuentra almacenada en el formulario.	
	<b>Datos de entrada</b>	Fecha del ingreso, descripción, numero a ocupar, id del insumo
	<b>Datos de salida</b>	Estado creado
<b>borrar- Libro</b>	Para poder eliminar los datos del libro diario, por medio de esta Web services.	
	<b>Datos de entrada</b>	Id de libro
	<b>Datos de salida</b>	Estado eliminado o mensaje de erro



<b>listar- Libro</b>	Esta Web services permite listar los datos del libro diario que se encuentran almacenados en el respectivo registro.	
	<b>Datos de entrada</b>	No tiene entrada
	<b>Datos de salida</b>	Retorna la lista del libro en formato JSON
<b>editar- Libro</b>	Esta Web services nos ayuda a editar los datos del libro diario, según la información del nombre que se le asigno.	
	<b>Datos de entrada</b>	Id del libro, fecha de ingreso, descripción y numero a ocupar
	<b>Datos de salida</b>	Estado actualizado.
<b>listarlibrox- insumo_id</b>	Permite visualizar el historial del libro diario por medio del código del insumo que se encuentran almacenados en el formulario del registro respectivo en formato JSON	
	<b>Datos de entrada</b>	Id del insumo
	<b>Datos de salida</b>	Retorna el listado del libro diario con el historial del insumo específico en formato JSON
<b>registrar- Inventario</b>	Este Web services nos permite registrar el libro diario según la información que se encuentra almacenada en el formulario.	
	<b>Datos de entrada</b>	Id del inventario, stock, id del insumo
	<b>Datos de salida</b>	Estado creado.
<b>listar- Inventario</b>	Esta Web services permite listar los datos del inventario que se encuentran almacenados en el respectivo registro en formato JSON	
	<b>Datos de entrada</b>	No recibe parámetros
	<b>Datos de salida</b>	Recibe la lista del inventario o caso contrario se muestra un mensaje de error.

<b>reducir- Stock</b>	Esta Web services permite reducir el stock es decir la cantidad total que se encuentra en el inventario según los insumos que se ocupan.	
	<b>Datos de entrada</b>	Id del insumo, id del libro diario
	<b>Datos de salida</b>	Se muestra un mensaje de ok o error.
<b>listar- fechaInv</b>	Esta Web services permite listar los datos del inventario por fecha de caducidad que se encuentran almacenados en el respectivo registro del formulario.	
	<b>Datos de entrada</b>	No recibe parámetros.
	<b>Datos de salida</b>	Retorna la lista del inventario por orden de fecha en formato JSON o caso contrario se muestra un mensaje de error.
<b>listar- StockInv</b>	Esta Web services permite listar los datos del inventario por stock mínimo que se encuentran almacenados en el respectivo registro del formulario.	
	<b>Datos de entrada</b>	No recibe parámetros
	<b>Datos de salida</b>	Se lista en formato JSON o error.
<b>registrar- Ajuste</b>	Este Web services nos permite registrar el ajuste según la información que se encuentra almacenada en el formulario.	
	<b>Datos de entrada</b>	Id del insumo, proceso, descripción, cantidad.
	<b>Datos de salida</b>	Estado creado o error
<b>listar- Ajuste</b>	Esta Web services permite listar los datos del ajuste que se encuentran almacenados en el respectivo registro una vez realizado el ajuste.	
	<b>Datos de entrada</b>	No recibe parámetros
	<b>Datos de salida</b>	Se lista en formato JSON

<b>borrar- Inventario</b>	Para poder eliminar los datos del inventario, por medio de esta Web services se lo realizará al eliminar los datos primeramente en el insumo.	
	<b>Datos de entrada</b>	Id de inventario
	<b>Datos de salida</b>	Estado eliminado o error
<b>listarLibro- DiariTodo</b>	En la presente Web services podremos visualizar todos los datos del libro diario que se encuentran almacenados en el respectivo registro.	
	<b>Datos de entrada</b>	No recibe parámetros
	<b>Datos de salida</b>	Se lista todo lo del libro diario en formato JSON

## 6.2 Código web services

Para tener un conocimiento de la realización de los webs services se mostrará los principales códigos que se utilizó para la realización de este sistema. Los métodos que se utilizaron fueron:

- Post el cual nos permite tener una conexión con el servidor para que de esta manera cumplir con las peticiones del cliente (Amat, 2018).
- Path se obtiene los recursos en donde se encuentran los datos para la ejecución de ese código para que de esa manera se logre acceder por medio de ese path a la página indicada (Saravia, 2001).
- FormParam cómo se puede ver en las figuras 35-37 son etiquetas que a los cuales se les colocan un nombre para que de esta manera los servicios se consuman (Montes-Cumbrera, 2015). Estas etiquetas toman el nombre de acuerdo con los atributos de la clase en la parte de nuestro backend.

Posteriormente lo que se puede apreciar es los EJB (Enterprise Java Beans) que nos ayuda a invocar al método para acceder a la información almacenada que se encuentra guardada en una variable; por lo general se utilizó son los bean de sesión (Wetherbee, 2018).

Los métodos mencionados para nuestra web services; son los que se usó para nuestro registro de insumos, ajuste y libro diario. Ahora se procederá a explicar de manera más detalla en lo que se realizó en cada una de ellas.

- **RegistrarInsumo**

En la figura 36, se puede apreciar el registro insumos; que nos permite ingresar los datos de los insumos a nuestra base de datos para ser utilizadas por el médico a través de la página web. Los atributos de la clase que incluye nuestros insumos son: insumos\_id, nombre\_insumo, imagen, descripcion, laboratorio, fechaCaducidad y cantidad; las cuales son llamados al método para ser consumidos por nuestro servicio. Finalmente, los EJB que se utilizó son los beans de sesión para poder acceder a cada uno de nuestros atributos correctos de nuestra clase y poder llamar al método del registro de nuestros insumos.

```
/**
 * Metodo post registrarInsumo
 */
@POST
@Path("/registrarInsumo")
@Consumes(MediaType.APPLICATION_FORM_URLENCODED)
public Response registrarInsumo(@FormParam("insumo_id") String insumo_id,
                                @FormParam("nombre_insumo") String nombre_insumo,
                                @FormParam("imagen") String foto,
                                @FormParam("descripcion") String descripcion,
                                @FormParam("laboratorio_id") String laboratorio,
                                @FormParam("fechaCaducidad") String fechaCaducidad,
                                @FormParam("cantidad") String cantidad){

    String imagen = foto.replace(" ", "+");
    System.out.println("Insumos-----" + nombre_insumo + ", " + imagen + ", " + descripcion + ", " +
        laboratorio + ", " + fechaCaducidad + ", " + cantidad);
    Laboratorio laboratorios = new Laboratorio();
    laboratorios = ejbLaboratorioFacade.laboratorioByNombre(laboratorio);
    System.out.println(laboratorio.toString());
    Insumo insumo = new Insumo();
    insumo.setInsumo_id(insumo_id);
    insumo.setNombre_insumo(nombre_insumo);
    insumo.setImagen(imagen);
    insumo.setDescripcion(descripcion);
    insumo.setLaboratorio_id(laboratorios);
    insumo.setFechaCaducidad(fechaCaducidad);
    insumo.setCantidad(cantidad);
    ejbInsumoFacade.create(insumo);
    return Response.ok("Ok").build();
}
```

Fig 36. Código de la web services registrarInsumo

- **RegistrarAjuste**

En la figura 37, se puede apreciar el registro de ajuste el cual, nos permite modificar el campo cantidad de los datos, ya registrados con anterioridad; de nuestros insumos en caso de que lo requiera, posibilitando aumentar o reducir el stock del inventario dentro de la base de datos. Los atributos de la clase que incluye el ajuste son: insumos\_id, proceso, descripcion y cantidad; las cuales son llamados al api para ser consumidos por nuestro servicio. Finalmente, los EJB que se utilizaron fueron los beans de sesión para poder acceder a cada uno de nuestros atributos correctos de nuestra clase y poder llamar al método del registro de ajuste.

```
/**
 * Metodo POST registrarAjuste
 */
@POST
@Path("/registrarAjuste")
@Consumes(MediaType.APPLICATION_FORM_URLENCODED)
public Response registrarAjuste(@FormParam("insumo_id") String insumo_id,
                                @FormParam("proceso") String proceso_aju,
                                @FormParam("descripcion") String descripcion_aju,
                                @FormParam("cantidad") int cantidad_aju) {
    System.out.println("Ajuste-----" + insumo_id + ", " + proceso_aju + ", " + descripcion_aju + ", "+ cantidad_aju);
    Insumo insumo = new Insumo();
    insumo = ejbInsumoFacade.InsumoById(insumo_id);
    Ajuste aju = new Ajuste();
    aju.setAjuste_id (null);
    aju.setInsumo_id (insumo);
    aju.setProceso (proceso_aju);
    aju.setDescripcion (descripcion_aju);
    aju.setCantidad (cantidad_aju);
    ejbAjusteFacade.create(aju);
    ejbAjusteFacade.flushEntities();
    if(aju.getProceso().equals("Aumentar")) {
        ejbInventarioFacade.aumentarStockAjuste (insumo.getInsumo_id (), aju.getAjuste_id ());
    }else if(aju.getProceso ().equals("Reducir")){
        ejbInventarioFacade.reducirStockAjuste (insumo.getInsumo_id (), aju.getAjuste_id ());
    }
    return Response.ok("ok" + aju).build();
}
```

Fig 37. Código de la web services registrarAjuste

- **RegistrarLibroDiario**

En la figura 38, se puede apreciar el registro libro diario; que nos permite registrar los insumos consumidos o utilizados en

nuestro día a día esto permitirá cambios y una reducción en el stock en nuestro inventario. Los atributos de la clase que incluye nuestro libro diario son: fechaLibroD, descripcion, numOcupar e insumo\_id; las cuales son llamados al método para ser consumidos por nuestro servicio. Finalmente, los EJB que se utilizó son los beans de sesión para poder acceder a cada uno de nuestros atributos correctos de nuestra clase y poder llamar al método del registro del libro diario.

```

/**
 * Metodo post registrarLibroDiario
 */
@POST
@Path("/registrarLibroDiario")
@Consumes(MediaType.APPLICATION_FORM_URLENCODED)
public Response registrarLibroDiario(
    @FormParam("fechaLibroD") String fechaLibroD,
    @FormParam("descripcion") String descripcion,
    @FormParam("numOcupar") int numOcupar,
    @FormParam("insumo_id") String insumo_id) {
    System.out.println("Libro Diario-----"+ descripcion + ", " + fechaLibroD + ", " + numOcupar);
    Insumo insumo = new Insumo();
    insumo = ejbInsumoFacade.InsumoByNombre(insumo_id);
    LibroDiario ld = new LibroDiario();
    ld.setLibro_id(null);
    ld.setDescripcion(descripcion);
    ld.setFechaLibroD(fechaLibroD);
    ld.setNumOcupar(numOcupar);
    ld.setInsumo_id(insumo);
    ejbLibroDiarioFacade.create(ld);
    ejbLibroDiarioFacade.flushEntities();

    ejbInventarioFacade.reducirStock(insumo.getInsumo_id(), ld.getLibro_id());

    return Response.ok("Ok").build();
}

```

Fig 38. Código de la web services registrarLibroDiario

- **Actualización Inicio de Sesión mediante tokens**

El problema principal con la que contaba es el inicio de sesión; ya que al momento de recargar la página la sesión caducaba o se salía. Por ello fue necesario la implementación del token mediante JSON denominado *localStorage*, la cual nos permite corregir este error y la sesión no caducara a menos que nosotros le demos al botón de salir (Vanamala, 2019). Algo necesario para que el ingreso sea exitoso, los datos del médico a más de estar registrados deben estar habilitados, caso contrario no podrá ingresar al sistema.

```

ngOnInit() {
  if(this._localStorage.length < 1 || JSON.parse(localStorage.getItem('estado'))==null){
    this.router.navigate(['/inicio-sesion'])
    this.menuCtrl.enable(false);
  }
  this.inicioservice.$getObjectSource.subscribe(
    data=>{
      console.log("datos en el drawer",data)
      this.correo=data
      this.correop=this.correo.correo
      // console.log(JSON.parse(localStorage.getItem('estado')))
      this.miperfilservice.getMedico(JSON.parse(localStorage.getItem('estado'))).subscribe((dataa=>{
        this.medico=dataa
        console.log("datos medicos en el app component",this.medico)
        console.log('Token:',Md5.hashStr(JSON.parse(localStorage.getItem('estado'))));
        this.menuCtrl.enable(true);
      })))
    })
}

```

Fig 39. Código de la web services tokens inicio de sesión

### 6.3. Interfaces de usuario finales

Esta sección presenta el diseño final de las interfaces de usuario, las cuales fueron creadas en el framework de Ionic:

- Interfaz página inicial

En esta interfaz se encuentran cargados los nuevos modulos de insumo, inventario y laboratorios.

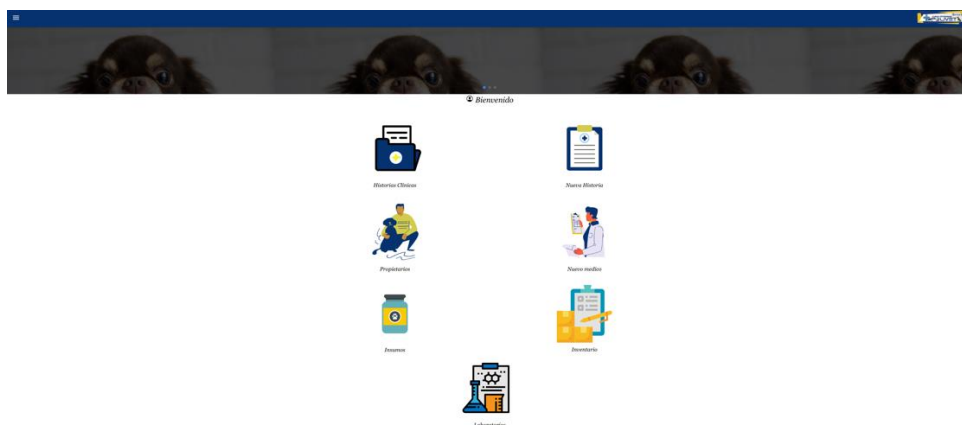
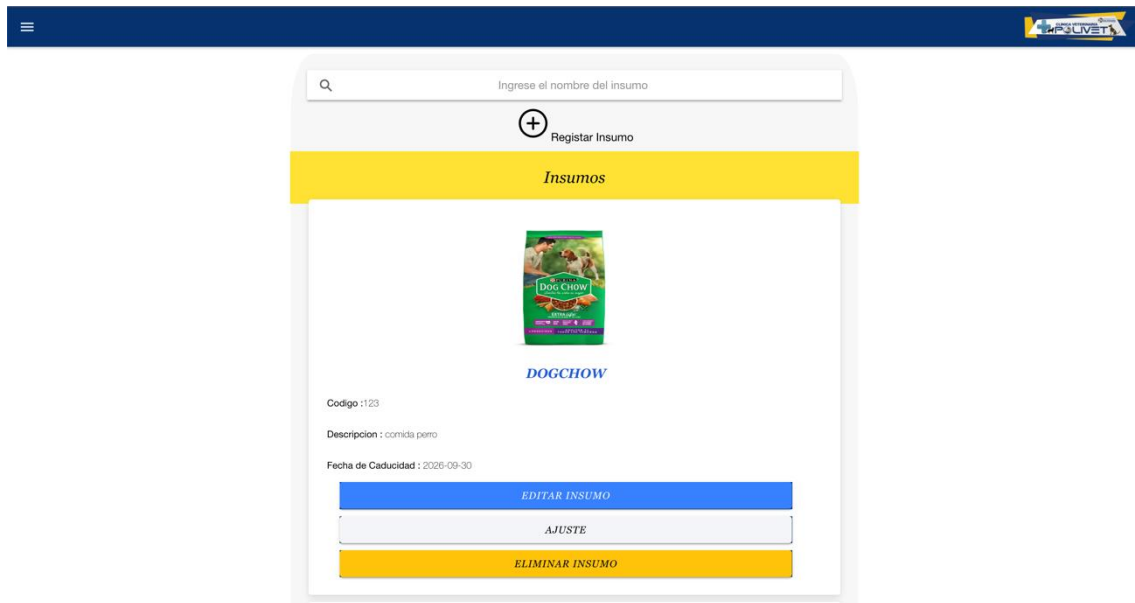


Fig 40. Interfaz página inicial Administrador

- Interfaz insumos

En esta interfaz se encuentran cargados todos los insumos médicos con los que trabaja la veterinaria.



*Fig 41. Interfaz insumos*

- Interfaz registrar insumo

En esta interfaz se encuentran todos los campos a llenar para registrar un insumo.



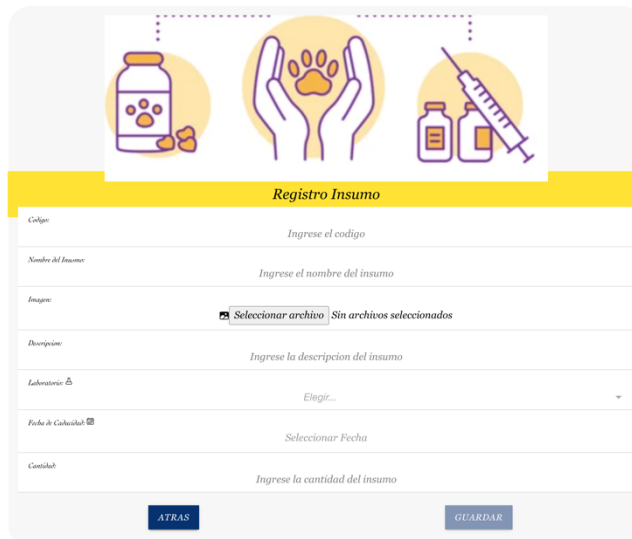


Fig 42. Interfaz registrar insumo

- Interfaz modificar insumos

En esta interfaz se encuentran cargados todos los campos del insumo seleccionado, con la finalidad de que el usuario los pueda modificar si así lo desea.



Fig 43. Interfaz modificar insumos

- Interfaz ajuste de inventario

En esta interfaz se puede actualizar el stock de un producto del inventario a través de sus procesos: aumentar y reducir.

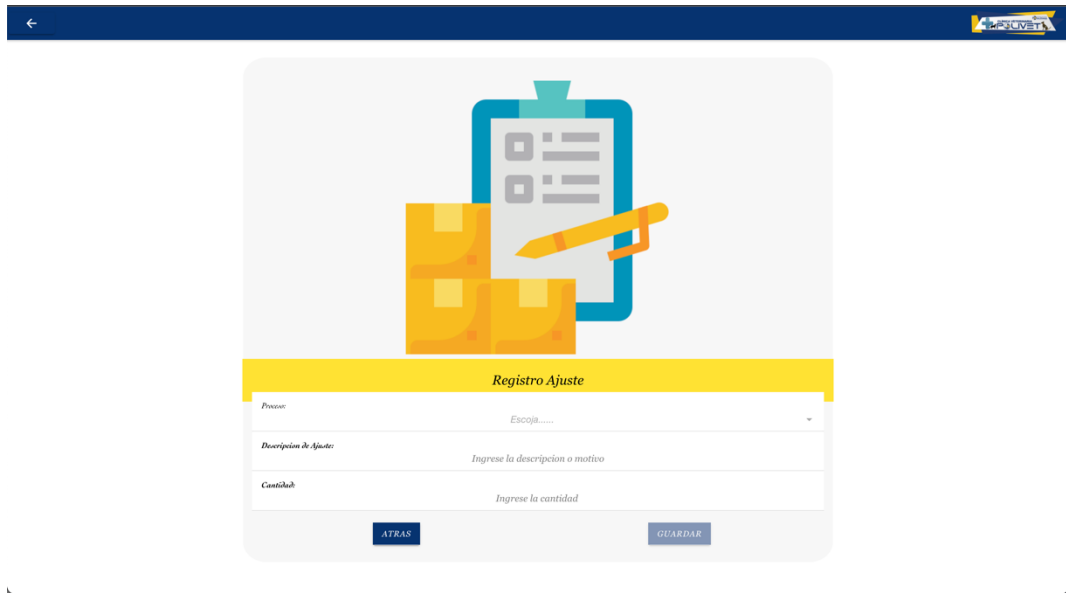


Fig 44. Interfaz ajuste de inventario

- Interfaz inventario

En esta interfaz se puede visualizar los campos importantes del insumo, especialmente el stock, ya que este campo informa la cantidad de insumos disponibles.



Fig 45. Interfaz inventario

- Interfaz historial ajuste

En esta interfaz se puede visualizar los cambios realizados en el inventario a través de los ajustes realizados.

Codigo	Insumo	Descripcion del Ajuste	Proceso	Cantidad de Ajuste
123	DOGCHOW	fallo	Aumentar	5
124	GATUCCO	fallo	Reducir	3
123	DOGCHOW	fallo	Aumentar	5
123	DOGCHOW	fallo	Reducir	5
125	Pruebaz	fallo en el libro diario	Aumentar	25
123	DOGCHOW	mal ingreso en el libro diario	Aumentar	20
125	Pruebaz	mal ingreso libro diario	Aumentar	20
125	Pruebaz	Mal ingreso libro diario (2022/06/27 00:12)	Aumentar	5
125	Pruebaz	aumento de stock	Aumentar	25
123	DOGCHOW	aumento prueba	Aumentar	10

Fig 46. Interfaz historial ajuste

- Interfaz registro libro diario

En esta interfaz realiza el ingreso de un insumo en el libro diario.

**Registro Libro Diario**

Fecha de Ingreso:  Seleccionar Fecha

Descripcion:  Ingrese la descripcion o motivo

Numero a ocupar:  Ingrese el numero de insumos a ocupar

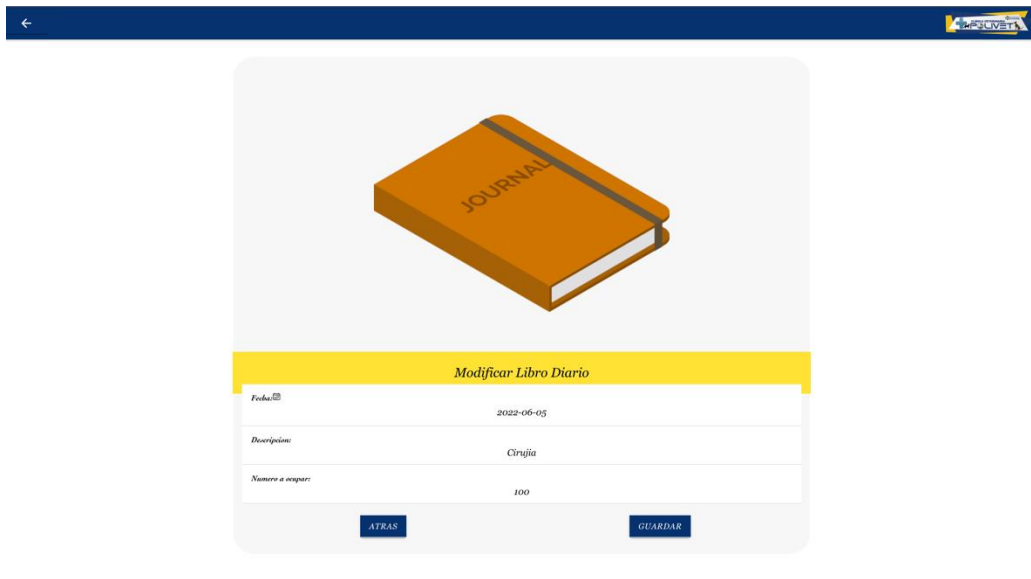
Numero del Insumo:  DOGCHOW

[ATRAS](#) [GUARDAR](#)

*Fig 47. Interfaz registro libro diario*

- Interfaz modificar libro diario

En esta interfaz se encuentran cargados todos los campos del libro diario seleccionado, con la finalidad de que el usuario los pueda modificarlos si así lo desea.



*Fig 48. Interfaz modificar libro diario*

- Interfaz historial libro diario

En esta interfaz se encuentran cargado el historial de todos los ingresos que ha tenido un insumo dentro del libro diario.

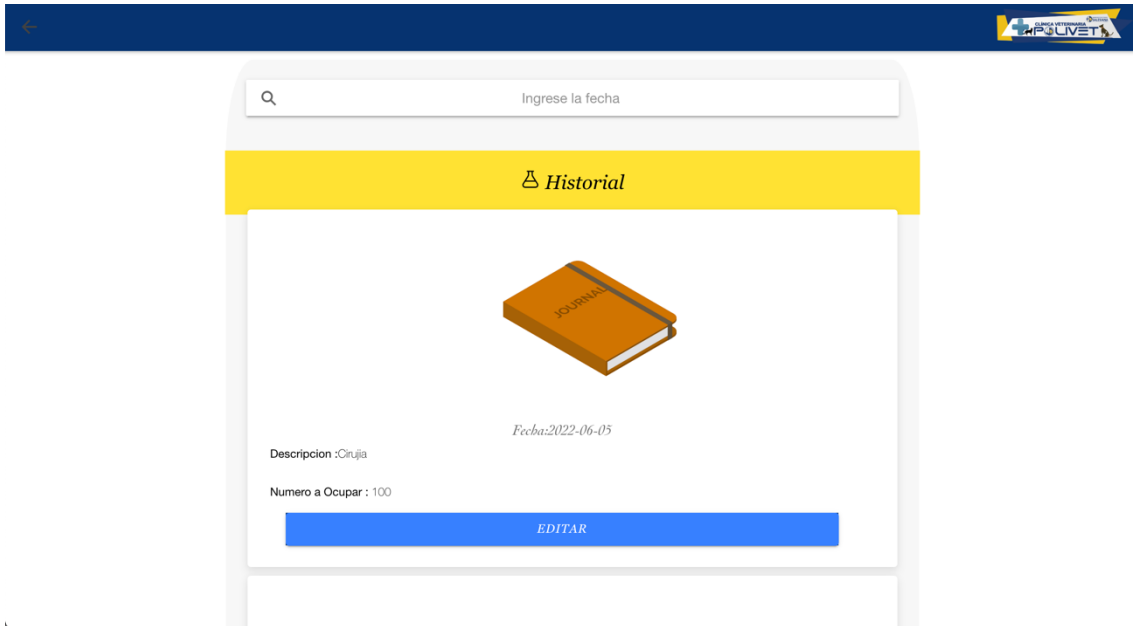


Fig 49. Interfaz historial libro diario

- Interfaz libro diario

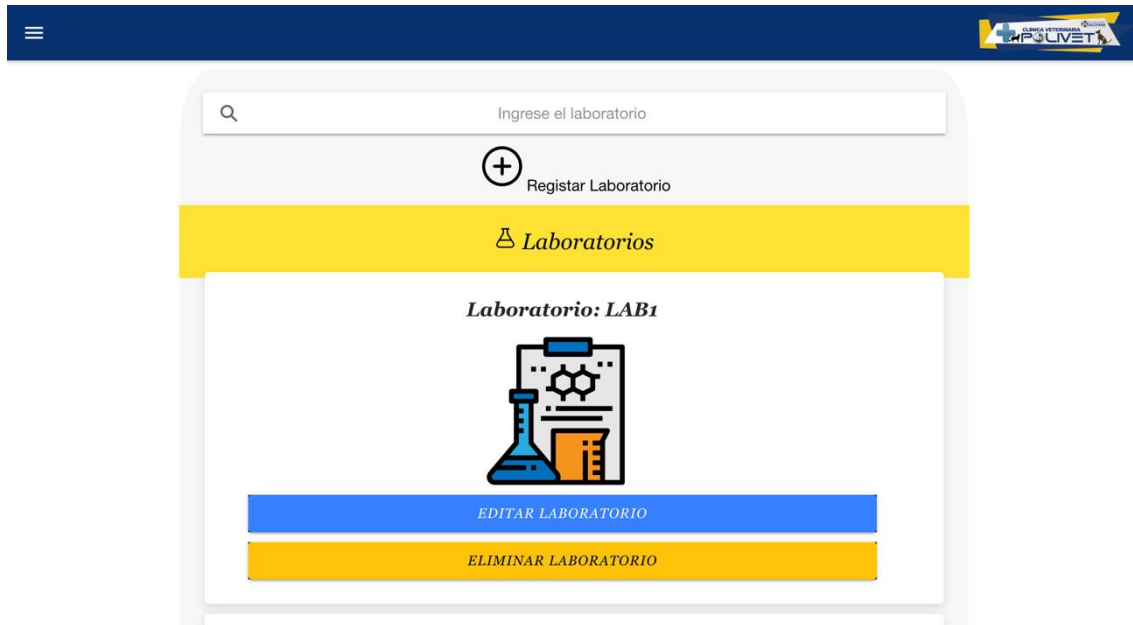
En esta interfaz se encuentran cargados todos los insumos que han sido registrados en el libro diario.

Insumo_Id	Insumo	Descripcion	Fecha de Ingreso	Numero a Ocupar
123	DOGCHOW	Cirujia	2022-06-05	100
123	DOGCHOW	prueba	2022-05-31	12
124	GATUCO	prueba1	2022-05-31	2
123	DOGCHOW	prueba2	2022-05-31	2
125	Prueba2	cirujia	2022-06-01	100
125	Prueba2	comida	2022-06-01	100
125	Prueba2	cirujia	2022-06-07	50
123	DOGCHOW	cirujia	2022-06-24	20
125	Prueba2	prueba nombre	2022-06-27	5
123	DOGCHOW	cirujia	2022-06-29	10

Fig 50. Interfaz libro diario

- Interfaz laboratorios

En esta interfaz se encuentran cargados todos los laboratorios que tienen insumos.



*Fig 51. Interfaz laboratorios*

- Interfaz registrar laboratorio

En esta interfaz se encuentran todos los campos a llenar para registrar un laboratorio.



*Fig 52. Interfaz registrar laboratorio*

- Interfaz modificar laboratorio

En esta interfaz se encuentran cargados todos los campos del laboratorio seleccionado, con la finalidad de que el usuario los pueda modificarlos si así lo desea.



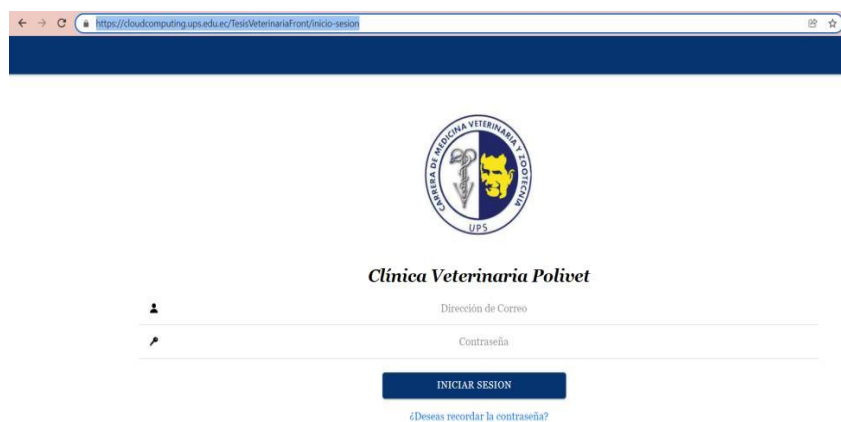
*Fig 53. Interfaz modificar laboratorio*

## 6.4 Despliegue de la aplicación

La aplicación se implementó en los servidores del grupo de investigación Cloud Computing, Smart City y High Performance Computing los cuales, se encuentran distribuidos de la siguiente manera: Base de Datos Mysql, servidor de aplicaciones Wildfly y un servidor web apache.

- **App PWA desplegada en la ip pública:**

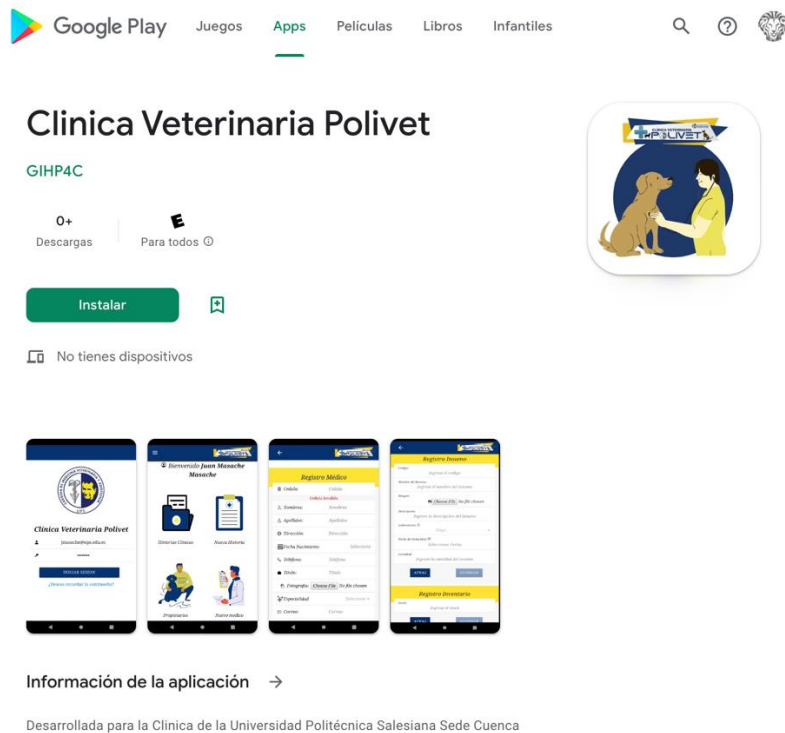
<https://cloudcomputing.ups.edu.ec/TesisVeterinariaFront/inicio-sesion>



*Fig 54. App PWA desplegada en apache a través de una ip pública*

- **App PWA desplegada en la Google Play Store:**





*Fig 55. Despliegue de la app en la Google play store*

## 6.5 Encuesta

Con la encuesta lograremos conocer el nivel de aprobación que se obtuvo al realizar el sistema multiplataforma de la veterinaria. En la sección de anexos se mostrará las preguntas y las respuestas que se consiguió con dicha encuesta.

Para que nuestra encuesta sea válida y que nos corrobore que nuestro programa cumple con todos los requerimientos establecidos, la dicha encuesta se le aplicó a la Dra. Carmen Angélica Yanzaguano Romero quien está a cargo del manejo del sistema veterinario “Polivet”. Los resultados que se obtuvo con las siguientes preguntas son:

### 6.5.1 Resultado Pregunta 1:

El resultado de la primera pregunta que se planteó sobre si la aplicación desarrollada es sencilla al momento de utilizar; la respuesta fue afirmativa.

### 6.5.2 Resultado Pregunta 2:

En respuesta de la segunda pregunta sobre si se puede navegar por las distintas ventanas de la aplicación multiplataforma de acuerdo a los componentes según la información física con relación de los insumos; la respuesta de afirmativa.

### **6.5.3 Resultado Pregunta 3:**

En respuesta a la pregunta de la navegación con las diferentes interfaces por la cual se navega en las distintas ventanas del sistema, de acuerdo a los componentes en relación al libro diario; la relación fue afirmativa.

### **6.5.4 Resultado Pregunta 4:**

En respuesta a la pregunta 4 sobre si el sistema cumple con la solución de los requerimientos en la gestión de insumos, la respuesta fue afirmativa.

### **6.5.5 Resultado Pregunta 5:**

En respuesta del sistema multiplataforma sobre la posibilidad de interactuar con los insumos que dispone el formulario de los datos ingresado; la respuesta fue afirmativa.

### **6.5.6 Resultado Pregunta 6:**

En respuesta a la pregunta de que si el sistema cumple con todos lo plantado con relación a los requisitos iniciales; la respuesta fue afirmativa.

### **6.5.7 Resultado Pregunta 7:**

En respuesta de que si está conforme con la aplicación multiplataforma y que si lo usaría en el campo laboral; la respuesta fue afirmativa.

En la siguiente sección veremos el cronograma de actividades la que se siguió para cumplir con el objetivo final que es la creación y el funcionamiento adecuado de la aplicación de la veterinaria.

## 7. Cronograma de actividades

### Cronograma de Actividades

*Tabla 35. Total, horas de los desarrolladores*

<b>Total, Horas 600 Horas</b>	
<b>Kevin Adrián Godoy Mendía (K.G.)</b>	<b>300 Hrs</b>
<b>William Eduardo Chabla Palaguachi (W.C.)</b>	<b>300 Hrs</b>
<b>Ing. Gabriel Alejandro León Paredes, PhD (G.L.)</b>	<b>50 Hrs</b>
<b>Fecha inicio: 28/03/2022</b>	<b>Fecha Fin:26/07/2022</b>

*Tabla 36. Cronograma de actividades de cada objetivo específico*

<b>Sprint</b>	<b>Objetivo de aprendizaje</b>	<b>Nombre de Tarea</b>	<b>Horas</b>	<b>Duración</b>	<b>Comienzo</b>	<b>Fin</b>	<b>Responsable</b>
Uno	<b>OE.1</b>	<b>ACT 1.</b> Determinar y adentrarse en las tecnologías más comunes para la implementación de sistemas multiplataforma (JavaScript, CSS, Ionic, PWA, HTML)	25 horas	50	lun 28/03/22	mie 30/03/22	W.C.
		<b>ACT 2.</b> Bocetos en el campo de la medicina veterinaria en: área de contabilidad, requerimientos del mercado y demanda de herramientas tecnológicas.	25 horas		jue 31/03/22	mar 05/04/22	K.G.
	<b>OE. 2</b>	<b>ACT 1.</b> Determinar procesos de administración de la veterinaria con relación a la gestión de insumos.	24 horas	30	mar 05/04/22	lun 11/04/22	K.G.

		<b>ACT 2.</b> Realizar diseño y modelo de cada proceso de administración en la veterinaria relacionados con la gestión de insumos.	30 horas		mar 12/04/22	jue 14/04/22	W.C.
		REVISIÓN / CORRECCIÓN	10 horas		vie 15/04/22	lun 17/04/22	G.L. - K.G. - W.C.
Uno -	<b>OE. 3</b>	<b>ACT 1.</b> Junta con los stakeholders.	32 horas	70	lun 18/04/22	mie 20/04/22	K.G. - W.C.
		<b>ACT 2.</b> Recopilación de requisitos.	50 horas		Jue 21/04/22	lun 25/04/22	W.C.
		<b>ACT 3.</b> Estudio de requisitos con los stakeholders.	35 horas		mar 26/04/22	jue 28/04/22	K.G.
		<b>ACT 4.</b> Diseño de modelo ER (Entidad Relación).	40 horas		vie 29/04/22	mar 03/05/22	K.G. - W.C.
		<b>ACT 5.</b> Elaboración del prototipo multiplataforma para la aplicación.	40 horas		mie 04/05/22	jue 05/05/22	K.G. - W.C.
		REVISIÓN / CORRECCIÓN	10 horas		vie 06/05/22	lun 09/05/22	G.L. - K.G. - W.C.
	<b>OE. 4</b>	<b>ACT 1.</b> Elaboración de los servicios web.	50 horas	350	mar 10/05/22	mie 25/05/22	K.G. - W.C.
		<b>ACT 2.</b> Diseño de backend e implementación de los servicios web.	80 horas		jue 26/05/22	dom 05/06/22	K.G. - W.C.

Dos	<b>ACT 3.</b> Diseño de frontend.	80 horas	lun 06/06/22	lun 13/06/22	K.G. - W.C.
	<b>ACT 4.</b> Implementación del backend y frontend.	75 horas	mar 14/06/22	mie 22/06/22	K.G. - W.C.

Dos- Tres	<b>OE. 5</b>	<b>ACT 1.</b> Despliegue de aplicación en el servidor del grupo de investigación GiHP4C	21 horas	70	jue 07/07/22	mie 13/07/22	KG. - W.C.
		<b>ACT 2.</b> Ejecución del plan de pruebas.	21 horas		jue 14/07/22	lun 18/07/22	KG. - W.C.
		<b>ACT 3.</b> Recolección de resultados del plan de pruebas.	23 horas		mar 19/07/22	jue 21/07/22	KG. - W.C.
		REVISIÓN / CORRECCIÓN	30 horas	30	vie 22/07/22	mar 26/07/22	G.L. - K.G. - W.C.

## 8. Presupuesto

Tabla 37. Presupuesto Fijo

DENOMINACIÓN	CANTIDAD	COSTO UNITARIO	COSTO TOTAL
	unidades	dólares	dólares
<b>1.Bienes</b>			
Copias	100	0,01	1.0
Impresiones	200	0,05	10
Empastados	2	12	24
<b>2.Tecnológico</b>			
Computador Portátil	2	1500	3000
Celular Inteligente	2	500	1000
Servidor en la nube	1	100	100
<b>3.Servicios</b>			
Servicio de transporte	366	2,55	933,3
Servicio de internet	12	40	480
Servicio de luz	12	20	240
Taxis	0	0	0
Alimentación	250	2,50	625
<b>4.Personal</b>			
Estudiante investigador	600 horas (dos estudiantes)	2.65 por hora	1590
<b>5.Otros</b>			
Imprevistos	2	100	200
<b>Total</b>			<b>\$ 8203. 30</b>



## 9. Conclusiones

En la actualidad las clínicas veterinarias necesitan aplicativos que les ayude en las tareas diarias, para que de esta manera sea una solución práctica y eficiente; así lograr soluciones más modernas, como lo vemos en la actualidad y la necesidad de tener nuestras aplicaciones conectadas en la red; una facilidad en donde el médico pueda gestionar las diferentes tareas como: registrar, consultar, administrar y llevar un reporte de todas sus actividades, a fin de brindar un mejor servicio a sus clientes.

Mediante este aplicativo multiplataforma se logró la creación para gestión de insumos médicos que se utilizan diariamente en la clínica veterinaria denominada “Polivet”, obteniendo resultados favorables en el desarrollo de la misma. En vista de que el aplicativo tenía que ser multiplataforma se implementaron herramientas modernas que se están utilizando en la actualidad y además son con las que se trabajó en un primer indicio. La herramienta para la parte del backend se usó IntelliJ IDEA, para la parte de frontend se centró en Ionic y la base de datos con la que se trabajo es MySQL. El módulo de la gestión de insumos fue desarrollado mediante estas herramientas, obteniendo resultados óptimos y solventes.

La aplicación multiplataforma permite que el médico pueda registrar los insumos médicos, para que de esta manera llevar un mejor control de sus productos y tener un control en su inventario y conocer el stock con el que cuenta. Además, el libro diario ayuda a tener un mejor control según como se vayan utilizando los insumos médicos y en caso de algún fallo o equivocación, tanto en el registro del libro diario como de los insumos; el médico podrá realizar un ajuste en su inventario. Además de ello; el médico podrá generar reportes tanto del inventario, libro diario y el ajuste. Finalmente se logró corregir el inicio de sesión mediante tokens, a fin de que en el momento de que el usuario recargue la página la sesión no finalizara.

Para que todo esto quede en constancia de su correcto funcionamiento se realizó las pruebas pertinentes, como la de una encuesta al médico encargado del uso del sistema; la misma que se puede observar en la sección de resultados, en la cual se obtuvo respuestas positivas que nos indican que la interfaz gráfica desarrollada es fácil de usar para los usuarios, además de mostrar a los mismos toda la información referente

a los insumos y libro diario. Por otro lado, se pudo observar que el sistema cumple con los requerimientos de gestión de insumos además de posibilitar la interacción entre el usuario y los insumos ya registrados. Con lo cual, concluimos que la aplicación cumple con el correcto funcionamiento de todos los módulos, está solventada para un futuro crecimiento a corto, mediano o largo plazo, y podrá ser utilizada dentro del ámbito laboral de la clínica para la cual fue desarrollada.

## 10. Recomendaciones

- Otorgar credenciales de acceso únicamente al personal autorizado de la clínica veterinaria, en este caso solamente a los médicos.
- Asignar nombres a los archivos descargables en formato csv. Esto se debe realizar con el fin de llevar una mejor organización con respecto a cada uno de los ficheros, sean estos de inventario, historial o libro diario.
- Con respecto a la carga de imágenes en los distintos módulos de la aplicación multiplataforma se recomienda usar extensiones JPG o PNG ya que son las que mejor se acoplan con el campo de la imagen.
- Realizar pruebas locales antes de publicar el archivo con extensión war en el servidor de aplicaciones, para así garantizar la funcionalidad de todos los API rest.
- Se recomienda dar lectura al manual de usuario con los módulos desarrollados, esto a fin de entender como están estructurados estos y a su vez aprender a realizar de manera correcta las diferentes acciones que la aplicación ofrece a los médicos veterinarios.
- Se recomienda dar lectura al manual de despliegue de la aplicación, esto a fin de entender cómo se encuentra desplegada en el servidor, ya que estos pasos garantizarán la reducción de tiempo a los desarrolladores.

## Referencias Bibliográficas

- Iza Cajia, C. M., & Companioni Vargas, H. (2022). Desarrollo de una aplicación multiplataforma utilizando el marco de desarrollo ionic para la gestión de la historia clínica de la veterinaria “Polivet” . *Universidad Politécnica Salesiana*.
- Cornejo Gómez, D. (2016). Análisis de la falta de control en el sistema de inventarios y logística de la empresa fármaco veterinaria SA Favasa de la ciudad de Guayaquil en el periodo 2010-2014.
- Guerrero Coveña, Y. E. (2017). DESARROLLO DE UN MÓDULO EN EL FRAMEWORK ODOO PARA LA ADMINISTRACIÓN ELECTRÓNICA DEL INVENTARIO DE INSUMOS MÉDICOS EN LA CLÍNICA SAN PABLO DE LA CIUDAD DE MANTA.
- Guamán, & Patiño. (2016). Diseño e Implementación de un sistema web para el diagnóstico de enfermedades en el hospital veterinario de la UNL.
- Segarra Briones, J. A., & Segarra Briones, J. S. (2020). IMPLEMENTACIÓN DE UN SISTEMA DE SERVICIOS, GESTIÓN Y CONTROL DE COMPRAS, VENTAS E INVENTARIOS OFRECIDOS POR LA VETERINARIA “DR GUAU” .
- García, O. R. (2009). Historia de la Medicina Veterinaria. REDVET, 10(4).
- Garcia Loor, Y. Y. (2019). Desarrollo de aplicación web para la gestión de consultas y agendamiento de citas de mascota de la clínica veterinaria burgos (Bachelor's thesis).
- DEL AGRO-AGROCALIDAD, C. A. (2016). Considerado.

- Lagos Rugama, D. S., López Ruiz, J. M., & Vásquez M., D. S. (2015). Aplicación del control interno de inventario de Insumos médicos y distribución por el Centro de Salud Leonel Rugama a los puestos de Salud del Municipio de Estelí .
- Al-Sarayreh, K. T., M. K., & Abran, A. (2021). Software engineering principles: A systematic mapping study and a quantitative literature review. *Engineering Science and Technology, an International Journal*.
- Wong, W. M., & Li, Y. (2021). A bibliometric assessment of software engineering themes, scholars and institutions. *Journal of Systems and Software*, 180, 111029. 10.1016/j.jss.2021.111029.
- Zave, P. (1997). Classification of research efforts in requirements engineering. *ACM Computing Surveys* , 315-321.
- Chaves, M. A. (2005). La ingeniería de requerimientos y su importancia en el desarrollo de proyectos de software. . *Revista de las Sedes Regionales*, 1-13.
- Dick, J. H., & Jackson, K. (2017). System Modelling for Requirements Engineering. In *Requirements Engineering*. . *Springer, Cham.*, 57-92.
- Oliveros, A., & Antonelli, R. L. (2015). Técnicas de elicitación de requerimientos. *XXI Congreso Argentino de Ciencias de la Computación (Junín)* .
- Al Abid, F. B., & Karim, A. R. (2017). Cross-platform development for an online food delivery application. . In *2017 International Conference on Computing Networking and Informatics. IEEE.*, 1-4.
- Ionic Docs. (2022). Open-Source UI Toolkit to Create Your Own Mobile or Desktop Apps.
- Tabarés, R. (2021). HTML5 y la evolución de HTML; rastreando los orígenes de las plataformas digitales. *Tecnología en la sociedad* , 65 , 101529.
- Coremans, C. (2015). HTML: A Beginner's Tutorial. *Brainy Software Inc.*

- Mohedano, J. S., & Román, P. S. (2012). Iniciación a javascript. *Ministerio de Educación*.
- Ribes Alba, P. (2013). Manual de Javascript. Editorial CEP, S.L. .
- Bierman, G. A., & Torgersen, M. (2014). Understanding typescript. *European Conference on Object-Oriented Programming* , 257-281.
- Freeman, A. (2021). Understanding typescript. *Essential TypeScript 4* , 35-41.
- Hao, Z., Limiao, Z., & Hua, H. (2012). Un modo de diseño web para navegadores con problemas de compatibilidad con CSS. *Cuarta Conferencia Internacional sobre Redes y Seguridad de la Información Multimedia* , (págs. 160-163). IEEE.
- York, R. (2004). Beginning Css Cascading Style Sheets For Web Design.
- Ogala, J. O., & Ojie, D. V. (2020). Comparative analysis of c, c++, c# and java programming languages.
- Munoz, F. R., & Villalba, L. G. (2012). Preproceso de formularios para el análisis de seguridad de las aplicaciones web. *Reunión Española sobre Criptología y Seguridad de la Información(RECSI 2012)*.
- Hernández, D. N. (2007). Propuesta de desarrollo basada en estándares para aplicar una Arquitectura Orientada a Servicios en la Universidad de la Ciencias Informáticas (Doctoral dissertation).
- Morales, C. A. (2010). Estado del Arte: Servicios Web. *Universidad Nacional de Colombia*.
- Castro, M. S. (2013). Aplicación de Servicios Web SOAP/REST para funcionalidades existentes en sistemas informáticos provinciales. *VII Simposio Argentino de Informática en el Estado* .

- . Hidalgo Macas, L. N., & Jiménez Acaro, M. E. (2016). Estudio Comparativo de los Servicios Web Restfull Jersey y SOAP JAX-WS para el Desarrollo de una Aplicación Android con Wikitude Aplicada a la Gestión de Información Geolocalizada del Turismo de la Provincia de Chimborazo. *Escuela Superior Politécnica de Chimborazo* .
- Galarza Acosta, G. (2017). Diseño e Implementacion de una Aplicación Móvil para Uso de Pacientes con Problemas de Diabetes de Tipo 1, Tipo 2 y Gestacional, enfocado en Arquitectura de Web Services, Creacion y Disponibilizacion de Servicios Web RestFul Hacia la Base de Datos MARIADB. *Universidad de Guayaquil*.
- Aguirre, V., Ortu, A., Delía, L. N., Thomas, P. J., Corbalán, L. C., Cáseres, G., & Pesado, P. M. (2019). PWA para unificar el desarrollo Desktop, Web y Mobile. . *XXV Congreso Argentino de Ciencias de la Computación (CACIC)(Universidad Nacional de Río* .
- Tiwari, A., Prakash, J., Groß, S., & Hammer, C. (2020). A Large Scale Analysis of Android — Web Hybridization. *Journal of Systems and Software*.
- Valverde, V., Portalanza, N., & Mora, P. (2019). Análisis descriptivo de base de datos relacional y no relacional. *Cuadernos de Educación y Desarrollo, (108)*.
- Setyawati, E., Wijoyo, H., & Soeharmoko, N. (2020). Relational Database Management System (RDBMS).
- Cano, E. L.-C. (2021). A Scrum-based framework for new product development in the non-software industry. *Journal of engineering and technology management*.
- Cano, E. L., García-Camús, J. M., Garzás, J., Moguerza, J. M., & Sánchez, N. N. (2021). (2021). A Scrum-based framework for new product development in the non-software industry. *Journal of engineering and technology management*.

- HABER, F. C. (s.f.). “ATENCIÓN RÁPIDA “LIBRO DIARIO. 1.1 ANTECEDENTES. 198.
- Danielecki, D. M. (2019). Security First approach in development of Single-Page Application based on Angular. *Master's thesis, University of Twente.*
- Amat, D. A. (2018). F. I. R. COMPONENTE DE GENERACIÓN DINÁMICA DE SERVICIOS REST.
- Saravia, L. E. (2001). Servidor web multiplataforma con implementación CGI. *In VII Congreso Argentino de Ciencias de la Computación.*
- Montes-Cumbrera, J. &.R. (2015). Creación de un servicio Web REST y su despliegue en Tomcat.
- Wetherbee, J. N. (2018). EJB Session Beans. *In Beginning EJB in Java EE 8*, pp. 39-92.
- Vanamala, M. Y. (2019). A course module on HTML5 new features and security concerns. *Journal of Computing Sciences in Colleges*, 34(5), 23-30.



## Anexos

### Anexo 1. Prospección de la Encuesta

- **Interrogante 1:** ¿Está de acuerdo que la aplicación desarrollada es sencilla de utilizar?

*Tabla 38. Interrogante 1. Factibilidad al momento de usar*

Opción	Respuesta
Afirmativo	X
Negativo	

- **Interrogante 2:** ¿Las diferentes interfaces por la cual se navega en las distintas ventanas del sistema multiplataforma se puede observar los componentes según a la información física con relación a los insumos?

*Tabla 39. Interrogante 2. Sistema según la información del inventario*

Opción	Respuesta
Afirmativo	X
Negativo	

- **Interrogante 3:** ¿Las diferentes interfaces por la cual se navega en las distintas ventanas del sistema multiplataforma se puede observar los componentes según a la información física con relación al libro diario?

*Tabla 40. Interrogante 3. Sistema según la información del libro diario*

Opción	Respuesta
Afirmativo	X
Negativo	

- **Interrogante 4:** ¿En momento el sistema multiplataforma cumple con solucionar los requerimientos en la gestión de los insumos?

*Tabla 41. Interrogante 4. Sistema de gestión de insumos*

Opción	Respuesta
Afirmativo	X
Negativo	

- **Pregunta 5:** ¿El sistema multiplataforma le posibilita interactuar con los insumos que dispone en el formulario; ósea con los datos ya ingresados?

*Tabla 42. Interrogante 5. Interacción con el formulario del insumo*

Opción	Respuesta
Afirmativo	X
Negativo	

- **Interrogante 6:** ¿De manera general la aplicación satisface el cumplimiento de todo lo planteado de cuerdo a los requisitos iniciales?

*Tabla 43. Interrogante 6. Cumple con los requerimientos*

Opción	Respuesta
Afirmativo	X
Negativo	

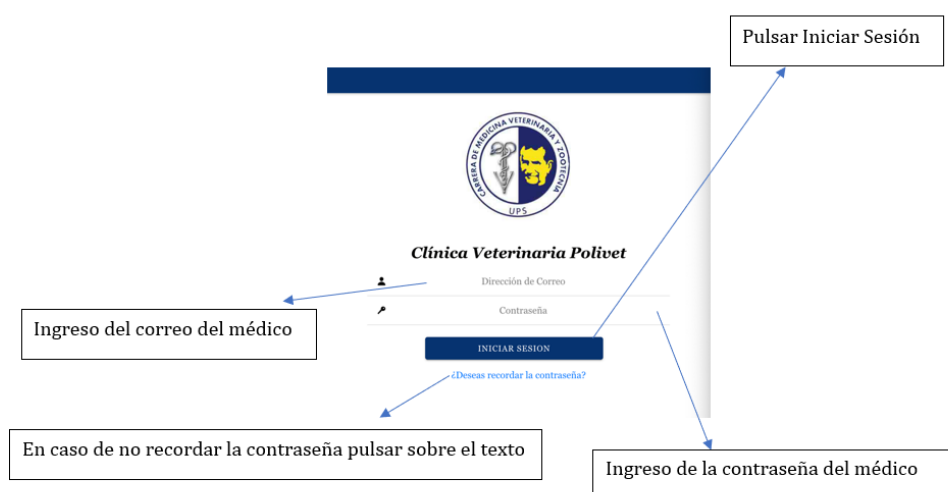
- **Interrogante 7:** ¿Según lo mencionado anteriormente y los requerimientos cumplidos estaría usted conforme a empezar a usar la aplicación multiplataforma en su campo laboral?

**Tabla 44. Interrogante 7. Aplicación en el campo laboral**

Opción	Respuesta
Afirmativo	X
Negativo	

## Anexo 2. Manual de usuario

### Anexo 2.1 Página de inicio de sesión



### Anexo 2.2 Página error 404



## Anexo 2.3 Página listado de Inventario

The screenshot shows the 'Inventario de Insumos' interface. Callouts provide the following information:

- LIBRO DIARIO**: Al pulsarlo accede al módulo del libro diario
- INSUMO**: Al pulsarlo accede al módulo de los Insumos
- GENERAR REPORTE**: Al pulsarlo genera un reporte en formato csv
- La barra de búsqueda**: permite buscar en el inventario por insumo
- A Caducar**: Al pulsarlo ordena los items por fecha de caducidad de manera ascendente
- Stock**: Al pulsarlo ordena los items por Stock de manera ascendente
- Paginación**: La paginación esta dividida en 10 items por página
- ATRAS**: Al pulsarlo accede al módulo de Insumos
- HISTORIAL DE AJUSTE**: Al pulsarlo muestra el historial de ajuste de inventario

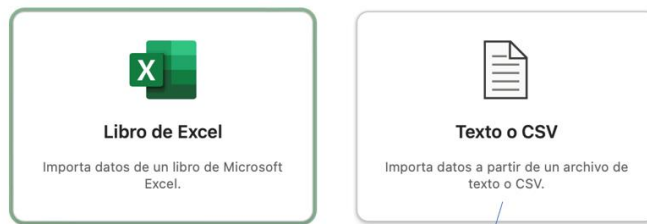
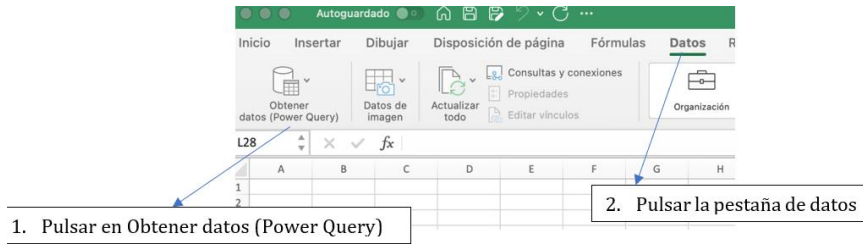
Código	Insumo	Descripción	Laboratorio	F.Caducidad	Stock
123	DOGCHOW	comida perro	LAB1	2026-09-30	70
124	GATUCO	GATOS	LAB2	2022-09-30	90
125	Pruebas	insumo de prueba	POLITECLAB	2024-08-29	2775

## Anexo 2.4 Generación de reportes

The screenshot shows the Microsoft Excel start screen. Callouts provide the following information:

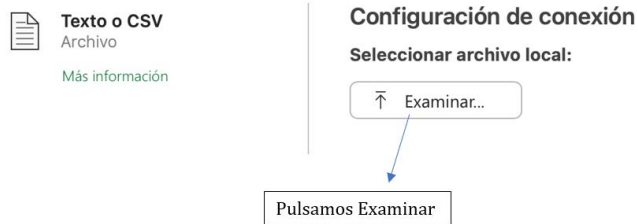
- Selección de libro en blanco**: Seleccionar libro en blanco (pointing to the 'Libro en blanco' template)
- Pulsar Crear**: Pulsar Crear (pointing to the 'Crear' button)

## Anexo 2.5 Carga CSV



Seleccionamos la opción de Texto o CSV

### Conectarse a un origen de datos



Se procede a buscar el archivo generado en la carpeta donde se haya descargado el archivo y posteriormente lo seleccionamos y le damos click en obtener datos.



Configuración de conexión

Seleccionar archivo local:

Examinar... Reporte.csv  
/Users/kevingodoy/Downloads/Reporte.csv



Cancelar Siguiente

Pulsamos Siguiente

Vista previa de los datos del archivo

Ruta de acceso al archivo: /Users/kevingodoy/Downloads/Reporte.csv

Origen de archivo: 65001: Unicode (UTF-8) | Delimitador: Coma | Detección del tipo de datos: Basado en las primeras 200 filas

Codi...	Insu...	Descripción	Laborat...	F.Caducid...	Stock
123	DOOCH...	comida perro	LAB1	30/9/2026	70
124	GATUCO	GATOS	LAB2	30/9/2022	90
125	Provebat	insumo de p...	POLITECLAB	28/8/2024	2775

Atrás Cancelar Cargar

Finalmente pulsamos cargar

## Anexo 2.6 Listado Libro Diario

Al pulsarlo accede a la página listado-insumo-libro-diario

Al pulsarlo genera un reporte en formato csv (Se recomienda revisar la sección 3.1 GENERACIÓN DE REPORTE EN ESTE DOCUMENTO)

La barra de búsqueda permite buscar en el libro diario por insumo o fecha de ingreso

Al pulsarlo accede al módulo de Insumos

Al pulsarlo accede al módulo de los Insumos

La paginación esta dividida en 10 items por página

Insumo_id	Insumo	Descripcion	Fecha de Ingreso	Numero a Ocupar
123	DOGCHOW	Cirujia	2022-06-05	100
123	DOGCHOW	prueba	2022-05-31	12
124	GATUCO	prueba1	2022-05-31	2
123	DOGCHOW	prueba3	2022-05-31	2
125	Prueba1	cirujia	2022-06-01	100
125	Prueba1	comida	2022-06-01	100
125	Prueba1	cirujia	2022-06-07	50
123	DOGCHOW	cirujia	2022-06-24	20
125	Prueba1	prueba nombre	2022-06-27	25

## Anexo 2.7 Listado Insumos Libro Diario

La barra de búsqueda permite buscar por nombre del insumo

Al pulsarlo accede a la página de registro de libro diario.

Al pulsarlo accede al historial de ingreso de ese insumo en el libro diario

La paginación esta dividida en 10 items por página

Al pulsarlo accede al módulo de insumos.

## Anexo 2.8 Listado Historial insumo

La barra de búsqueda permite buscar por fecha de ingreso del insumo.

Al pulsarlo accede a la página de modificar libro diario.

La paginación esta dividida en 10 items por página

## Anexo 2.9 Registro Libro Diario

Al pulsarlo regresamos a la página listado-insumos-libro

Al pulsarlo se nos abrirá una caja con fechas a seleccionar (Se recomienda revisar la sección Selección de fecha en el registro del libro diario)

Se deberá ingresar el motivo del ingreso de ese insumo (definir cuál será su uso)

Se deberá ingresar la cantidad de insumos a utilizar (Esto reducirá la cantidad del stock en el inventario)

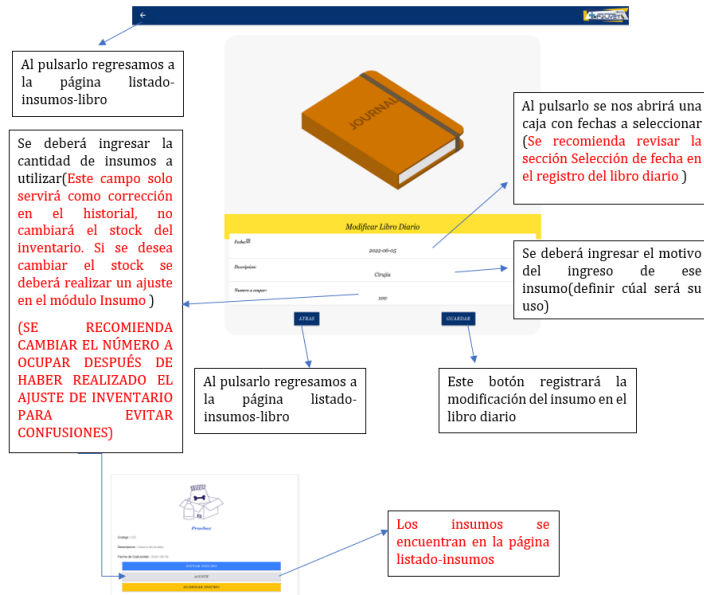
Al pulsarlo regresamos a la página listado-insumos-libro

Este parámetro no se debe tocar, ya que viene del insumo al cual se ha seleccionado el registrar

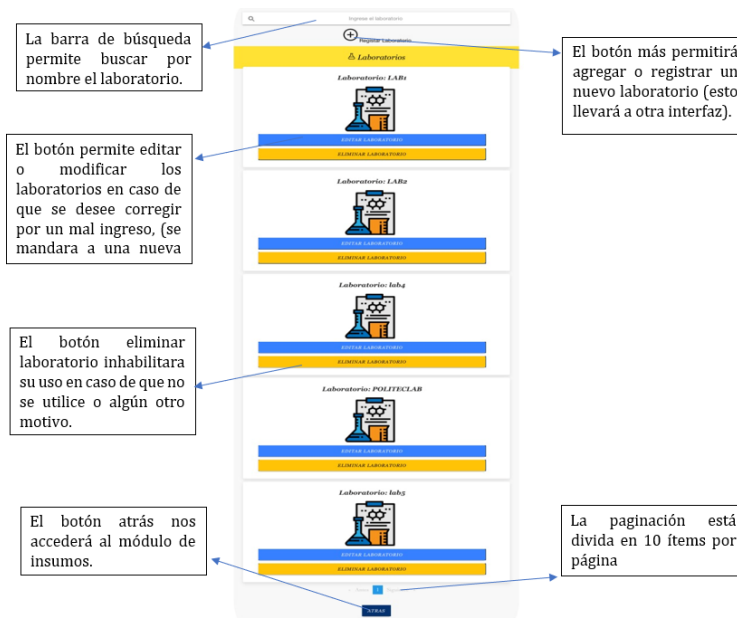
Se deberá haber ingresado todos los campos vacíos para que se active el botón de guardar; el cual registrará el insumo en el libro diario



## Anexo 2.10 Modificar Libro Diario



## Anexo 2.11 Listado Laboratorio



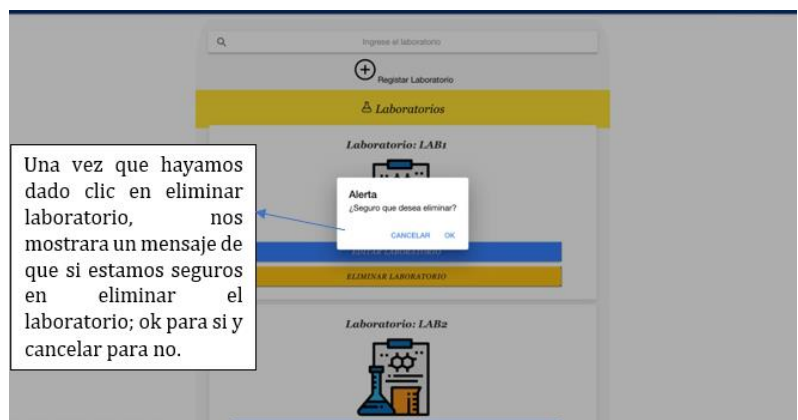
## Anexo 2.12 Registro Laboratorio



## Anexo 2.13 Modificar Laboratorio



## Anexo 2.14 Eliminar Laboratorio



## Anexo 2.15 Listado insumo

El buscador permite encontrar el insumo por nombre.

El más permitirá registrar un nuevo insumo.

El editar insumo ayuda a que logremos corregir el insumo en caso de fallos a la hora de ingreso del mismo, (se realizará en una nueva interfaz).

El ajuste permitirá realizar un reajuste en el inventario ya sea aumentar o disminuir la cantidad del insumo, (se realizará en una nueva ventana).

El eliminar permitirá quitar de nuestro inventario o registro dicho insumo

Atrás permitirá dirigirnos al menú del insumo

La paginación se mostrará en 10 ítems

## Anexo 2.16 Registrar Insumo

Ingreso del código del insumo (el código debe ser único).

Ingreso del nombre del insumo.

Se puede elegir el laboratorio que pertenece el insumo.

Colocamos la fecha de caducidad del insumo.

Colocamos la cantidad del insumo que se registra.

Regresa al menú del inventario.

Ingreso de la imagen del insumo guardado en nuestro dispositivo electrónico.

En esta sección se ingresa la descripción del insumo ya sea cajas, milímetros, empaques, etc.

Guarda los insumos en el registro de insumos.

Al ingresar la cantidad en su sección automáticamente en el stock se mostrará lo que ingresará al inventario y se actualizará al dar clic en guardar.

Guarda los insumos en el registro del inventario para el aumento en el stock.

## Anexo 2.17 Modificar Insumo

Como se observa al hacer clic en modificar, podemos ver que se recupera los datos y nosotros podemos modificar empezando desde el nombre

Al dar clic en seleccionar imagen podemos agregar una nueva imagen o la misma de nuestro dispositivo.

Al hacer clic en laboratorio se nos mostrara los laboratorios disponibles y podemos seleccionar el correcto.

Atrás permite regresar al menú de insumos

Una vez que estemos seguros podemos guardad los datos actualizados.

La fecha de caducidad viene dada por defecto la fecha actual, pero se puede modificar si se lo requiere.

Si queremos podemos modificar la descripción o la podemos dejar como esta si no es necesario corregirla.

## Anexo 2.18 Eliminar insumo

En caso de eliminar el insumo se nos mostrara una alerta de si estamos seguros o no en eliminar el insumo.

## Anexo 2.19 Registrar Ajuste

Al momento de realizar el ajuste observamos que el código se muestra por defecto. El código no es posible modificar desde aquí, debido a que es un ajuste de cantidad y no una modificación.

Al dar clic en proceso nos mostrara dos opciones aumentar o reducir (explicación en la siguiente interfaz).

Como se muestra en la sección podemos ingresar el motivo por la que se hizo el ajuste y la fecha que se realizo

De esta manera podemos seleccionar lo que deseamos hacer o realizar en el ajuste

Atrás para ingresar al menú de insumos

Al guardar podemos ver en el inventario el total con la que se cuenta en el stock

## Anexo 2.20 Historial de Ajuste

Al pulsarlo accede al módulo del libro diario

Al pulsarlo genera un reporte en formato csv

Al pulsarlo accede al módulo de los Insumos

La barra de búsqueda permite buscar en el historial de ajuste por insumo o proceso que en este caso solo pueden ser 2 : Aumentar o Reducir

Al pulsarlo accede al módulo de Insumos

La paginación esta dividida en 10 items por página

Colocamos la cantidad a reducir o aumentar la misma que se observará reflejada en el stock

Código	Insumo	Descripción del Ajuste	Proceso	Cantidad de Ajuste
123	DOGCHOW	fallo	Aumentar	5
124	GATUCO	fallo	Reducir	3
123	DOGCHOW	fallo	Aumentar	5
123	DOGCHOW	fallo	Reducir	5
125	Pruebas	fallo en el libro diario	Aumentar	25
123	DOGCHOW	mal ingreso en el libro diario	Aumentar	20