

CAPÍTULO I: GENERALIDADES

I.1 Introducción

Desde su inauguración en 2019, la Clínica Veterinaria “Málaga” ubicada en la calle España, entre Belgrano e Ingavi, en la ciudad de Tarija, Bolivia, ha proporcionado servicios de atención a mascotas, principalmente perros y gatos, tales como consultas, medicina preventiva y cirugía. Sin embargo, a lo largo de su trayectoria, ha enfrentado desafíos operativos significativos, especialmente en la gestión manual de historias clínicas, la interacción efectiva con los propietarios y el control de inventarios de medicamentos, aspectos que demandan mejoras sustanciales.

Este documento tiene como propósito principal presentar el proyecto de implementación de un software diseñado para automatizar y agilizar las tareas clínicas y administrativas, mejorando la precisión y accesibilidad de los registros y los historiales médicos de nuestros pacientes.

Esta iniciativa no solo fortalecerá la posición de la clínica como líder en servicios veterinarios, sino que también mejorará la experiencia tanto para los pacientes como para sus dueños.

I.2 Descripción del Proyecto

I.2.1 Antecedentes

La Clínica Veterinaria “Málaga” ha desempeñado un papel fundamental en la atención de mascotas desde su inauguración en el año 2019. Sin embargo, a lo largo de su funcionamiento, se identificaron desafíos significativos en la gestión de datos de pacientes, los cuales se registraban sin un formato estándar y se almacenaban en distintos lugares. Inicialmente, el proceso de registro se llevaba a cabo en formatos físicos, utilizando hojas impresas como historias clínicas.

Este método, aunque común en sus inicios, presentaba limitaciones en cuanto a la eficiencia y la organización de la información. Con el tiempo, se intentó mejorar la situación trasladando el registro a archivos Excel. Este enfoque implicaba la creación de tablas y archivos específicos para cada paciente, con la intención de estructurar la información de manera más ordenada.

En relación al inventario, el personal responsable lleva a cabo el registro diario de las ventas de productos expedidos bajo receta médica en un cuaderno. Sin embargo, no se realiza un seguimiento preciso del stock actual de estos productos. Esta práctica implica que no se cuenta con la información inmediata necesaria en caso de realizar una venta, ya que el sistema actual no gestiona este proceso. Es importante destacar que el sistema propuesto se enfoca exclusivamente en la gestión de medicamentos utilizados al emitir recetas médicas durante las atenciones clínicas y el control de stock de los mismos, sin abordar el manejo financiero asociado a las ventas realizadas.

A pesar de estos esfuerzos, la falta de un formato estandarizado y la dispersión de la información en diferentes archivos dificultaban la accesibilidad y la consistencia en la gestión de datos clínicos. Además, la necesidad de consultar registros físicos o buscar archivos específicos en la computadora generaba demoras y complicaciones en el proceso de atención.

Con el fin de superar estos desafíos y avanzar hacia un enfoque más eficiente y centralizado, se propone el desarrollo de un sistema web SPA (Single Page Application). Este sistema permitirá la automatización de tareas clínicas y administrativas, mejorando la precisión y accesibilidad de los registros médicos. El objetivo es establecer un estándar para la gestión de datos, facilitando la toma de decisiones y elevando la calidad del servicio ofrecido por la Clínica Veterinaria “Málaga”.



Figura 1 Logo

Fuente: Clínica Veterinaria Málaga (2019)

Para abordar la mejora en el proceso de atención a pacientes de la clínica, se examinaron proyectos similares. A continuación, se destacan algunos trabajos relacionados:

- En el proyecto de Ariana Narcisa Rivas Caicedo y Marlon Alexis Pluas Guaman (2018), se realizó el Diseño e Implementación de una Aplicación Web para Inventario

de Productos y Control de Mascotas para Veterinaria “Multicentro de las Mascotas” utilizando la metodología ágil Scrum y herramientas de modelado web, logrando una alta calidad del software que se encargará de realizar consultas, ingresos de nuevas mascotas e inventario de productos y la facturación correspondiente de cada uno de los servicios.

- Jhovany Amilcar Davila De La Riva (2021) creó un "Sistema web para el registro y control de la veterinaria 'Don Gato'" aplicando la metodología SCRUM, lo que permitió un eficiente manejo de registros y seguimiento de historias clínicas de los pacientes.
- Paola Patricia Buhezo Suyo (2016) se centró en “Mejorar la gestión administrativa del Centro Médico Veterinario 'SANDY’” a través de las TIC. Este proyecto generó reportes sobre clientes, pacientes, servicios realizados y consultas de pacientes, mejorando la eficiencia administrativa.

Estos proyectos sirven como referencia para nuestro objetivo de mejorar la calidad de atención y la gestión eficiente de información en la Clínica Veterinaria “Málaga”.

I.2.2 Justificación del Proyecto

I.2.2.1 Tecnológico

La implementación de un sistema web en la Clínica Veterinaria "Málaga" responde a la necesidad de adoptar tecnologías modernas para mejorar la eficiencia y precisión en la gestión de datos clínicos y administrativos. La elección de tecnologías avanzadas como Python, React JS, y FastAPI permitirá una plataforma robusta y escalable. La utilización de un enfoque de Mapeo Objeto-Relacional (ORM) con SQLAlchemy garantiza una gestión eficiente de la base de datos. Este avance tecnológico no solo optimizará las operaciones internas, sino que también proporcionará una base sólida para futuras expansiones y actualizaciones del sistema.

I.2.2.2 Económico

Desde una perspectiva económica, la implementación del sistema representa una inversión estratégica. La automatización de procesos internos, la mejora en la gestión del inventario y la optimización de la atención al cliente contribuirán a la reducción de costos operativos a

largo plazo. La capacidad del sistema para generar informes y análisis de datos facilitará la toma de decisiones informadas, evitando pérdidas económicas asociadas con la falta de control del inventario. Además, la capacitación efectiva del personal minimizará los costos relacionados con la adaptación al nuevo sistema y garantizará una transición sin contratiempos.

I.2.2.3 Social

En el ámbito social, la implementación del sistema beneficia tanto a los propietarios de mascotas como al personal de la clínica. La mejora en la calidad de atención al cliente, la agilización de procesos y la disponibilidad rápida de información contribuirán a una experiencia más positiva para los propietarios de mascotas. Por otro lado, la capacitación del personal no solo mejora su competencia profesional, sino que también fomenta un ambiente de trabajo más eficiente y colaborativo. Este proyecto no solo busca la eficiencia operativa, sino que también tiene un impacto positivo en la relación entre la clínica y la comunidad a la que sirve.

I.3 Planteamiento del Problema

Dificultades en la gestión de información clínica y administrativa en la Clínica Veterinaria Málaga.

I.4 Marco Teórico

I.4.1 Introducción

El presente capítulo tiene por objetivo recalcar que, en el campo de la atención veterinaria, la atención eficiente y organizada de los servicios es esencial para garantizar la calidad y satisfacción de propietarios y profesionales. Por tanto, la implementación de un sistema de gestión veterinaria se convierte en una herramienta fundamental. Este sistema busca automatizar procesos, centralizar información y optimizar recursos, con el objetivo de mejorar la calidad de atención y facilitar la toma de decisiones basadas en información precisa.

En este trabajo se presenta el desarrollo de sistema de gestión veterinaria adaptado a las necesidades actuales. A través de la identificación de requisitos específicos se busca desarrollar una solución integral y eficiente.

I.4.2 Conceptualización de los Términos Clave

A continuación, se presentará una descripción de algunos conceptos clave:

- **Gestión Veterinaria:** Se refiere al conjunto de actividades, procesos y recursos necesarios para administrar de manera efectiva los servicios y la atención veterinaria. [1].
- **Historia Clínica Electrónica:** Es un registro digital que contiene la información médica y sanitaria, en este caso, de un animal. [2][3].
- **Calendario de citas:** Se refiere al proceso de agendar y organizar citas. [4].
- **Gestión de Inventarios:** Hace referencia al control y administración de los suministros de medicamentos. [5].

I.4.3 Revisión de literatura

- “Diseño y desarrollo de un sistema de gestión para clínicas veterinarias”: P. I. Gómez (2019) se diseñó y desarrolló de un sistema web que les permita a administradores de veterinarias configurar un sitio web informativo, tienda en línea y reservas.

El enfoque del trabajo realizado fue el desarrollo de software y el análisis de la solución y las herramientas utilizadas para lograr las funcionalidades propuestas. Al finalizar el proyecto se completaron los dos de los objetivos propuestos dejando de lado la parte de pagos debido a problemas de implementación.

- “Sistema web de gestión de historiales clínicos veterinarios” V. H. Tórrez (2015) se desarrolló un sistema que gestionará los procesos y procedimientos que intervengan en la generación de información para los historiales clínicos realizados en la clínica veterinaria de Animales S.O.S. el cual fue completado y entregado en su totalidad.
- Sistema web para el registro y control de la veterinaria “Don Gato” J. A. Dávila (2021) se entregó un sistema web completamente funcional, compuesto de una serie de módulos que ayudará a los procesos diarios de la veterinaria, mejorándolos y modernizándolos.

I.4.4 Fundamentos Teóricos

I.4.4.1 Gestión Clínica Veterinaria

- *Principios de la gestión clínica en el ámbito veterinario:* La gestión clínica en medicina veterinaria se fundamenta en la organización eficiente de procesos, desde la atención al paciente hasta la administración de medicamentos, para garantizar una atención integral y de calidad. [6][9].
- *Importancia de la documentación y registro en la atención veterinaria:* La documentación meticulosa y el registro preciso de historias clínicas, citas y medicamentos son pilares esenciales para proporcionar cuidados veterinarios efectivos, facilitar el seguimiento y garantizar la continuidad del tratamiento. [7].
- *Desafíos comunes en la gestión de historias clínicas, citas y medicamentos en clínicas veterinarias:* Las clínicas veterinarias a menudo enfrentan desafíos en la gestión manual de registros, programación de citas y seguimiento de medicamentos, lo que puede resultar en procesos laboriosos y propensos a errores. [7].

I.4.4.2 Metodologías Ágiles

- *Descripción general de metodologías ágiles:* Las metodologías ágiles son enfoques colaborativos y flexibles para el desarrollo de proyectos, destacando por su capacidad para adaptarse a cambios en los requisitos y entregar incrementos de valor de manera constante. Estas metodologías buscan maximizar la satisfacción del cliente a través de una comunicación cercana, equipos auto organizados y una respuesta rápida a los cambios en el entorno del proyecto. [10].

Scrum es una de las metodologías ágiles más ampliamente adoptadas. Se basa en la entrega incremental de productos en cortos períodos llamados "Sprints". Scrum utiliza roles definidos, como el Scrum Master, Product Owner y el Equipo de Desarrollo, para fomentar la colaboración y la toma de decisiones eficaz. Su enfoque en la transparencia, inspección y adaptación continua permite a los equipos responder rápidamente a las necesidades cambiantes del proyecto, promoviendo la entrega de funcionalidades valiosas de manera regular. [11].

- *Ventajas de aplicar metodologías ágiles en proyectos de desarrollo de software:* La aplicación de metodologías ágiles agiliza el desarrollo, fomenta la comunicación

efectiva y garantiza la entrega oportuna de características valiosas, optimizando así la gestión del proyecto. [10].

- *Casos de éxito de implementación de Scrum en proyectos similares:* Scrum ha demostrado su eficacia en proyectos de desarrollo de software, promoviendo la colaboración, transparencia y adaptabilidad, lo que ha llevado a resultados exitosos en diversos contextos.

Por ejemplo, en el proyecto de Ariana Narcisca Rivas Caicedo y Marlon Alexis Plusas Guaman (2018), se realizó el *Diseño e Implementación de una Aplicación Web para Inventario de Productos y Control de Mascotas para Veterinaria “Multicentro de las Mascotas”* utilizando la metodología ágil Scrum y herramientas de modelado web, logrando una alta calidad del software que se encarga de realizar consultas, ingresos de nuevas mascotas e inventario de productos y la facturación correspondiente de cada uno de los servicios.

I.4.4.3 Desarrollo de Software

- *Roles y responsabilidades en el desarrollo de software:* La colaboración entre roles como desarrolladores, testers y diseñadores es esencial para el éxito del desarrollo de software, asegurando una contribución integral y especializada. [12][13].
- *Importancia de la elección de tecnologías adecuadas para un proyecto:* La elección de tecnologías adecuadas, como FastAPI y React, es crucial para garantizar la eficiencia y compatibilidad del sistema desarrollado.
- *Uso de frameworks como FastAPI y React en el desarrollo web:* FastAPI destaca en la creación eficiente de API RESTful, mientras que React proporciona una interfaz de usuario dinámica, combinando tecnologías poderosas para el desarrollo web.

I.4.4.4 ORM (Object-Relational Mapping)

- *Conceptos básicos de ORM:* El ORM actúa como un traductor entre la lógica del programa y la base de datos, simplificando la manipulación de datos al representar entidades de la base de datos como objetos de código. [14]
- *Ventajas de utilizar un ORM en el desarrollo de sistemas:* La utilización de un ORM, como SQLAlchemy, simplifica la interacción con bases de datos, garantizando coherencia, modularidad y mantenimiento efectivo del código. [15].

- *Importancia de la abstracción de la capa de datos y su relación con la eficiencia y seguridad:* La abstracción de la capa de datos proporciona una gestión eficiente y segura de la información, asegurando un desacoplamiento efectivo entre la lógica de la aplicación y la implementación específica de la base de datos. [16].

I.4.4.5 Tecnologías Específicas Utilizadas

- *Python y su aplicación en el desarrollo de backend:* Python sirve como lenguaje principal para la lógica del negocio, destacándose en el desarrollo de backend robusto y eficiente. [17].
- *React y su papel en la creación de interfaces de usuario dinámicas:* React, como framework de JavaScript, ofrece una experiencia de usuario dinámica y atractiva al facilitar la creación de interfaces interactivas y eficientes. [18].
- *FastAPI y su enfoque eficiente en la construcción de API RESTful:* FastAPI se destaca por su velocidad y facilidad de uso, siendo una opción eficiente para la creación de API RESTful. [19].
- *Aplicación de herramientas como Swagger para documentación de API:* Swagger facilita la documentación clara y accesible de la API, mejorando la comprensión y colaboración en el desarrollo del proyecto. [19].

I.4.4.6 Gestión de Datos

- *La importancia de una base de datos en el contexto clínico:* Una base de datos eficiente es esencial para la gestión de información clínica, proporcionando un almacenamiento estructurado y accesible. [20].
- *Elección de SQL como sistema de gestión de base de datos:* La elección de SQL garantiza la consistencia y seguridad en las operaciones de la base de datos, siendo un estándar confiable en la gestión de datos.
- *Funciones y ventajas de SQLAlchemy en la manipulación de datos:* SQLAlchemy facilita la interacción fluida entre entidades de la base de datos y objetos de Python, contribuyendo a la modularidad y mantenimiento efectivo del código. [21].

I.4.4.7 Historias Clínicas

- *Definición y función de las historias clínicas en el contexto veterinario:* Las historias clínicas son registros detallados de la atención médica de los pacientes,

fundamentales para el seguimiento y tratamiento efectivo en el ámbito veterinario. [8].

- *Importancia de un registro detallado para el seguimiento y tratamiento de pacientes:* Un registro detallado en las historias clínicas es esencial para garantizar un seguimiento preciso y una atención continua y efectiva. [7][8].
- *Desafíos comunes en la gestión manual de historias clínicas y cómo la tecnología puede abordarlos:* La gestión manual de historias clínicas puede ser propensa a errores y laboriosa; la tecnología proporciona soluciones para mejorar la eficiencia y precisión en la gestión. [7].

I.4.4.8 Seguimiento de Pacientes

- *Concepto de seguimiento de pacientes en clínicas veterinarias:* El seguimiento de pacientes implica un monitoreo continuo de tratamientos y citas, asegurando una atención integral y personalizada. [6].
- *Ventajas de un sistema automatizado para el seguimiento de tratamientos y citas:* La automatización en el seguimiento de tratamientos y citas mejora la eficiencia y reduce posibles errores en la atención, proporcionando beneficios tanto al personal como a los clientes. [22].
- *Mejoras esperadas en la calidad de la atención a través de un seguimiento más eficiente:* Un seguimiento más eficiente contribuye directamente a una atención de mayor calidad, mejorando la experiencia global de pacientes y propietarios. [7].

I.4.4.9 Tipos de Mascotas

- *Consideraciones específicas al gestionar información sobre diferentes tipos de mascotas:* La gestión de datos debe adaptarse a las necesidades particulares de diversos animales, considerando diferencias en especies y razas. [8].
- *Personalización en la gestión de datos según las necesidades específicas de cada tipo de animal:* Un sistema flexible permite la personalización de datos según las necesidades de cada tipo de mascota, asegurando un enfoque específico y completo. [8].
- *Relevancia de un sistema flexible que se adapte a distintos casos, como perros y gatos:* La flexibilidad del sistema es crucial para abordar la diversidad en el cuidado

de mascotas, desde perros y gatos hasta otras especies, garantizando una gestión adaptativa y completa. [7].

I.4.5 Metodologías y Enfoques Técnicos:

I.4.5.1 Metodologías

I.4.5.1.1 SCRUM

Según lo que se encuentra en "La Guía Definitiva de Scrum: Las Reglas del Juego" de Ken Schwaber y Jeff Sutherland (2020), Scrum se puede definir como un marco de trabajo ágil que brinda apoyo a personas, equipos y organizaciones en la generación de valor a través de soluciones adaptables para problemas complejos.

En resumen, Scrum requiere la presencia de un Scrum Master para fomentar un entorno en el que:

1. Un Product Owner ordene el trabajo relacionado con un problema complejo en un Product Backlog.
2. El Scrum Team convierta una selección de ese trabajo en un Incremento de valor durante un Sprint.
3. El Scrum Team y sus interesados inspeccionen los resultados y se adapten para el próximo Sprint.

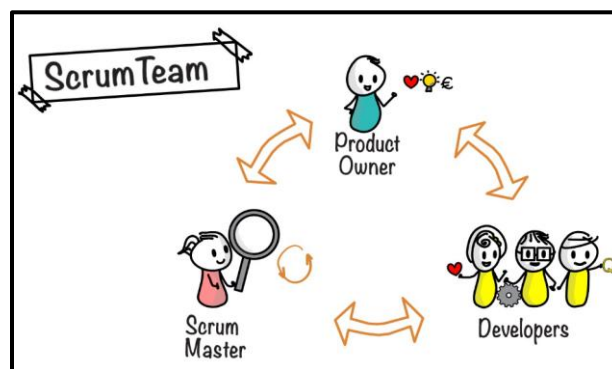


Figura 2 Scrum team
Fuente: ITDO

El marco de Scrum se basa en la inteligencia colectiva de las personas que lo utilizan. En lugar de proporcionar instrucciones detalladas, las reglas de Scrum guían las relaciones e interacciones entre los participantes.

Scrum se basa en los principios del empirismo y el pensamiento Lean. El empirismo sostiene que el conocimiento proviene de la experiencia y la toma de decisiones basada en la observación. El pensamiento Lean se centra en reducir el desperdicio y se enfoca en lo esencial.

Emplea un enfoque iterativo e incremental para optimizar la previsibilidad y controlar el riesgo. Implica a grupos de personas con todas las habilidades y experiencia necesarias para llevar a cabo el trabajo, permitiendo compartir o adquirir esas habilidades según sea necesario.

Scrum combina cuatro eventos formales para inspección y adaptación en un evento contenedor llamado Sprint. Estos eventos funcionan porque implementan los pilares empíricos de Scrum: transparencia, inspección y adaptación.

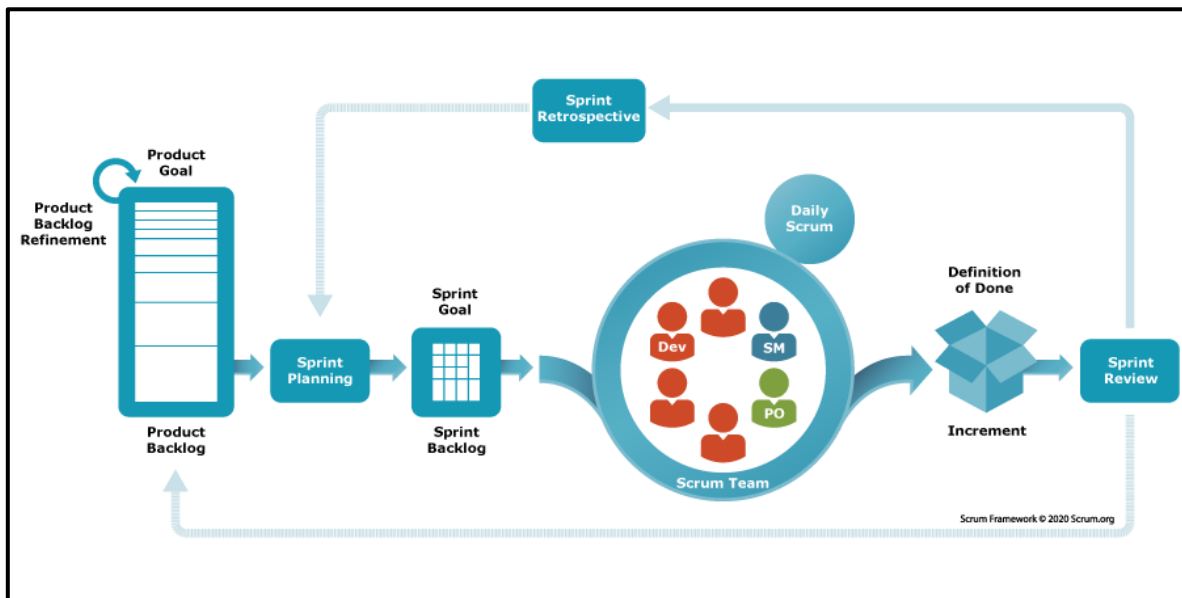


Figura 3 Scrum

Fuente: Scrum Framework (2020)

I.4.5.1.1.1 Eventos de Scrum

El Sprint es el contenedor para todos los demás eventos de Scrum. Cada evento en Scrum ofrece una oportunidad formal para inspeccionar y adaptar los artefactos Scrum, y están diseñados específicamente para habilitar la transparencia requerida. No llevar a cabo ningún evento según lo prescrito resulta en la pérdida de oportunidades para inspeccionar y adaptarse.

El Sprint, que dura un mes o menos, es el corazón de Scrum, donde las ideas se convierten en valor. Un nuevo Sprint comienza inmediatamente después de la conclusión del Sprint anterior. Durante un Sprint, no se realizan cambios que pongan en peligro el Objetivo del Sprint, la calidad no disminuye, el Product Backlog se refina según sea necesario, y el alcance se puede aclarar y renegociar con el Product Owner. Los Sprints permiten la previsibilidad al garantizar la inspección y adaptación del progreso hacia un Objetivo del Producto al menos una vez al mes.

I.4.5.1.1.1 Sprint

Son eventos de duración fija de un mes o menos para crear consistencia. Un nuevo Sprint comienza inmediatamente después de la conclusión del Sprint anterior. Todo el trabajo necesario para lograr el Objetivo del Producto, incluido la Sprint Planning, Daily Scrums, Sprint Reviews y Sprint Retrospective, ocurre dentro de los Sprints.

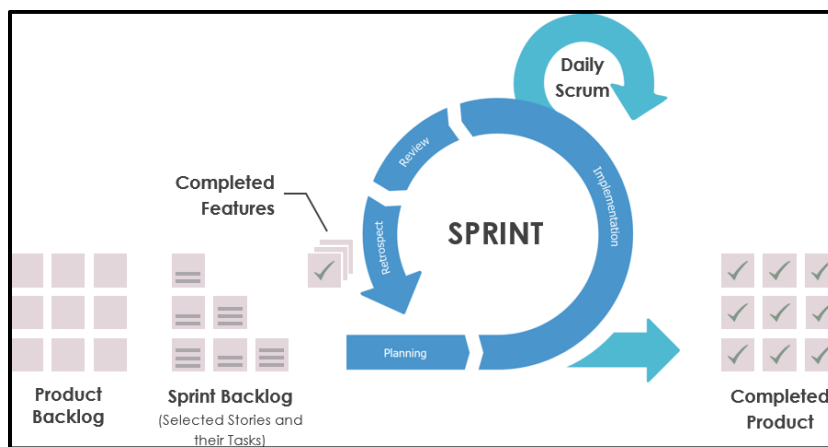


Figura 4 Sprint

Fuente: Cybermedian

Durante el Sprint:

- No se realizan cambios que pongan en peligro el Objetivo del Sprint
- La calidad no disminuye
- El Product Backlog se refina según sea necesario
- El alcance se puede aclarar y renegociar con el Product Owner

Los Sprints permiten la previsibilidad al garantizar la inspección y adaptación del progreso hacia un Objetivo del Producto al menos una vez al mes.

Cuando el horizonte de un Sprint es demasiado largo, el Objetivo del Sprint puede volverse inválido, la complejidad puede crecer y el riesgo puede aumentar. Se pueden emplear Sprints más cortos para generar más ciclos de aprendizaje y limitar el riesgo de costo y esfuerzo a un período de tiempo menor. Cada Sprint puede considerarse un proyecto corto.

Un Sprint podría cancelarse si el Objetivo del Sprint se vuelve obsoleto. Solo el Product Owner tiene la autoridad para cancelar el Sprint.

I.4.5.1.1.1.2 Sprint Planning

La Sprint Planning inicia el Sprint al establecer el trabajo que se realizará para el Sprint. El Scrum Team crea este plan resultante mediante trabajo colaborativo. El Product Owner se asegura de que los asistentes estén preparados para discutir los elementos más importantes del Product Backlog y cómo se relacionan con el Objetivo del Producto.

El Objetivo del Sprint, los elementos del Product Backlog seleccionados para el Sprint, más el plan para entregarlos se denominan juntos Sprint Backlog.

La Sprint Planning tiene un límite de tiempo de máximo ocho horas para un Sprint de un mes. Para Sprints más cortos, el evento suele ser de menor duración.

I.4.5.1.1.1.3 Daily Scrum

El propósito de el Daily Scrum es inspeccionar el progreso hacia el Objetivo del Sprint y adaptar el Sprint Backlog según sea necesario, ajustando el trabajo planificado.

El Daily Scrum es un evento de 15 minutos para los Developers del Scrum Team. Para reducir la complejidad, se lleva a cabo a la misma hora y en el mismo lugar todos los días hábiles del Sprint. Si el Product Owner o Scrum Master están trabajando activamente en elementos del Sprint Backlog, participan como Developers.

Los Developers pueden seleccionar la estructura y las técnicas que deseen, siempre que su Daily Scrum se centre en el progreso hacia el Objetivo del Sprint y produzca un plan viable para el siguiente día de trabajo. Esto crea enfoque y mejora la autogestión.

Los Daily Scrums mejoran la comunicación, identifican impedimentos, promueven la toma rápida de decisiones y, en consecuencia, eliminan la necesidad de otras reuniones.

El Daily Scrum no es el único momento en el que los Developers pueden ajustar su plan. A menudo se reúnen durante el día para discusiones más detalladas sobre cómo adaptar o volver a planificar el resto del trabajo del Sprint.

I.4.5.1.1.1.4 Sprint Review

El propósito de la Sprint Review es inspeccionar el resultado del Sprint y determinar futuras adaptaciones. El Scrum Team presenta los resultados de su trabajo a los interesados clave y se discute el progreso hacia el Objetivo del Producto.

Durante el evento, el Scrum Team y los interesados revisan lo que se logró en el Sprint y lo que ha cambiado en su entorno. El Product Backlog también se puede ajustar para satisfacer nuevas oportunidades. La Sprint Review es una sesión de trabajo y el Scrum Team debe evitar limitarla a una presentación.

La Sprint Review es el penúltimo evento del Sprint y tiene un límite de tiempo de máximo cuatro horas para un Sprint de un mes. Para Sprints más cortos, el evento suele ser de menor duración.

I.4.5.1.1.1.5 Sprint Retrospective

El propósito de la Sprint Retrospective es planificar formas de aumentar la calidad y la efectividad.

El Scrum Team inspecciona cómo fue el último Sprint con respecto a las personas, las interacciones, los procesos, las herramientas y su definición de terminado. Los elementos inspeccionados suelen variar según el ámbito del trabajo. Se identifican los supuestos que los llevaron por mal camino y se exploran sus orígenes. El Scrum Team analiza qué salió bien durante el Sprint, qué problemas encontró y cómo se resolvieron (o no) esos problemas.

El Scrum Team identifica los cambios más útiles para mejorar su efectividad. Las mejoras más impactantes se abordan lo antes posible. Incluso se pueden agregar al Sprint Backlog para el próximo Sprint.

El Sprint Retrospective concluye el Sprint. Tiene un tiempo limitado a máximo tres horas para un Sprint de un mes.

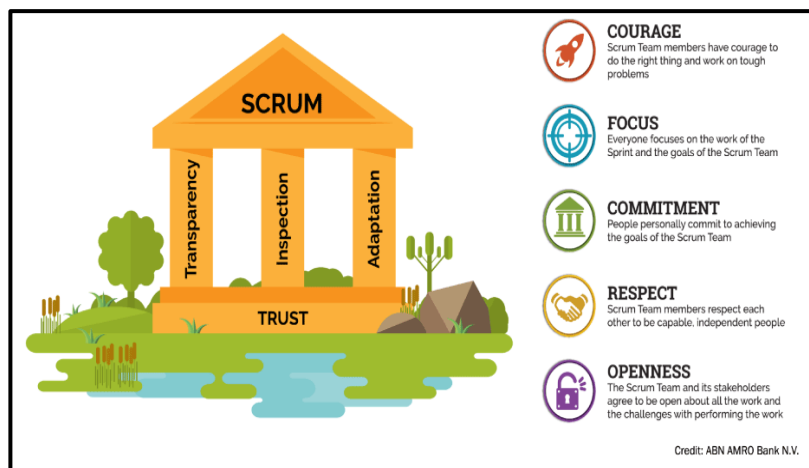


Figura 5 Scrum values

Fuente: ABN AMRO() Bank N.V. (2020)

I.4.5.1.1.2 Artefactos de Scrum

Los artefactos de Scrum representan trabajo o valor. Están diseñados para maximizar la transparencia de la información clave. Por lo tanto, todas las personas que los inspeccionan tienen la misma base de adaptación.

Cada artefacto contiene un compromiso para garantizar que proporcione información que mejore la transparencia y el enfoque frente al cual se pueda medir el progreso:

- Para el Product Backlog, es el Objetivo del Producto.
- Para el Sprint Backlog, es el Objetivo del Sprint.
- Para el Increment, es la Definición de Terminado.

I.4.5.1.1.2.1 Product Backlog

El Product Backlog es una lista emergente y ordenada de lo que se necesita para mejorar el producto. Es la única fuente del trabajo realizado por el Scrum Team.

El refinamiento del Product Backlog es el acto de dividir y definir aún más los elementos del Product Backlog en elementos más pequeños y precisos. Esta es una actividad continua para agregar detalles, como una descripción, orden y tamaño. Los atributos suelen variar según el ámbito del trabajo.

Los Developers que realizarán el trabajo son responsables del dimensionamiento. El Product Owner puede influir en los Developers ayudándolos a entender y seleccionar sus mejores alternativas.

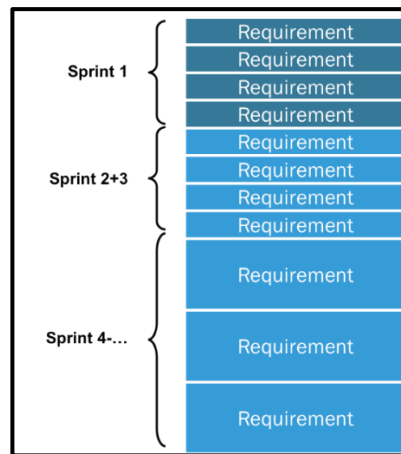


Figura 6 Product Backlog

Fuente: Scrum.org

I.4.5.1.1.2.2 Sprint Backlog

El Sprint Backlog se compone del Objetivo del Sprint (por qué), el conjunto de elementos del Product Backlog seleccionados para el Sprint (qué), así como un plan de acción para entregar el Increment (cómo).

El Sprint Backlog es un plan realizado por y para los Developers. Es una imagen muy visible y en tiempo real del trabajo que los Developers planean realizar durante el Sprint para lograr el Objetivo del Sprint.

En consecuencia, el Sprint Backlog se actualiza a lo largo del Sprint a medida que se aprende más. Debe tener suficientes detalles para que puedan inspeccionar su progreso en la Daily Scrum.

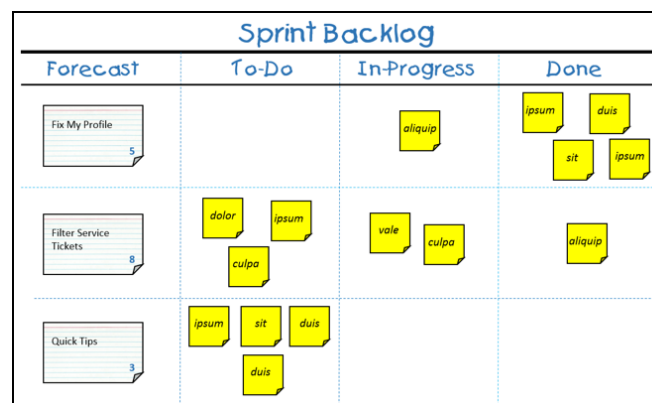


Figura 7 Sprint Backlog progress

Fuente: Scrum

I.4.5.1.1.2.3 Increment

Un Increment es un peldaño concreto hacia el Objetivo del Producto. Cada Increment se suma a todos los Increments anteriores y se verifica minuciosamente, lo que garantiza que todos los Increments funcionen juntos.

Para proporcionar valor, el Increment debe ser utilizable. Se pueden crear múltiples Increments dentro de un Sprint. La suma de los Increments se presenta en la Sprint Review apoyando así el empirismo.

Sin embargo, se puede entregar un Increment a los interesados antes del final del Sprint. El trabajo no puede considerarse parte de un Increment a menos que cumpla con la definición de Terminado.

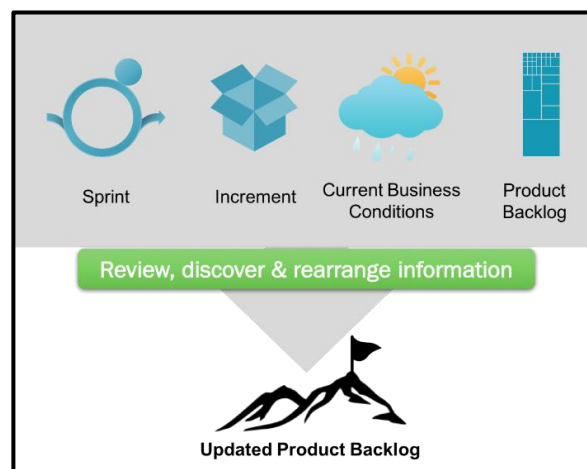


Figura 8 Product Backlog

Fuente: Scrum

Finalmente, se decidió trabajar con la metodología ágil Scrum debido a su enfoque flexible y su adaptabilidad a cambios en los requisitos, permitiendo ajustes y entregas frecuentes según las peticiones del cliente. Por otro lado, facilita al equipo de desarrollo (mi persona) en la toma de decisiones en los procesos y el seguimiento de los mismos. Además, se hizo uso intensivo de la plataforma Git, la cual fue una herramienta esencial para la gestión eficiente del código fuente y la colaboración en equipo (revisiones con el tutor). Adicionalmente se emplearon las issues board de Git para el desarrollo del back-end, facilitando una visualización clara de las tareas pendientes, en progreso y completadas.

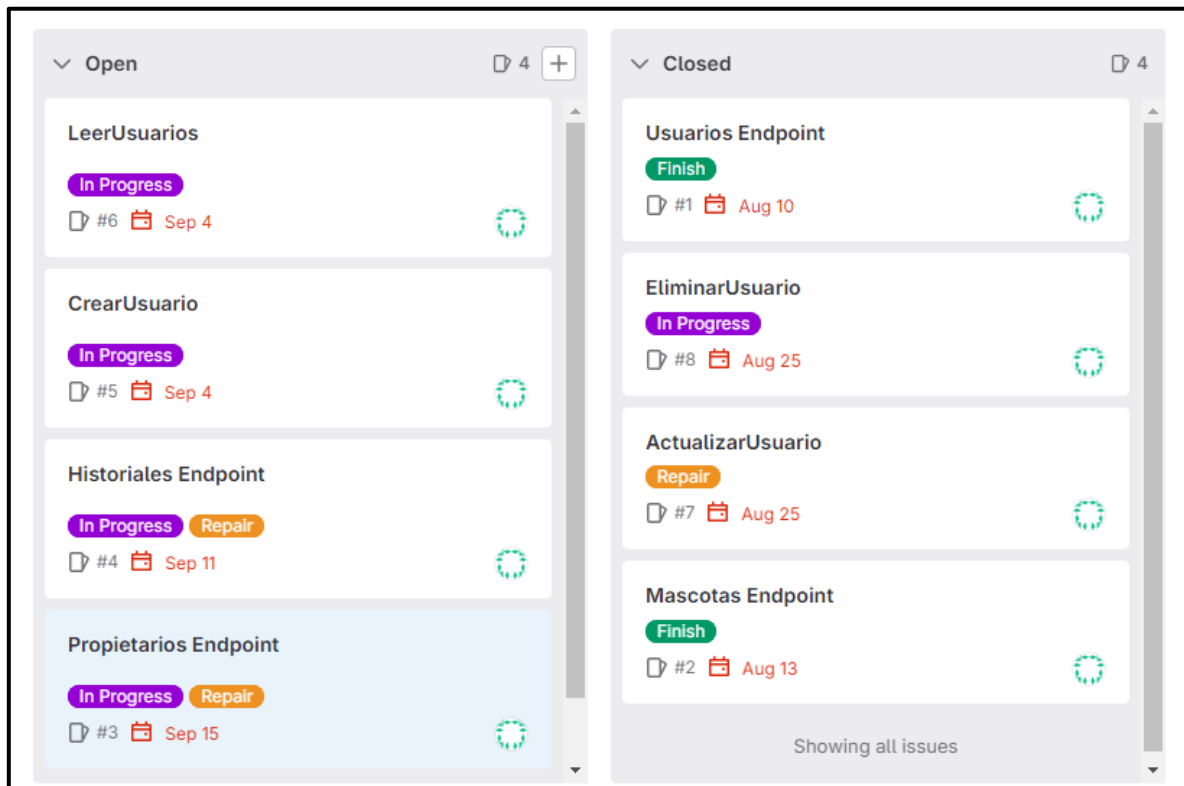


Figura 9: Issues Board

Fuente: Elaboración propia

I.4.5.1.2 UML (Lenguaje Unificado de Modelado)

La elección de UML como herramienta para el modelado del desarrollo de este proyecto se basa en su notación gráfica clara y comprensible, y su versatilidad para modelar diferentes aspectos del sistema, tales como casos de uso, diagramas de clases y de actividades.

UML proporciona una representación visual unificada y holística de un sistema, programa, software o incluso un proceso. En su mayoría, los desarrolladores crean un diagrama UML para que los usuarios finales o los desarrolladores comerciales conozcan el funcionamiento del sistema. Considérelo como un modelo del sistema, con reglas y notaciones preestablecidas para todo.

Por lo tanto, al observar el diagrama UML, podemos comprender cómo funciona el sistema. Aparte de eso, también puede ayudarnos a resolver problemas y tomar decisiones orientadas a resultados. Estos diagramas están relacionados con la programación y los principios

orientados a objetos. Dado que están inspirados en entidades y problemas del mundo real, también facilita que los usuarios los entiendan. [23].

I.4.5.1.2.1 Tipos de diagramas

UML Existen diferentes tipos de diagramas UML con los que puede trabajar en función de sus requisitos. Los tipos de diagramas UML se pueden distinguir como estructurales o de comportamiento. [23].

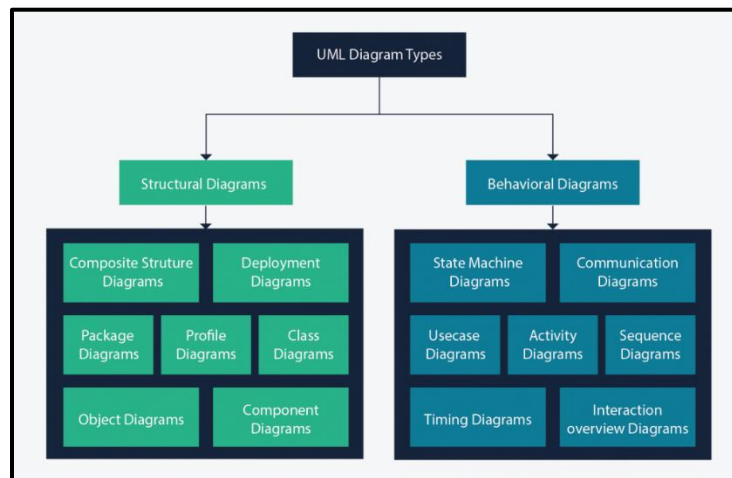


Figura 10 Diagramas UML

Fuente: edrawsoft

I.4.5.1.2.1.1 Diagramas UML Estructurales

Como sugiere el nombre, representan la estructura general de un sistema: sus objetos, atributos y relaciones entre diferentes entidades. Es decir, nos preocupa más de qué se trata el sistema que de cómo se comporta. [23].

- Diagrama de clase
- Diagrama de componentes
- Diagrama de implementación
- Diagrama de objetos
- Diagrama del paquete
- Diagrama de perfil
- Diagrama de estructura compuesta

I.4.5.1.2.1.2 Diagramas UML de comportamiento

Estos diagramas definen el comportamiento del sistema y la funcionalidad general de cada unidad. Además, algunos de estos diagramas también representan el flujo de información/control en el sistema. [23][24].

- Diagrama de máquina de estado
- Diagrama de actividades
- Diagrama de casos de uso
- Diagrama secuencial
- Diagrama de comunicación
- Diagrama de tiempo
- Diagrama de descripción general interactivo

I.4.5.2 Enfoques Tecnológicos

I.4.5.2.1 Sistema Web SPA (Single Page Application)

Una web SPA o single page application se refiere a una forma de desarrollo web en la que la página web está contenida en un único archivo. De esta forma, se carga todo en HTML y, mientras navegamos por la página, irá solicitando el contenido al servidor via ajax. Es decir, todo el contenido html, css y javascript de la página web al completo se carga una primera vez y nada más. De esta forma se mejoran los tiempos de respuesta y, por consiguiente, la experiencia de usuario. [25].



Figura 11: SPA

Fuente: Qué es una SPA?.omatech.com

I.4.5.2.1.1 Diferencias entre Multipage Applications y Single Page Applications

Hasta ahora, lo normal a la hora de desarrollar una web, era diseñarla como una multipage application (MPA o también conocido como page redraw).

Una MPA consiste en que cada vez que visitemos una página específica de una web se cargará desde cero. Mientras que, en las SPA, al visitar la página por primera vez se cargará completamente su estructura y, cada vez que pinchemos en el enlace, sólo solicitaremos al servidor el contenido ya que html, css y javascript están ya cargados.

La gran diferencia, entonces, entre una multipage application y una single page application es a la hora de navegar por la página web. Mientras que, con una SPA, por cada enlace dentro de la web solo se cargará el contenido, que es lo que nos envía el servidor vía ajax; con una MPA se cargará todo desde cero. [26].

I.4.5.2.1.2 Varias vistas en lugar de varias páginas

En una SPA tenemos una única página, pero contaremos con diferentes vistas. Cuando hablamos de vistas, nos referimos a los distintos apartados de nuestra web, es decir, como una pantalla en una aplicación de escritorio. Así que, en esta misma página, se irán intercambiando distintas vistas, dando la sensación de que visitamos varias páginas, aunque en realidad estemos todo el rato ante la misma página, pero con diferente vista. Un ejemplo sencillo sería nuestro servidor de correo, que cuenta con una página de login pero una vez dentro, cada vez que cargamos un correo distinto será distinta vista pero la misma página.

De todos modos, aunque sea la misma página, la url podrá cambiar sin problemas según cambiemos de vista/página, ya que la clave para que sea una SPA es que esta vista/página nunca se recargará. Es más, es muy importante, de cara a tener una estructura clara de la web, que las url cambien también, aunque se haya desarrollado con una SPA, para poder navegar de forma más eficiente por la misma. [26].

I.4.5.2.1.3 Lenguaje de programación de las páginas web SPA

Una SPA se creará siempre en Javascript. Ya que, al ser una aplicación web ejecutada del lado del cliente, no hay otro lenguaje que pueda hacer eso. A esto habrá que sumarle, lógicamente, el HTML y CSS.

Dentro de Javascript, contaremos con multitud de librerías y frameworks que nos facilitarán el desarrollo de las SPA. Algunas de las más usadas son:

- AngularJS
- Angular
- VUE
- React
- EmberJS
- Polymer

Este listado son las librerías y frameworks más comunes, aunque hay muchos otros. Escoger una librería y un framework dependerá, básicamente, de tus gustos, la experiencia que tengas con los mismos, el tamaño de la aplicación a desarrollar.

De todos modos, esto sólo aplica al front-end. Si queremos ampliar la pregunta al lado del back-end, es decir, del servidor, nos valdría cualquier lenguaje de back-end para producir la parte del servidor. Lo que tendríamos que hacer es crear un API REST que devuelva el JSON necesario para alimentar de datos a la SPA. Es decir, la SPA nos dará igual cómo esté desarrollada del lado del back-end, es totalmente independiente a este. [25].

I.4.5.2.1.4 Ventajas de las páginas web SPA

Desarrollar una página web SPA tiene muchas ventajas frente al MPA. Algunas de ellas son:

- Se tratan de aplicaciones fáciles de desarrollar, desplegar y depurar. Al contar con infinitos frameworks y librerías, será muy sencillo de desarrollar.
- Nos muestran el contenido de forma sencilla y elegante al cargar todo el contenido en una única página.
- Además, ejecutando la lógica del lado del navegador, hace que las interacciones del usuario se ejecuten más rápido.
- La analítica de la web es mucho más sencilla puesto que solo analiza una única página. [25].

I.4.5.2.1.5 Desventajas de las páginas web SPA

- En sitios muy grandes, el mantenimiento del código puede ser algo complicado y hay que tener desde el principio muy clara la estructura de la página.
- Al tener el código fuente directamente en el navegador, si no seguimos buenas prácticas, la seguridad del sitio se podría ver expuesta.
- Aunque las SPA son muy rápidas, al cargar la página por primera vez de forma completa, esto hace que esta carga sea algo lenta, lo que puede ahuyentar posibles clientes. De todos modos, actualmente, hay métodos para mejorar esta velocidad de carga. [25].

La elección de una SPA se fundamenta en su capacidad para cargar todo el contenido en una única página web, mejorando significativamente los tiempos de respuesta y la experiencia del usuario. Además, se aprovechó la característica de tener varias vistas en una única página, facilitando la navegación entre secciones sin recargar la página. La implementación de JavaScript fue esencial, utilizando librerías como React para crear interfaces dinámicas y elegantes.

I.4.5.2.2 FastAPI

Es un framework basado en Python. Ha llamado mucho la atención en los últimos años. Su estructura de desarrollo es similar a Flask, que fue la opción más popular para aquellos que querían aprender el desarrollo web de Python.

FastAPI es fácil de usar y su documentación es clara y fácil de seguir; es utilizado por gigantes tecnológicos como Microsoft, Netflix y Uber.

El propósito principal de FastAPI es construir endpoints API, como el nombre sugiere. Esto se puede hacer tan eficientemente como regresar diccionarios de Python, datos en JSON o utilizando la interfaz de usuario Swagger interactiva de OpenAPI.

FastAPI no se limita a las API. Se puede usar para casi todo lo que hace un marco web, desde entregar páginas web con el motor de plantillas Jinja2 hasta servir aplicaciones basadas en WebSockets. [27].



Figura 12 FastAPI

Fuente: <https://fastapi.tiangolo.com/es/>

En el desarrollo de este proyecto, se eligió FastAPI para la construcción de los endpoints API, destacando su facilidad de uso y documentación clara.

I.4.5.2.3 React

Es una librería open source de JavaScript para desarrollar interfaces de usuario. Fue lanzada en el año 2013 y desarrollada por Facebook, quienes también la mantienen actualmente junto a una comunidad de desarrolladores independientes y compañías.

Hoy en día muchas empresas de primer nivel utilizan React para el desarrollo de sus aplicaciones, y es que entre ellas podemos encontrar Facebook, Instagram y el cliente web de WhatsApp (todas propiedad de Facebook), y otras como Airbnb, Uber, Netflix, Twitter, Reddit o Paypal.

Desde su lanzamiento, su uso ha ido incrementando notablemente, convirtiéndose, a día de hoy, en una de las tecnologías front-end más utilizadas. [28].



Figura 13 React

Fuente: <https://www.react.org/>

Se utilizó React en el frontend del proyecto aprovechando al máximo los componentes reutilizables que ofrece para construir una interfaz de usuario interactiva y eficiente.

I.4.5.2.4 Vite

Vite es una herramienta de compilación que tiene como objetivo proporcionar una experiencia de desarrollo más rápida y ágil para proyectos web modernos. Consta de dos partes principales:

- Un servidor de desarrollo que proporciona mejoras enriquecidas de funcionalidades sobre módulos ES nativos.
- Un comando de compilación que empaqueta tu código con Rollup, preconfigurado para generar recursos estáticos altamente optimizados para producción. Vite es dogmático y viene con configuraciones predeterminadas listas para usar. [29].

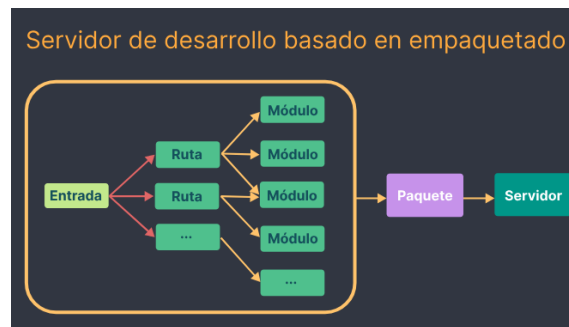


Figura 14 Vite

Fuente: <https://vitejs.dev/>

Al integrar Vite como herramienta de compilación en el desarrollo del proyecto, se busca lograr una experiencia de desarrollo más rápida y ágil para proyectos web modernos.

I.4.5.2.5 Linter

En programación, un linter se entiende como una herramienta a cargo de la mejora del código a través del análisis del código fuente, con el objetivo de identificar fallos o problemas en el sistema.

Estas inconsistencias pueden presentarse en situaciones como exámenes de análisis de código, así como debuggeo, entre otras opciones.

En lo que respecta al origen de un linter en programación, es importante destacar que esta opción proviene de un antiguo recurso implementado en el lenguaje de programación C,

denominado lint, que se encargaba de las actividades del código fuente, al tiempo que verificaba que se encontrara optimizado antes de su compilación. [30].

Para el desarrollo de este proyecto se está utilizando Eslint y Prettier como linters principales.

I.4.5.2.6 Tailwind

Tailwind, en pocas palabras, es un framework CSS que da prioridad a la utilidad sobre el propio estilo, pero además a diferencia de otros frameworks CSS como Bootstrap o Bulma, Tailwind no provee una serie de componentes predefinidos.

En su lugar, este framework opera en un nivel inferior y te proporciona un conjunto de clases de ayuda para estructura y estilado, de forma que, usando dichas clases, puedas crear rápidamente diseños personalizados con facilidad. [31].

Además, no es opinionado y gracias a su flexibilidad te permite crear un diseño realmente único.

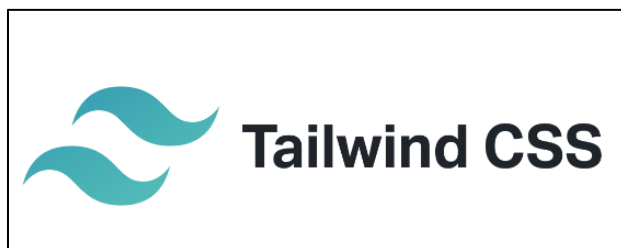


Figura 15 TailwindCSS

Fuente: <https://tailwindcss.com/>

Al adoptar Tailwind como framework CSS, se prioriza la utilidad sobre el estilo, lo que brindó una mayor flexibilidad en el diseño. Las clases de ayuda proporcionadas por Tailwind para estructurar y estilizar las páginas de manera eficiente, facilitó la creación de diseños personalizados de forma rápida y única, adaptándose así a las necesidades específicas del proyecto.

I.4.5.2.7 Uvicorn

Es un servidor web rápido y ligero para aplicaciones Python. Está diseñado específicamente para trabajar con frameworks web asincrónicos, como FastAPI, y es ampliamente utilizado en el ecosistema de Python para ejecutar aplicaciones web de alto rendimiento. [32].



Figura 16 Uvicorn

Fuente: <https://www.uvicorn.org/>

Para ejecutar aplicaciones Python de alto rendimiento, se utilizó Uvicorn como el servidor web del proyecto. Diseñado específicamente para trabajar con frameworks web asincrónicos como FastAPI, Uvicorn desempeñó un papel crucial en la ejecución eficiente de la aplicación.

I.4.5.2.8 Swagger

Cuando hablamos de Swagger nos referimos a una serie de reglas, especificaciones y herramientas que nos ayudan a documentar nuestras APIs. De esta manera, podemos realizar documentación que sea realmente útil para las personas que la necesitan. Swagger nos ayuda a crear documentación que todo el mundo entienda. [33].



Figura 17 Swagger

Fuente: <https://swagger.io/>

Se optó por la implementación de Swagger, aprovechando sus reglas, especificaciones y herramientas para crear documentación clara y comprensible. Swagger resultó esencial para comunicar eficazmente las características y funcionalidades de las APIs, facilitando así su comprensión y uso tanto por parte de desarrolladores como de otros interesados. Este enfoque se considera crucial para futuras interacciones.

I.4.5.2.9 SQLAlchemy

SQLAlchemy es un SQL toolkit de código abierto para interactuar con bases de datos desde Python, sino que también se integra de manera fluida en la arquitectura Onion. Situada en la capa de infraestructura, SQLAlchemy, al ser un ORM, simplifica la relación entre entidades de la base de datos y objetos Python. En este modelo arquitectónico, esta capa interna se encarga de las operaciones directas con la base de datos, ofreciendo un desacoplamiento eficiente y permitiendo que las capas superiores se beneficien de sus funcionalidades sin conocer los detalles específicos de la implementación de la base de datos. Este enfoque asegura una modularidad y mantenimiento eficaz del código a lo largo del desarrollo del proyecto.



Figura 18 SQLAlchemy

Fuente: <https://www.sqlalchemy.org/>

A este tipo de librerías se las conoce como ORM (en inglés Object-Relational Mapping).

La finalidad de un ORM es mapear (relacionar) entidades de una base de datos y objetos (p.e clases de Python) de un determinado lenguaje de programación con tablas de una base de datos.

Son muy comunes en los paradigmas de programación orientada a objetos, ya que hace uso de ellos para ofrecer una abstracción y una manera de interactuar con las bases de datos en un manera Pythonica sin necesidad de utilizar directamente queries de SQL para ello. Se encargará de realizar las operaciones en el motor de sql de manera transparente para el usuario. [34].

La elección de SQLAlchemy como ORM facilitó la abstracción de operaciones con la base de datos y mejoró la eficiencia en el manejo de datos.

I.4.5.2.10 Python

Python es un lenguaje de programación ampliamente reconocido y utilizado en diversos campos, como aplicaciones web, desarrollo de software, ciencia de datos y el machine learning (ML).

Una de las practicas destacadas al utilizar Python para el desarrollo del sistema fue la implementación de entornos virtuales como *venv* (*Virtual Environment*). Los entornos virtuales son espacios aislados que permiten gestionar las dependencias y bibliotecas específicas de un proyecto, garantizando que no haya conflicto con otras aplicaciones o proyectos que puedan estar en el mismo entorno. Esta práctica resulta especialmente beneficiosa, ya que facilita la reproductibilidad del entorno de desarrollo en diferentes sistemas y colaboraciones. Al utilizar entornos virtuales con Python, se asegura una consistencia en las versiones de las bibliotecas y facilita el control preciso de las dependencias del proyecto, contribuyendo así a la eficiencia y estabilidad durante el desarrollo del sistema. [35].



Figura 19 Python

Fuente: <https://www.python.org/>

La elección de Python como lenguaje principal para el desarrollo del sistema se basó en su sintaxis clara y legible, así como en su versatilidad y flexibilidad, destacando su sólido soporte para la integración de sistemas y APIs. Se implementó entornos virtuales, utilizando específicamente *venv*, para gestionar de forma aislada las dependencias y bibliotecas. Esta práctica contribuyó a la reproductibilidad del entorno de desarrollo, asegurando consistencia en las versiones de bibliotecas y facilitando el control preciso de las dependencias del proyecto.

I.4.5.2.11 JavaScript

JavaScript es un lenguaje de programación esencial para el desarrollo de páginas web interactivas. Los desarrolladores lo utilizan para mejorar la experiencia del usuario al

actualizar dinámicamente el contenido, como fuentes de redes sociales, animaciones y mapas interactivos.

Este lenguaje, tanto del lado del cliente como del servidor, desempeña un papel crucial en la World Wide Web. Por ejemplo, cuando encuentras funcionalidades como carruseles de imágenes o menús desplegados “click-to-show” (clic para mostrar) en la web, estás experimentando los efectos de JavaScript. Su versatilidad y capacidad para realizar acciones en tiempo real hacen que sea una herramienta esencial para el desarrollo web moderno. [36].

JavaScript también es fundamental en la implementación y manejo de JSON Web Tokens (JWT), una tecnología esencial para la seguridad en las comunicaciones web, Los JWT son como pequeños paquetes seguros de información que permiten a los diferentes componentes de un sistema comunicarse de manera segura y autenticada. En términos sencillos, cuando inicias sesión en una aplicación y ves que permaneces autenticado incluso al recargar la página, es probable que detrás de esa experiencia transparente esté trabajando un JWT, garantizando la integridad y seguridad de tus datos en la web. [37].



Figura 20 JavaScript

Fuente: <https://javascript.com/>

La inclusión de JavaScript en el proyecto fue fundamental para mejorar la experiencia del usuario en páginas web interactivas. Se utilizó JavaScript tanto en el lado del cliente como del servidor para dinamizar el contenido, proporcionando funciones como actualizaciones dinámicas y elementos interactivos. Además, JavaScript desempeñó un papel esencial en la implementación y gestión de JSON Web Tokens (JWT), garantizando la seguridad y autenticación en las comunicaciones web.

I.4.5.2.12 Figma

Figma es una plataforma de edición gráfica y diseño de interfaces. Además, es una plataforma online y colaborativa.

Con Figma se puede hacer un poco de todo a nivel de diseño gráfico, desde diseñar páginas web e interfaces gráficas de aplicaciones, o crear publicaciones para redes sociales, hasta la posibilidad de poder crear presentaciones. [38].



Figura 21 Figma

Fuente: <https://www.figma.com/>

Por este motivo, es una de las herramientas más valoradas por empresas y estudios de diseño gráfico. Se aprovechó las capacidades de Figma para crear de manera eficiente y colaborativa elementos gráficos, desde páginas web hasta presentaciones, asegurando coherencia en el diseño y facilitando la comunicación visual en el desarrollo del proyecto.

I.4.6 Marco conceptual

Los elementos clave del marco conceptual para el desarrollo del presente proyecto son los siguientes:

I.4.6.1 Arquitectura del Sistema

El sistema de gestión se basará en una arquitectura cliente-servidor, donde el usuario accederá a través de una interfaz web y los datos se almacenarán en una base de datos centralizada. Los componentes principales incluirán agenda, usuarios, propietarios, historial, medicamentos y dashboard.

I.4.6.2 Funcionalidades del Sistema

El sistema permitirá la gestión de la clínica veterinaria, incluyendo la programación de citas realizadas por el médico, registro de propietarios y mascotas, gestión de historias clínicas, seguimiento de tratamientos y generación de informes. Además, se implementarán funcionalidades específicas como gestión de inventario de medicamentos veterinarios.

I.4.6.3 Flujo de Trabajo en la Clínica Veterinaria

El sistema seguirá un flujo de trabajo estructurado, desde la llegada del paciente hasta la finalización del tratamiento. Los pasos clave incluirán la recepción del paciente, el registro de información relevante, la asignación de citas (en caso de ser requerido por el propietario) y la actualización de registros de tratamientos y diagnósticos.

A continuación, se muestra un flujograma de pasos que se sigue para realizar una atención clínica desde el momento en que el cliente ingresa a la Clínica con su mascota o mascotas:

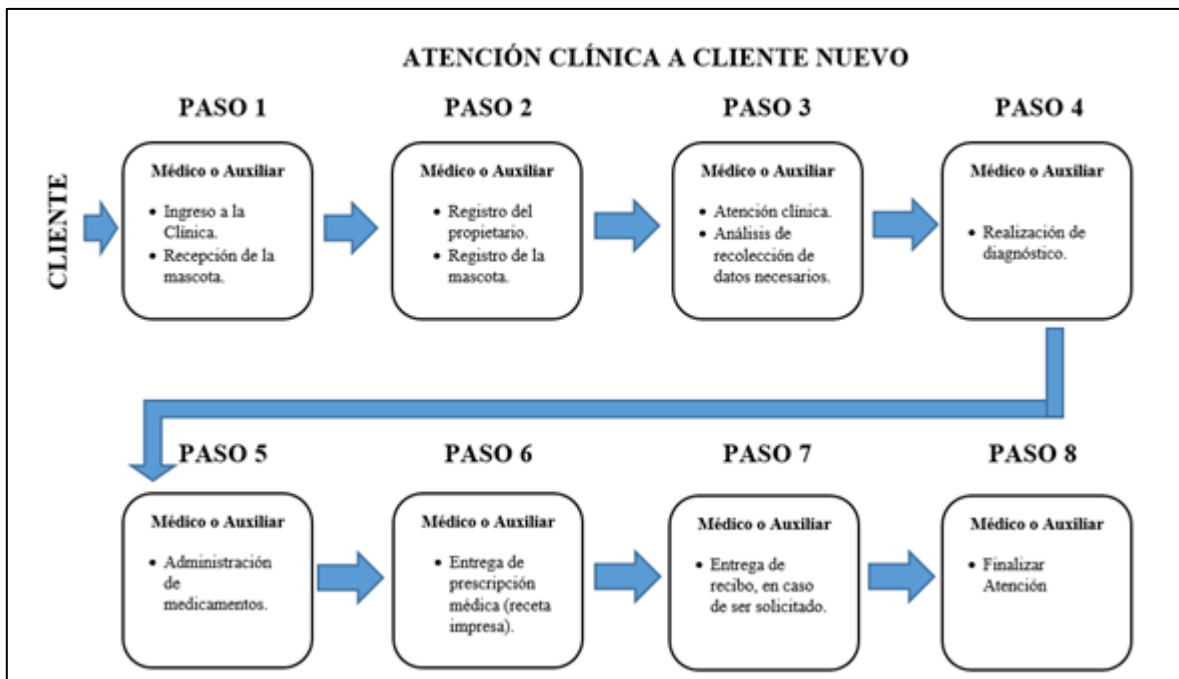


Figura 22 Flujo de atención

Fuente: Elaboración propia

I.4.6.4 Interfaz de Usuario Amigable

La interfaz de usuario será diseñada de manera intuitiva y amigable, con menús y opciones claras para facilitar la navegación y el uso del sistema. Se prestará especial atención al diseño de formularios y pantallas para garantizar la captura de datos de manera eficiente y precisa.

I.4.6.5 Integración de Tecnologías y Recursos

El sistema se construirá mediante la utilización del lenguaje de programación Python, respaldado por el framework FastAPI, con el objetivo de asegurar una implementación eficiente y altamente escalable. Para el almacenamiento de datos, se empleará una base de

datos del tipo ORM, específicamente SQLAlchemy, permitiendo una gestión estructurada y eficaz de la información. La interfaz de usuario se desarrollará utilizando React, garantizando una experiencia fluida e interactiva para los usuarios finales.

I.4.6.6 Seguridad y Privacidad

El proyecto incorporará rigurosas medidas de seguridad destinadas a resguardar la confidencialidad e integridad de la información tanto de los pacientes como de la clínica veterinaria. Se implementarán procedimientos de autenticación de usuarios, control de acceso basado en permisos y cifrado de datos.

Además, se empleará la tecnología JSON Web Token (JWT) como parte fundamental de la estrategia de seguridad. Los JWT proporcionan un método eficiente y seguro para transmitir información de forma firmada y encriptada entre partes. La elección de JWT se basa en su capacidad para autenticar a usuarios de manera segura y eficiente, así como en su capacidad para transmitir datos de manera segura entre el cliente y el servidor, contribuyendo a una capa adicional de seguridad en la autenticación y autorización del sistema.

I.5 Estado del Arte

I.5.1 Análisis de Sistemas Actuales

En la actualidad, se encuentra en el mercado una amplia variedad de sistemas de información desarrollados por distintas entidades, destinados a la administración de clínicas veterinarias. Al analizar minuciosamente cada una de estas aplicaciones, se percibe una marcada semejanza en sus funcionalidades, aunque algunas presentan atributos distintivos. Es evidente que comparten similitudes en su operatividad. No obstante, cada una destaca por características únicas.

A continuación, se proporciona un desglose de algunos de los sistemas más destacados en el mercado que guardan relación con el proyecto que se busca implementar:

I.5.1.1 QVET

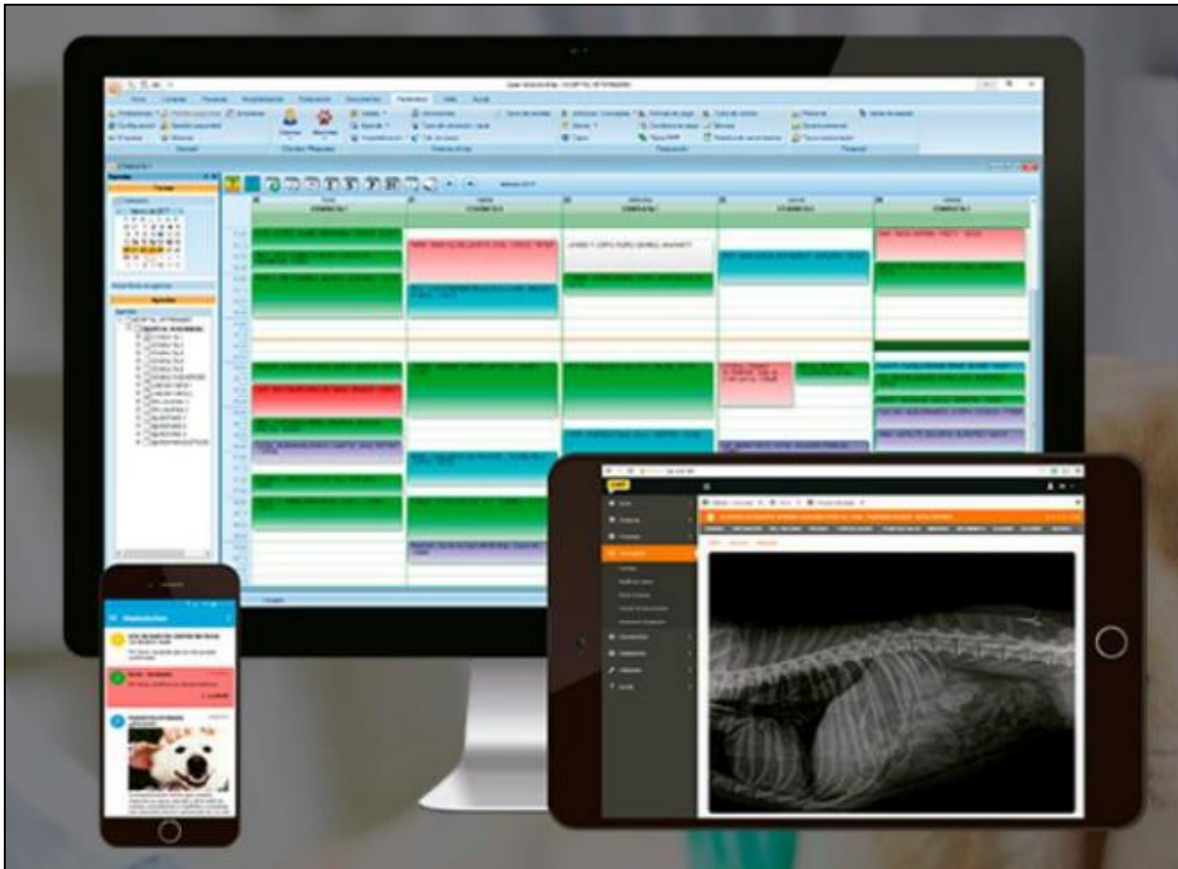


Figura 23: Página Web QVET

Fuente: <https://qvnet.net/>

QVET es un programa de gestión de clínicas y hospitales veterinarios, el cual está disponible tanto como plataforma web, mobile y desktop. Cuenta con las siguientes características:

Historia clínica online: Organización de la información clínica, facilitando su registro durante la consulta y localización para una rápida lectura del caso.

Compras, ventas y stock: Mantiene su nivel de inventario optimizado, proponiendo sus compras periódicas mejorando su rentabilidad evitando roturas de stock.

Agenda electrónica: Organización de las consultas programadas, cirugías, etc. de todos los profesionales de forma individual y se comunica con sus clientes para recordarles la asistencia a su cita.

Facturación electrónica: QVET cumple las normativas fiscales y certificaciones de los países en los que se utiliza, en facturación ordinaria y en facturación electrónica e impresoras discasles.

Relación con los clientes: Con la versión mobile, mejora la comunicación con los clientes realizando recordatorios, o mandando alertas para que ellos estén al tanto de las últimas noticias y promociones respectivas al negocio.

Gestión de negocio: Proporciona herramientas que le facilitan monitorizar los principales indicadores de su negocio de manera fácil, clara y rápida. [39].

I.5.1.2 VetCloud

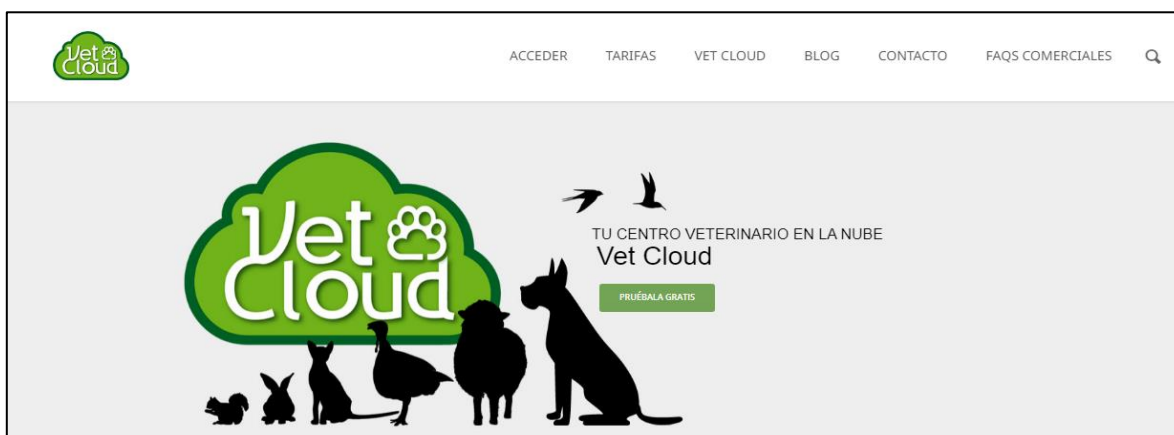


Figura 24: Página Web VetCloud

Fuente: <https://gestiondeclinicasveterinarias.com/>

Es un sistema de gestión para clínicas veterinarias ubicado en la nube, al cual los usuarios pueden acceder desde cualquier lugar con acceso a internet, adicionalmente cuenta con una APP que los usuarios pueden descargar para acceder a su información desde cualquier dispositivo móvil, está diseñado para ser multi-centro, lo cual quiere decir que permite administrar varias sedes de una misma clínica en un solo sistema, para acceder a esta plataforma es necesario el pago de 57 Euros mensuales. Cuenta con las siguientes características:

Administración de usuarios: Cuenta con un módulo de administración de usuarios, el cual permite definir diferentes perfiles de acceso a cada uno de los empleados de la clínica veterinaria; cada médico y empleado de la clínica puede visualizar su propia información de manera independiente a la de los demás.

Gestión de clientes y pacientes: Permite el registro y actualización de la información básica y de contacto de los clientes y de sus mascotas, admitiendo más de una mascota por cliente, cuenta con un módulo de consulta que permite ubicar fácilmente un cliente o un paciente por medio de diferentes criterios de búsqueda.

Agenda: Permite programar las citas médicas de cada uno de los pacientes, asignado el médico veterinario que lo atenderá, teniendo en cuenta la disponibilidad de la agenda que los médicos tengan en el momento, diferenciando gráficamente los días festivos, vacaciones, los días en los que cada médico no está disponible y los horarios ya separados por otros pacientes; también permite diferenciar el tipo de servicio que se va a prestar en cada cita registrada en la agenda (control, cirugía, desparasitación, vacunación, etc.). Esta aplicación no solo está disponible para los trabajadores de la clínica veterinaria, también permite a los clientes acceder a ella y programar sus citas médicas.

Notificaciones: Permite programar recordatorios automáticos sobre citas médicas próximas a cumplirse, vacunaciones, así como notificar promociones o campañas que realice la clínica, estas notificaciones son enviadas por medio de mensajes de texto o correo electrónico a los clientes dueños de las mascotas para las cuales va dirigida atención.

Historias clínicas: Permite llevar el registro de las consultas, exámenes de diagnóstico, vacunas y tratamientos aplicados a cada paciente, así como la generación de fórmulas digitales e impresión de expedientes médicos.

Servicios: Permite realizar la gestión de otros servicios como guardería.

Reportes: Cuenta con generación de reportes analíticos que le permiten al dueño de la clínica evidenciar sus ingresos y gastos del mes de forma numérica y gráfica, adicionalmente permite configurar la generación de diferentes reportes que le permitan a las clínicas veterinarias analizar el comportamiento de su negocio y exportarlos en libros de Excel o archivos PDF.

Gestión contable: Permite generar liquidaciones de honorarios a los empleados la clínica de forma automática, generación de facturas a los clientes, y llevar el registro de las facturas de las compras que se realizan a los proveedores, así como controlar el stock de sus productos, registro de ventas de productos, identificación de clientes morosos.

Seguridad: Cuenta con el registro de las transacciones realizadas en la plataforma en un log transaccional, generación automática de copias de seguridad de la información de la clínica, la cual será custodiada por el dueño de la clínica. [40].

I.5.1.3 Vetter Sistemas

Vetter₅
Gestión Veterinaria Integral

Soluciones Informáticas
para Veterinarias y Pet's Shop

Bolivia

Inicio Descargas Manuales Video Tutoriales Comprar Contacto Actualizaciones

Vetter₅
NUEVA VERSION
Software para Veterinarias
DESCARGAR AHORA Windows

Descargue **Vetter 5** a su PC y pruébelo 15 días **GRATIS**.
No es un demo, es la versión **FULL** con todas las utilidades.

Características Generales:

- Listado de Clientes y Pacientes
- Ficha Clínica
- Vacunación y Desparasitación
- Controles y Recordatorios
- Avisos por e-mail
- Métodos Complementarios
- Peluquería
- Guardería e Internación
- Fotos y Archivos Adjuntos
- Estadísticas e Informes
- Caja Diaria y Facturación
- Control de Stock
- Cuenta Corriente
- Proveedores
- Funciona en RED
- Sistema de Mutual
- ...y mucho mas.

Figura 25: Página Web Vetter Sistemas

Fuente: <https://www.vetter.com.ar/bolivia/bolivia.htm>

Es una aplicación de escritorio que permite la gestión de clínicas veterinarias y Pet's Shop, es compatible con Windows Vista y Windows 7, posee funciones que apoyan funciones como la administración de las historias clínicas y facturaciones, permite la descarga de una versión demo por un mes, para que los clientes prueben la plataforma antes de adquirir la licencia, para obtenerla es necesario el pago de 375 Dólares. Cuenta con las siguientes características:

Administración de usuarios: Cuenta con un módulo de administración de usuarios, el cual permite definir diferentes perfiles de acceso a cada uno de los empleados de la clínica veterinaria; cada médico y empleado de la clínica puede visualizar su propia información de manera independiente a la de los demás.

Gestión de clientes y pacientes: Permite el registro y actualización de la información básica y de contacto de los clientes y de sus mascotas, admitiendo más de una mascota por cliente,

cuenta con un módulo de consulta que permite ubicar fácilmente un cliente o un paciente por medio de diferentes criterios de búsqueda.

Notificaciones: Permite llevar el registro de las notificaciones realizadas a los clientes sobre citas médicas próximas a cumplirse, vacunaciones, promociones o campañas que realice la clínica, estas notificaciones son realizadas de manera manual o presencial por el médico veterinario y luego son registradas en el sistema para identificar los clientes ya notificados y los pendientes.

Historias clínicas: Permite llevar el registro de las consultas, exámenes de diagnóstico, radiología, vacunaciones y tratamientos aplicados a cada paciente y visualizarlos en el momento de la consulta, adjuntar imágenes a la historia clínica que permitan un manejo adecuado de las patologías, así como la generación de fórmulas digitales e impresión de expedientes médicos y calendarios de vacunación.

Servicios: Permite realizar la gestión de otros servicios como peluquería.

Reportes: Cuenta con generación de diferentes reportes que les permitan a las clínicas veterinarias analizar el comportamiento de su negocio.

Gestión contable: Permite controlar el stock de sus productos.

Gestión de Proveedores: Permite realizar la gestión de los proveedores de la clínica, registrando la información de contacto y de los productos que les proveen.

Seguridad: Cuenta con generación automática de copias de seguridad de la información de la clínica, la cual será custodiada por el dueño de la clínica. [41].

I.5.1.4 GestorVet

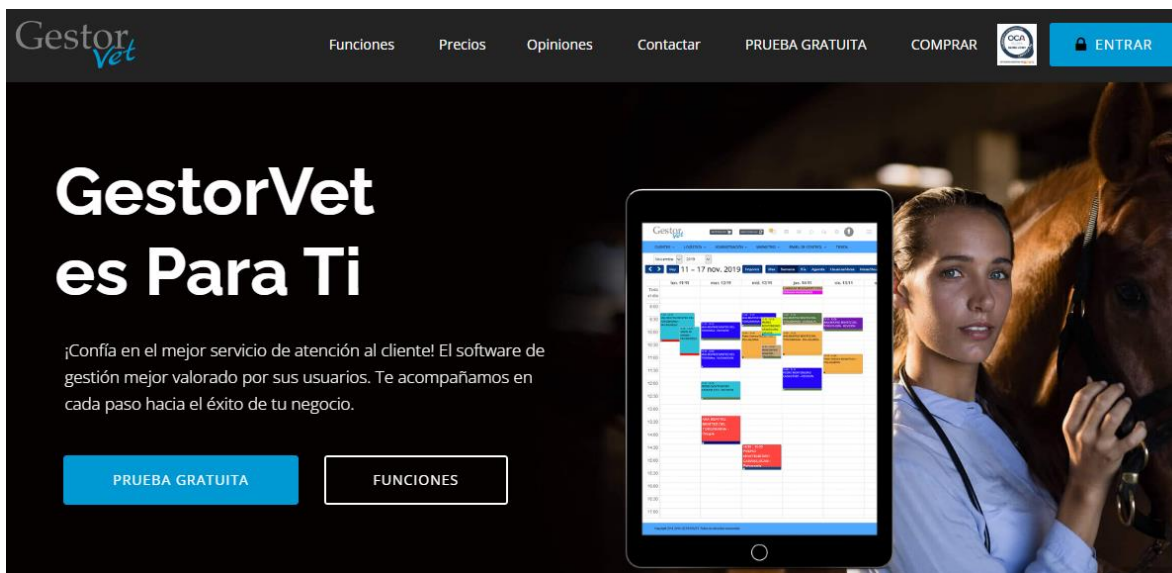


Figura 26: Página Web GestorVet

Fuente: <https://www.gestorvet.com>

Es un sistema de gestión para clínicas veterinarias ubicado en la nube, al cual los usuarios pueden acceder desde cualquier lugar con acceso a internet, adicionalmente cuenta con una APP que los usuarios pueden descargar para acceder a su información desde cualquier dispositivo móvil, permite una gestión avanzada de agendas y tareas.

Gestionar varios centros veterinarios a la vez, es decir que se pueden controlar varias sedes de la misma clínica en un solo sistema, generar informes de gestión, lleva la trazabilidad de los productos ofrecidos en la clínica, presentando Información en tiempo real y permite la actualización de tarifas de forma automatizada, para acceder a esta plataforma es necesario adquirir una licencia de uso bajo diferentes modalidades:

- Pago mensual de 38 Euros
- Pago anual de 420 Euros

Cuenta con las siguientes características:

Administración de usuarios: Cuenta con un módulo de administración de usuarios, el cual permite definir diferentes perfiles de acceso a cada uno de los empleados de la clínica veterinaria; cada médico y empleado de la clínica puede visualizar su propia información de manera independiente a la de los demás.

Gestión de clientes y pacientes: Permite el registro y actualización de la información básica y de contacto de los clientes y de sus mascotas, admitiendo más de una mascota por cliente, cuenta con un módulo de consulta que permite ubicar fácilmente un cliente o un paciente por medio de diferentes criterios de búsqueda.

Agenda: Permite programar las citas médicas de cada uno de los pacientes, asignado el médico veterinario que lo atenderá, teniendo en cuenta la disponibilidad de la agenda que los médicos tengan en el momento, diferenciando gráficamente los días festivos, vacaciones, los días en los que cada médico no está disponible y los horarios ya separados por otros pacientes; también permite diferenciar el tipo de servicio que se va a prestar en cada cita registrada en la agenda (control, cirugía, desparasitación, vacunación, etc.).

Notificaciones: Permite programar recordatorios automáticos sobre citas médicas próximas a cumplirse, vacunaciones, así como notificar promociones o campañas que realice la clínica, estas notificaciones son enviadas por medio de mensajes de texto o correo electrónico a los clientes dueños de las mascotas para las cuales va dirigida atención, adicionalmente permite llevar el registro de las notificaciones realizadas de manera manual o presencial a los clientes para identificar los clientes ya notificados y los pendientes.

Historias clínicas: Permite llevar el registro de las consultas, exámenes de diagnóstico, vacunas y tratamientos aplicados a cada paciente y visualizarlos en el momento de la consulta, así como la generación de fórmulas digitales e impresión de expedientes médicos.

Reportes: Cuenta con generación de reportes analíticos que le permiten al dueño de la clínica evidenciar sus ingresos y gastos del mes de forma numérica y gráfica, adicionalmente permite configurar la generación de diferentes reportes que le permitan a las clínicas veterinarias analizar el comportamiento de su negocio y exportarlos en libros de Excel o archivos PDF.

Gestión contable: Permite la generación de facturas a los clientes, y llevar el registro de las facturas de las compras que se realizan a los proveedores, así como controlar el stock de sus productos, registro de ventas de productos por medio de lector de código de barras, identificación de clientes morosos.

Gestión de Proveedores: Permite realizar la gestión de los proveedores de la clínica, registrando la información de contacto y de los productos que les proveen.

Seguridad: Cuenta con la generación automática de copias de seguridad de la información de la clínica, la cual será custodiada por el dueño de la clínica. [42].

I.5.2 Conclusiones

En líneas generales, todos los programas destinados a clínicas veterinarias cumplen con las funciones básicas necesarias. Estos ofrecen opciones de registro para clientes, mascotas y los procedimientos realizados en los animales. No obstante, se han identificado algunos inconvenientes en varios de ellos. Por ejemplo, Vetter Sistemas carece en gran medida de iconografía, lo que puede limitar la experiencia del usuario. Por otro lado, en el software Qvet, aunque se emplean iconos para representar acciones, algunos de ellos resultan confusos y no reflejan con precisión sus funciones, lo que puede generar confusiones, especialmente para usuarios menos experimentados.

Un aspecto desfavorable compartido por todos los sistemas mencionados es su diseño exclusivo para computadoras o laptops, lo que los limita a ser sistemas monousuarios.

Además de las características mencionadas, la clínica veterinaria ha expresado la necesidad de que el sistema sea portable, ya que los profesionales realizan consultas en lugares diversos. Por lo tanto, se busca desarrollar una plataforma que sea fácilmente accesible tanto en computadoras como en tabletas conectadas a internet.

I.6 Objetivos

I.6.1 Objetivo General

Mejorar el proceso de atención a los pacientes de la Veterinaria Málaga.

I.6.2 Objetivos Específicos

- Desarrollar un Sistema Informático para la Gestión eficiente de Datos.
- Capacitar al personal de la veterinaria para el uso correcto del sistema automatizado.

I.7 Matriz de Marco Lógico (MML)

Tabla 1: Matriz de Marco Lógico

RESUMEN NARRATIVO	INDICADORES	MEDIOS DE VERIFICACIÓN	SUPUESTOS
<p>Fin Contribuir a mejorar la atención de la Veterinaria Málaga.</p>	<p>Luego de 2 años de finalizado el proyecto, el grado de satisfacción del cliente aumentó un 80% con respecto al año base (2023).</p>	<p>Carta del médico veterinario encargado de la clínica avalando el grado de satisfacción en el uso y funcionamiento del sistema desarrollado.</p>	<ul style="list-style-type: none"> La veterinaria Málaga mantiene su plaza en el mercado. El personal responsable hace uso del sistema funcional automatizado.
<p>Propósito Mejorado del proceso de atención a los pacientes de la Veterinaria Málaga.</p>	<p>Al finalizar el proyecto, el 100% de los procesos relacionados a la atención de pacientes en la Veterinaria Málaga han sido automatizados. <i>Ver anexo F</i></p>	<p>Carta por parte del médico veterinario encargado que manifiesta la realización del proyecto de manera exitosa cumpliendo los requerimientos solicitados. <i>Ver anexo F</i></p>	<ul style="list-style-type: none"> Apoyo para el desarrollo y ejecución del proyecto por parte del personal. Disponibilidad de información necesaria por parte del personal para mejoras del sistema.
<p>Componentes</p> <p>C1 Desarrollar un Sistema Informático para la Gestión eficiente de Datos.</p>	<p>C1 A los 8 meses de iniciado el proyecto, se ha desarrollado un sistema informático, basado en los requerimientos expresados bajo la norma IEEE830.</p>	<p>C1 Informe de conformidad de cumplimiento de los requisitos funcionales del sistema descritos en la norma IEEE830, avalado por la docente de la materia de Taller 3. <i>Ver anexo F</i></p>	<p>C1 Los encargados de la Veterinaria Málaga apoyan el desarrollo del sistema para mejorar la atención de pacientes.</p>
<p>C2 Capacitar al personal de la veterinaria para el uso correcto del sistema automatizado.</p>	<p>C2 Al finalizar el proyecto, se han realizado al menos dos capacitaciones en el uso del sistema informático a los encargados de la Clínica Veterinaria Málaga.</p>	<p>C2 Informe de conformidad de cumplimiento de la capacitación, avalado por la docente de la materia de Taller 3 y el médico veterinario encargado.</p>	<p>C2 Disponibilidad por parte del personal responsable de la clínica en aprender el manejo del sistema desarrollado.</p>

<p>Actividades</p> <p>C1 Las fases a llevar a cabo siguiendo la metodología Scrum incluyen:</p> <ol style="list-style-type: none"> 1. Creación del Product Backlog 2. Sprints: <ul style="list-style-type: none"> • Planeación del Sprint • Desarrollo • Revisión del Sprint 3. Producción 	<p>C1 Al final de noviembre de 2023, se tiene el 100% de los procesos ya automatizados.</p> <p>Costos: de acuerdo al presupuesto general del proyecto. <i>Ver anexo B.</i></p>	<p>C1 • Calendario propuesto para el proyecto se cumple.</p> <p>• Documentación del desarrollo del sistema completada.</p>	<p>C1 • Se cuenta con la disponibilidad económica para sustentar las actividades propuestas.</p> <p>• El personal proporciona información para el desarrollo del sistema de acuerdo a las necesidades establecidas</p>
<p>C2 Estrategia de Socialización y Capacitación del modelo desarrollado e implementado.</p> <ul style="list-style-type: none"> • Elaborar las diapositivas • Preparar el Sistema (instalarlo en los equipos del personal a capacitar) • Cargar Datos de prueba al Sistema • Elaborar una guía de manejo paso a paso. 	<p>C2 Al finalizar el proyecto y la implementación, el personal estará capacitado en un 90%.</p>	<p>C2 • Material Multimedia (fotos) que avalen la realización de la capacitación pertinente.</p> <p>• Revisiones por parte de la docente de la materia Taller III.</p>	<p>C2 • Asistencia a la capacitación por parte del personal responsable.</p>

I.8 Resultados Esperados

- Sistema informático: El sistema permitirá administrar, almacenar adecuadamente los documentos de historias clínicas y datos de los pacientes de la Veterinaria “Málaga”.
- Capacitación en el uso del sistema: se realizará capacitaciones al personal de la veterinaria para el uso óptimo del software desarrollado.

I.9 Beneficiarios

I.9.1 Beneficiarios Directos

Los beneficiarios directos son: el médico encargado de la Veterinaria Málaga y los auxiliares que lleguen a trabajar en la misma, este proyecto les permitirá mejorar los procesos de atención, y el ágil acceso a la información de cada paciente para un mejor seguimiento de la Veterinaria Málaga.

I.9.2 Beneficiarios Indirectos

Los beneficiarios indirectos son la población en general, los vecinos de la zona que recibirán una atención personalizada y de calidad.

I.10 Alcances

- El sistema gestionará eficientemente la información relacionada con las mascotas.
- Se encargará de la administración de datos de los propietarios de las mascotas.
- Registrará y mantendrá datos precisos de las atenciones clínicas realizadas.
- Administrará de manera integral los historiales clínicos de los pacientes.
- Facilitará la gestión visual de una agenda de citas para una mejor organización.
- Gestionará el inventario de medicamentos, enfocándose en aquellos recetados por los médicos veterinarios.
- Generará informes y reportes necesarios para los responsables de la clínica veterinaria.

I.11 Limitaciones

- En esta fase inicial del proyecto, es importante destacar que el sistema no abordará procesos relacionados con la facturación de servicios ni llevará a cabo funciones vinculadas al control monetario.

- El enfoque principal se centrará en el desarrollo y optimización de las funcionalidades clínicas y administrativas, dejando fuera de su alcance cualquier aspecto relacionado con transacciones financieras o gestión monetaria.

1.12 Cronograma de Actividades

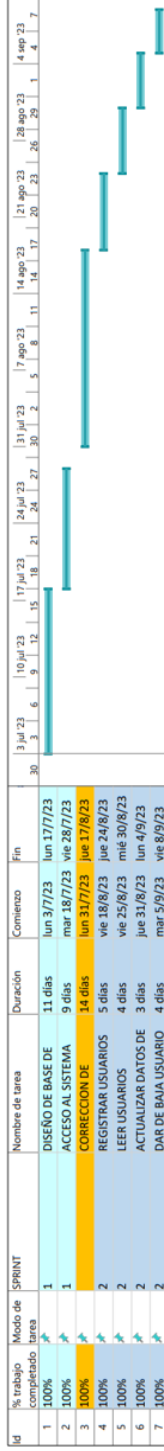


Figura 27: Cronograma de actividades Parte1
Fuente: Elaboración propia

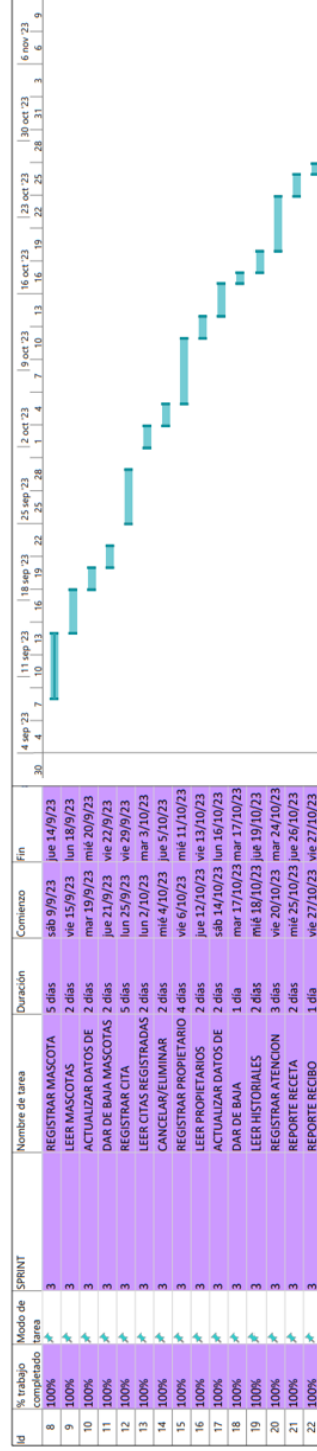


Figura 28: Cronograma de actividades Parte2
Fuente: Elaboración propia

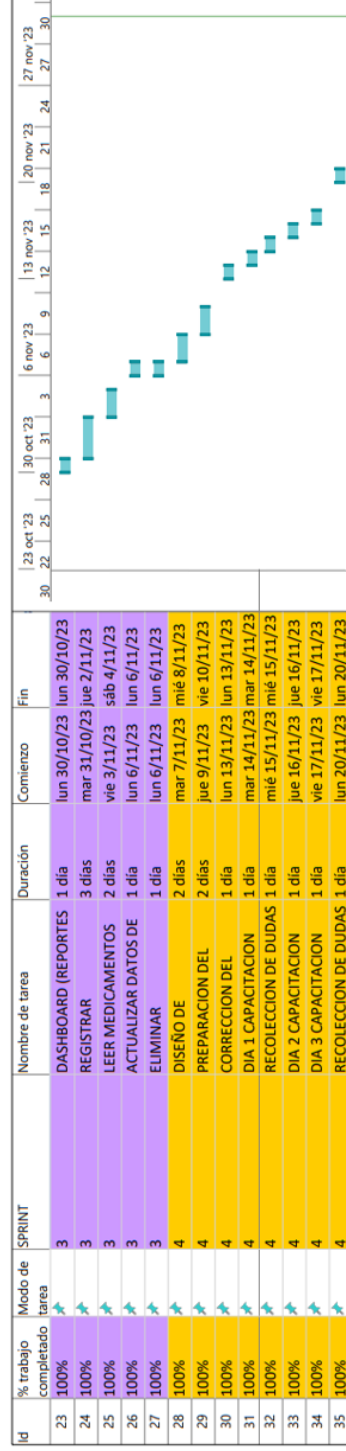


Figura 29: Cronograma de actividades Parte3
Fuente: Elaboración propia

CAPÍTULO II: COMPONENTES

II.1 Componente 1: Sistema

II.1.1 Plan de Desarrollo del Componente Sistema

II.1.1.1 Introducción

En el desarrollo del componente Sistema, hemos optado por la metodología Scrum, una metodología ágil que se caracteriza por su enfoque iterativo e incremental. Scrum nos brinda una estructura flexible y colaborativa que se adapta de manera efectiva a los cambios y evoluciones en el proyecto. A través de ceremonias como la Planificación del Sprint, el Daily Scrum, la Revisión del Sprint y la Retrospectiva del Sprint, Scrum nos guiará en la implementación del sistema de manera eficiente y centrada en el valor.

II.1.1.2 Propósito

El propósito del Plan de Desarrollo del Sistema es proporcionar la información necesaria para controlar el avance del proyecto. En él se describe el enfoque para el desarrollo del componente I.

II.1.1.3 Alcance

El Plan de Desarrollo del Sistema describe el plan global usado para el desarrollo de un Sistema Informático que está comprendido por los siguientes Módulos:

- Ingreso al sistema
- Gestionar citas
- Gestionar usuarios
- Gestionar propietarios
- Gestionar mascotas
- Historiales Clínicos
- Atención Clínica
- Gestionar medicamentos
- Dashboard (Reportes Gráficos)

La implementación del sistema automatizado dará beneficios al cliente, dando soluciones a los problemas que tiene con el manejo de la información de las mascotas y sus propietarios, de las atenciones clínicas y los historiales, el inventario de los medicamentos y el manejo de

reportes requeridos; este sistema proyectará una solución a mediano plazo que beneficiará a la institución.

II.1.1.4 Objetivo General

Diseñar, desarrollar e implementar un sistema de gestión para mejorar el proceso de atención a los pacientes de la Clínica Veterinaria Málaga, que cumpla con los requerimientos.

II.1.1.5 Objetivos Específicos

- Implementar medidas de seguridad robustas, como la autenticación de usuarios y la gestión de roles, para proteger el acceso no autorizado a la información sensible del sistema.
- Desarrollar funcionalidades que permitan la captura eficiente de datos de mascotas, incluyendo información médica, datos de propietarios y detalles específicos de las mascotas.
- Implementar un sistema de gestión de citas que permita a los veterinarios y al personal administrativo programar y gestionar citas de manera efectiva, evitando conflictos y optimizando la agenda.
- Crear un sistema de historiales clínicos electrónicos que permita el almacenamiento seguro y accesible de la información médica de los pacientes, facilitando el seguimiento de tratamientos y diagnósticos.
- Desarrollar funcionalidades para registrar y documentar las atenciones clínicas, incluyendo diagnósticos, tratamientos y procedimientos realizados durante las consultas.
- Implementar un sistema de administración de inventarios que permita llevar un registro preciso de los medicamentos utilizados en las atenciones clínicas, gestionando la disponibilidad y evitando agotamientos.
- Desarrollar herramientas para generar informes y estadísticas relevantes, proporcionando a la clínica una visión integral de su desempeño, la carga de trabajo y otros indicadores clave.

II.1.1.6 Metodologías Utilizadas

La metodología que se utilizó en la elaboración del presente proyecto de grado, específicamente para el desarrollo del componente sistema fue la Metodología Ágil Scrum,

la cual no tiene una estructura cien por ciento rígida o autocrática, más por el contrario nos anima a adaptarla a las necesidades de cada proyecto por lo que a continuación se explicará la forma en que se utilizó esta metodología en el actual proyecto.

Como primera parte se procedió a definir el Product Goal y realizar el Product Backlog, lo cual en líneas generales significa que se definió la meta del producto que queremos llegar a obtener con sus limitaciones y alcances, va muy ligado a los objetivos del perfil del proyecto a su vez en el Product Backlog se identificó mini objetivos y características principales que debe tener el Sistema, los cuales se los colocaron en un estilo de lista para que puedan ser seleccionados y priorizados en cada Sprint a lo largo del proyecto.

A continuación, se realizaron una serie de pasos que se repitieron en forma cíclica acorde a la metodología Scrum.

El primero de ellos es el Sprint Planning donde se seleccionaron cierto número de ítems del Product Backlog y se los pasaron al Sprint Backlog en forma de Historias de Usuario definiendo que es lo que cada ítem debería cumplir para ser considerado terminado. En esta parte también se define la prioridad de cada uno de los ítems.

Y para finalizar el Sprint se tuvo una revisión denominada Sprint Review con la docente encargada de la materia que jugó el papel de Scrum Master brindando lo que se conoce como Sprint Retrospective que vendría a ser las observaciones encontradas que deberían ser mejoradas o cuando menos discutidas, este proceso de Sprint se repitió tres veces a lo largo del proyecto.

II.1.1.7 Software de Base Utilizado

El desarrollo del sistema para el manejo de datos de la clínica veterinaria implica la utilización de una combinación de tecnologías y herramientas que permiten la creación de una aplicación eficiente. Los principales componentes principales utilizados en el proyecto fueron:

- *Lenguajes de Programación:*
 - **Python:** Utilizado como lenguaje principal para el desarrollo del backend de la aplicación, aprovechando su versatilidad y amplio soporte en el ámbito de desarrollo web.

- **JavaScript (JS):** Empleado para la creación de la interfaz de usuario dinámica en el frontend, facilitando una experiencia interactiva para los usuarios finales.
- *Bibliotecas y Frameworks:*
 - **React:** Adoptado como el framework de JavaScript para el desarrollo de la interfaz de usuario de tipo SPA, proporcionando un enfoque moderno y eficiente para la navegación y la interactividad en la aplicación.
 - **FastAPI:** Utilizado como el framework de backend en Python, ofrece un rendimiento excepcional y una fácil implementación de API RESTful, optimizando la comunicación entre el frontend y el backend.
 - **SQLAlchemy:** Empleado como ORM (Object-Relational Mapping) para la interacción con la base de datos, facilitando la gestión y manipulación de datos de manera eficiente y segura.
- *Base de Datos:*
 - Se utilizó una base de datos relacional para almacenar y gestionar los datos de la clínica veterinaria. La combinación de FastAPI y SQLAlchemy permitió una integración fluida con la base de datos, garantizando la consistencia y la integridad de los datos.

II.1.1.8 Requisitos Funcionales

RF1 Autenticación de Ingreso al Sistema

- *RF1.1 Iniciar Sesión* - El acceso al sistema solo será posible mediante la introducción de un nombre de usuario y una contraseña válidos. Si la información no es válida, se mostrará un mensaje indicando que el usuario no está registrado.
- *RF1.2 Asignar Usuario* - Después de la validación, el sistema asignará el rol de usuario que está ingresando al sistema.

RF2 Gestión Usuarios

- *RF2.1 Registrar Nuevo Usuario* - Esta función permite el registro de un usuario nuevo al proporcionar un nombre de usuario y una contraseña.
- *RF2.2 Leer Usuarios Registrados* - Esta función muestra una lista de usuarios registrados con una opción de búsqueda que facilita la navegación a través de la lista.
- *RF2.3 Actualizar datos de Usuarios* - Esta función permite realizar modificaciones válidas en los datos de un usuario ya registrado.

- *RF2.4 Dar de baja Usuario* - En caso de que un usuario ya no forme parte del equipo de la Clínica Veterinaria, esta función permite desactivar su cuenta.

RF3 Gestión Propietarios

- *RF3.1 Registrar Nuevo Propietario* - Esta función permite el registro de un nuevo propietario, ingresando la información necesaria.
- *RF3.2 Leer Propietarios Registrados* - Esta función muestra una lista de propietarios registrados, con una opción de búsqueda para facilitar el acceso a los datos.
- *RF3.3 Actualizar datos de Propietarios* - Esta función permite realizar modificaciones válidas en los datos de un propietario registrado.
- *RF3.4 Dar de baja Propietario* - En caso de que un propietario ya no tenga mascotas en la clínica, esta función permite desactivar su registro.

RF4 Gestión Mascotas

- *RF4.1 Registrar Nueva Mascota* - Permite el registro de una nueva mascota con un propietario ya registrado previamente.
- *RF4.2 Leer Mascotas Registrados* - Esta función muestra una lista de mascotas registradas de acuerdo a un propietario seleccionado.
- *RF4.3 Actualizar datos de Mascotas* - Esta función permite realizar modificaciones válidas en los datos de una mascota registrada.
- *RF4.4 Dar de baja Mascota* - En caso de que una mascota ya no sea atendida en la clínica, esta función permite desactivar su registro.

RF5 Agenda de Citas

- *RF5.1 Agendar Cita Nueva* - Esta función permite el registro de una cita nueva mediante la selección del día requerido en un calendario que recopila la información necesaria para el registro.
- *RF5.2 Leer Citas Registradas* - En el calendario de la vista, se pueden observar las citas según los días, semanas o meses.
- *RF5.3 Cancelar Citas Registradas* - Si es necesario, se pueden quitar del calendario las citas seleccionadas con el fin de liberar espacio en la agenda.

RF6 Atención Clínica

- *RF6.1 Registrar Atención Clínica* – Esta función permite el registro de las atenciones clínicas de acuerdo a la mascota seleccionada.
- *RF6.2 Receta (Reporte)* – Se podrá generar una receta, esta podrá imprimirse o guardarse en PDF.
- *RF6.3 Recibo (Reporte)* – Se podrá generar un recibo, si el cliente lo solicita, con los medicamentos recetados en la consulta; este podrá imprimirse o guardarse en PDF.

RF7 Gestión Historiales Clínicos

- *RF7.1 Leer Historiales Clínicos* - Se mostrará una lista de mascotas con la opción de acceder a los historiales de cada mascota. Los historiales se presentarán ordenados por fechas, detallando las atenciones realizadas.
- *RF7.2 Reporte Historia Clínica por fecha* – Se podrá generar un reporte impreso de la historia clínica de cualquier paciente.

RF8 Gestión Inventario de Medicamentos

- *RF8.1 Registrar Medicamento Nuevo* - Esta función permite el registro de un nuevo medicamento en el inventario.
- *RF8.2 Leer Medicamentos Registrados* - Se mostrará una lista de medicamentos registrados, con un buscador que facilita la búsqueda y el filtrado de datos.
- *RF8.3 Actualizar datos de Medicamentos* - Esta función permite realizar modificaciones válidas en los datos de los medicamentos registrados, como la actualización del stock disponible y el precio.
- *RF8.4 Borrar Medicamentos* – Esta función permite eliminar un medicamento registrado que ya no se utiliza en la veterinaria.

RF9 Dashboard (Reportes Gráficos)

- *RF9.1 Reporte Medicamentos* - Se mostrará un reporte gráfico de los medicamentos más administrados por mes.
- *RF9.2 Reporte Servicios* - Se mostrará un reporte gráfico de los servicios más administrados por mes.
- *RF9.3 Reporte Atención Médicos* - Se mostrará un reporte gráfico de los médicos que realizaron más atenciones en un mes.
- *RF9.4 Reporte distribución de Géneros* - Se mostrará un reporte gráfico de las mascotas más atendidas según su género por mes.
- *RF9.5 Reporte distribución de Especies* - Se mostrará un reporte gráfico de las mascotas más atendidas según su especie por mes.
- *RF9.6 Reporte Atenciones Clínicas* - Se mostrará un reporte gráfico de las atenciones clínicas anuales.

II.1.1.9 Requisitos No Funcionales

- *Requisitos de rendimiento*
 - Seguridad: Asegura la autenticación, control de acceso y registro de transacciones, con procedimientos documentados para el manejo, además de encriptación de contraseñas.
 - Fiabilidad: Exige una tasa de error del 0%, reportando cualquier fallo en las transacciones para su corrección.
 - Disponibilidad: Mantiene el sistema disponible 24/7.
 - Mantenibilidad: Garantiza la trazabilidad del sistema a través de la documentación para mantenimiento.

II.1.1.10 Modelado del Sistema

II.1.1.10.1 Diagrama de Componentes

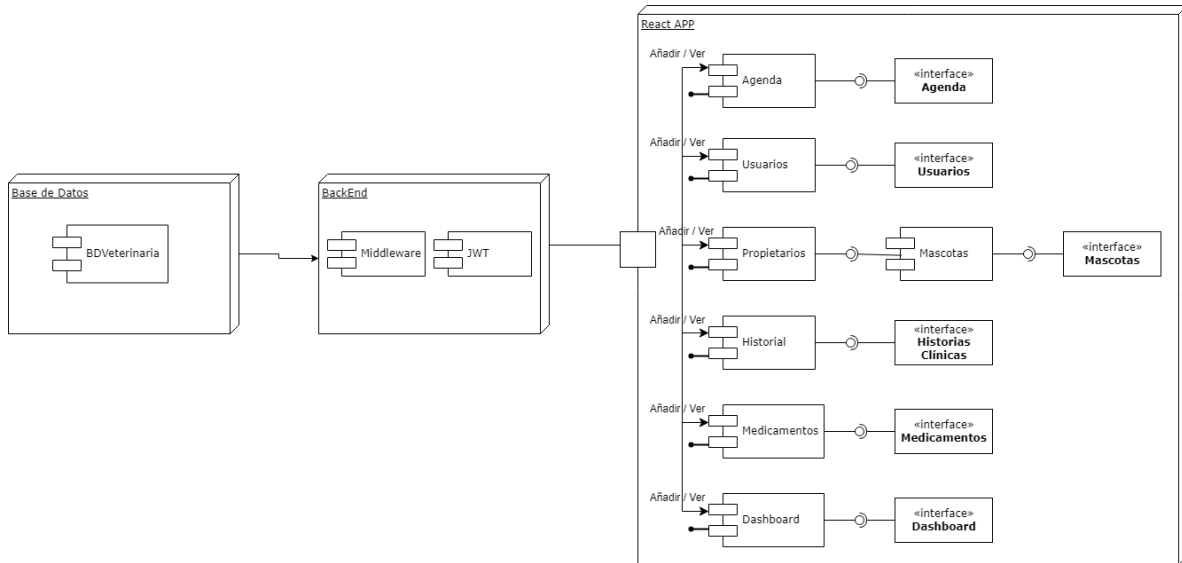


Figura 30: Diagrama de Componentes

Fuente: Elaboración propia

II.1.1.10.2 Diagrama de Paquetes

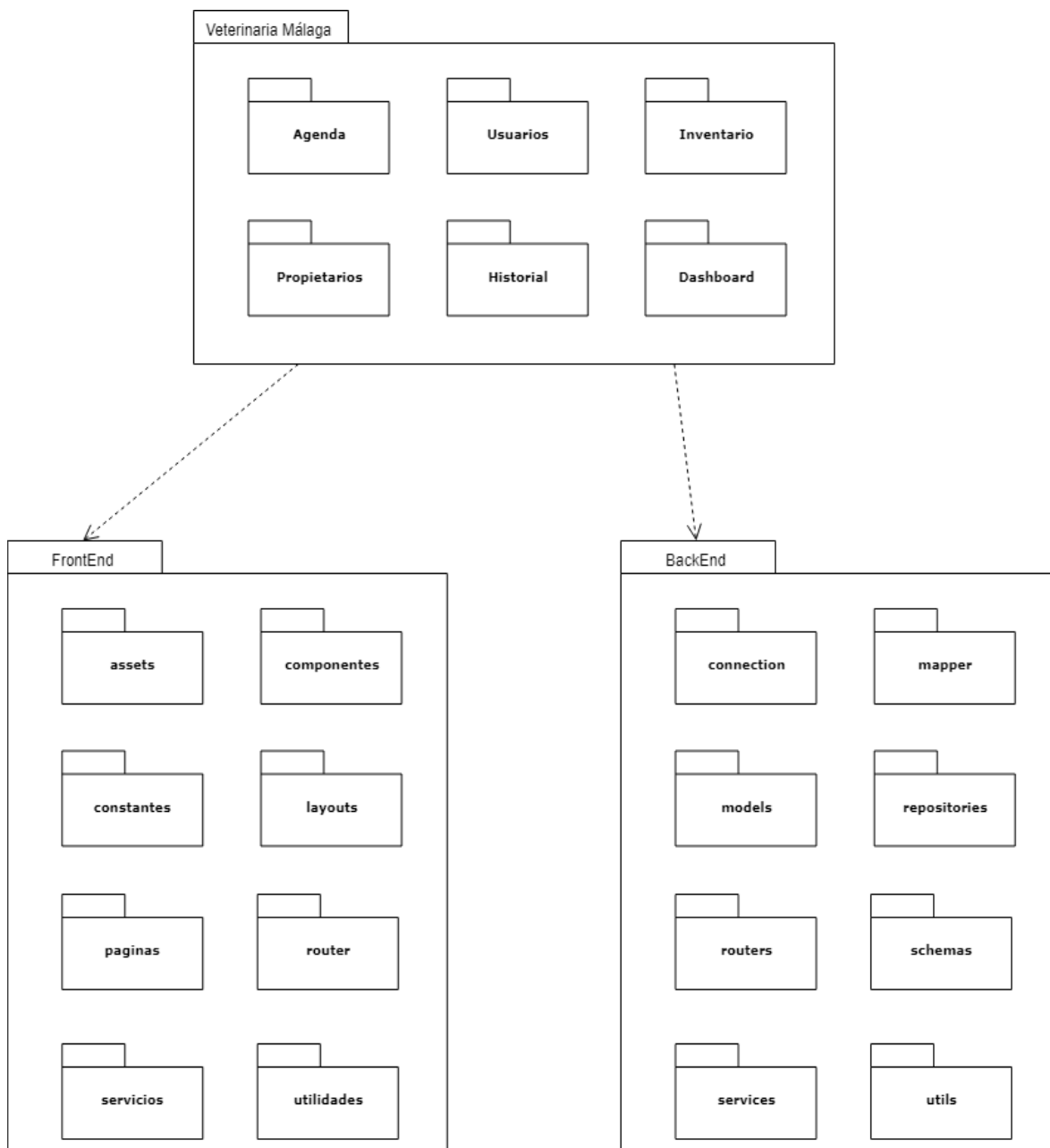


Figura 31: Diagrama de Paquetes

Fuente: Elaboración propia

II.1.1.10.3 Diagramas de Casos de Uso

- Diagrama de Casos de Uso General del Sistema

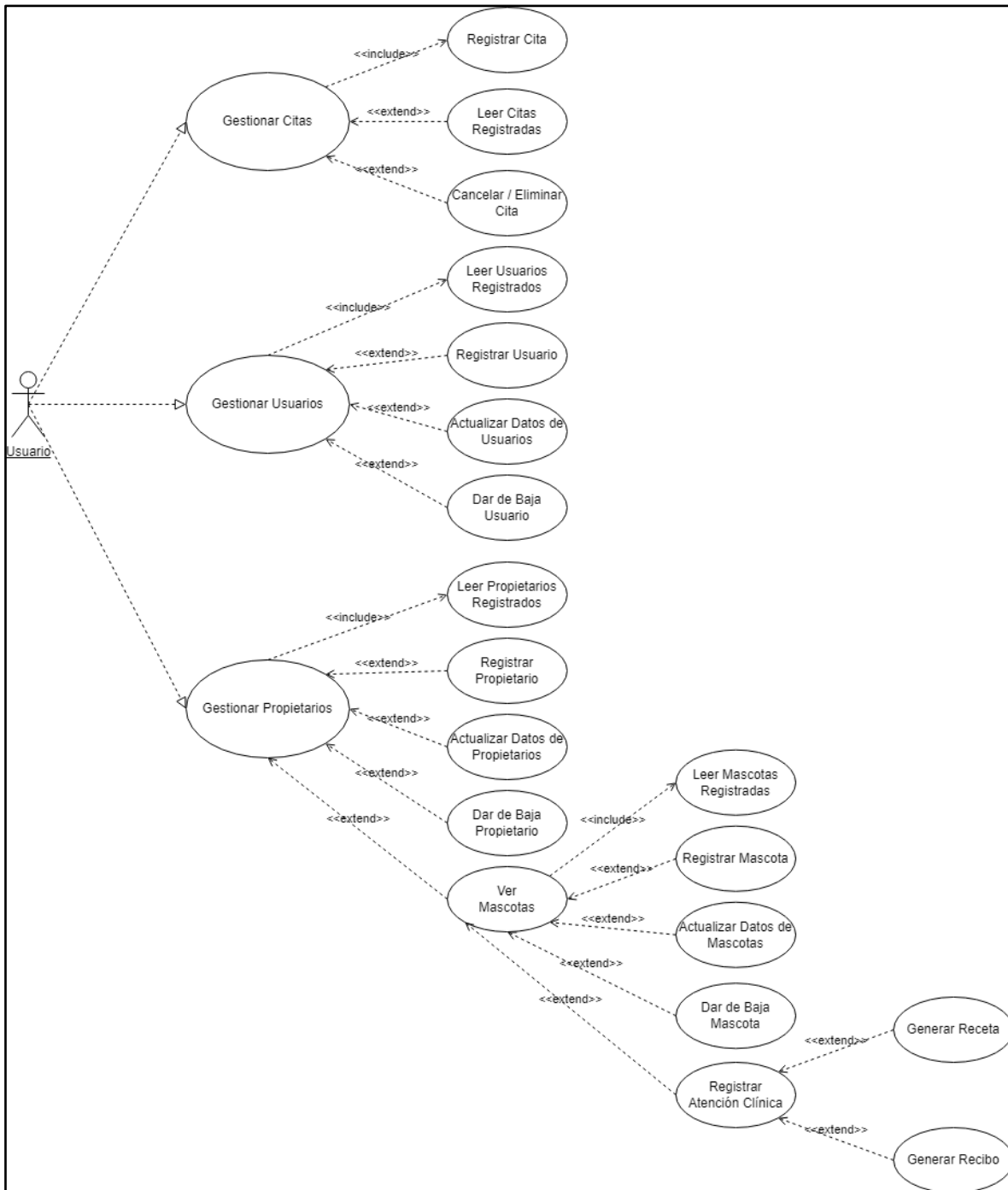


Figura 32: Diagrama General de Casos de Uso del sistema (1)

Fuente: Elaboración propia

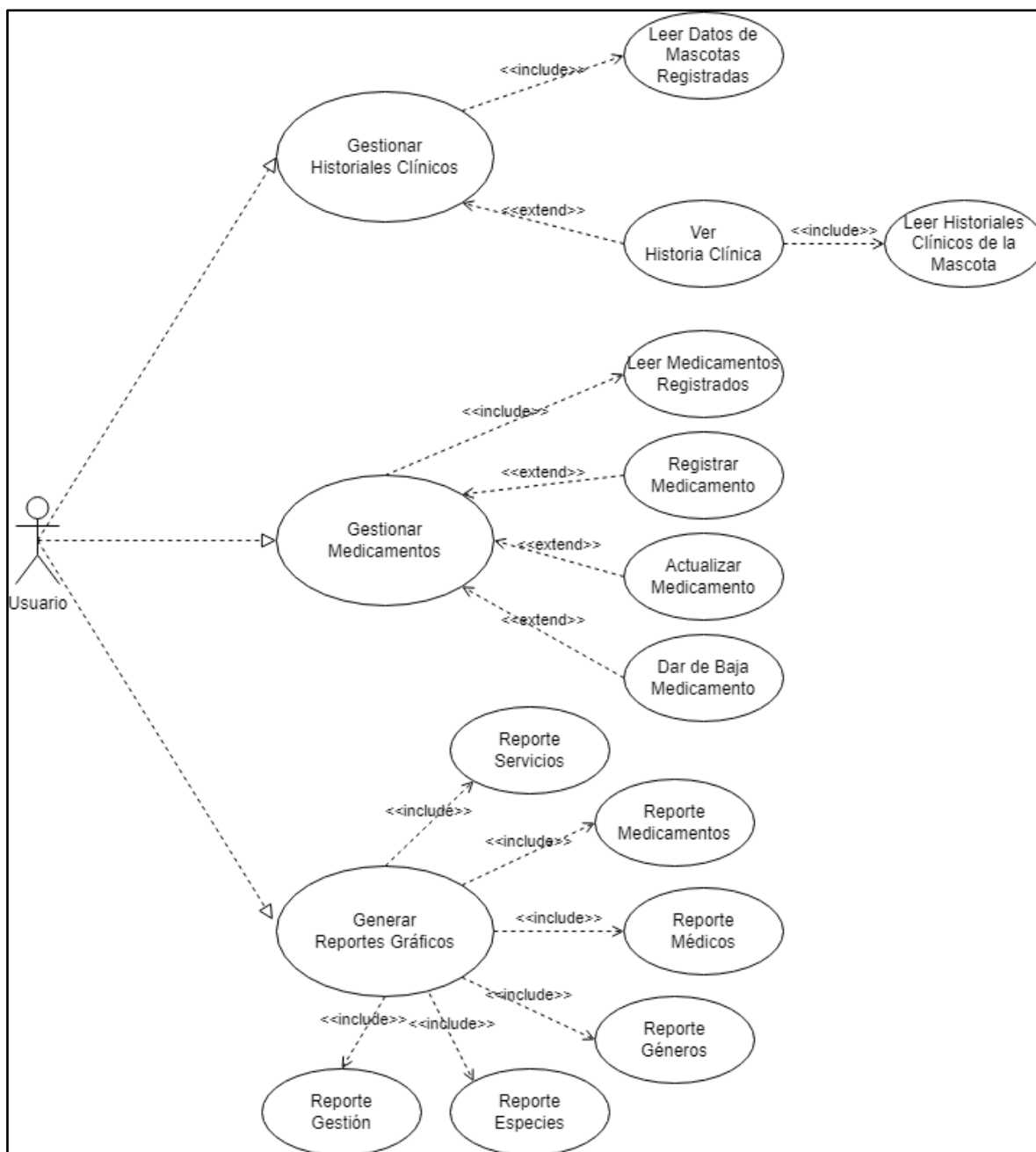


Figura 33: Diagrama General de Casos de Uso del sistema (2)

Fuente: Elaboración propia

- Diagrama de Casos de Uso Específico del Sistema

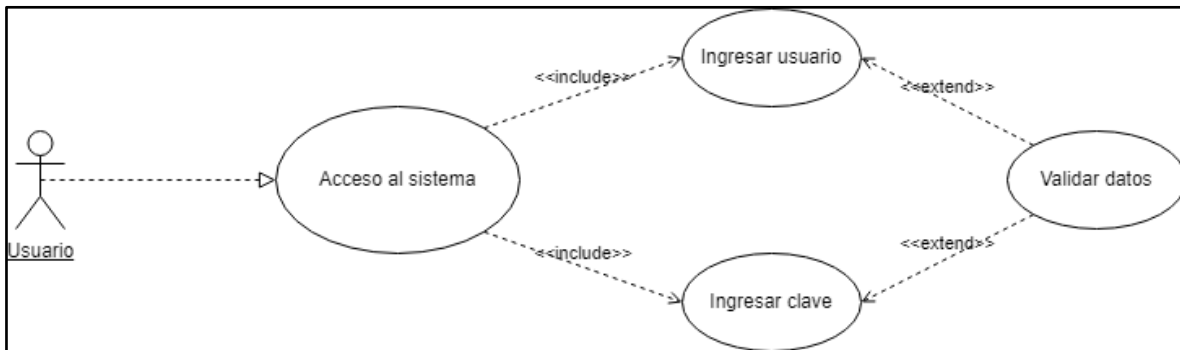


Figura 34: Acceso al Sistema

Fuente: Elaboración propia

Tabla 2: Descripción Caso de Uso - Acceso al Sistema

Nombre del caso de uso	Acceso al Sistema
Descripción	Permite ingresar al sistema web
Actores	Administrador, Médico
Escenarios	Cada actor tiene la capacidad de acceder al sistema con el propósito de llevar a cabo los procesos asignados a su rol específico.

Fuente: Elaboración propia

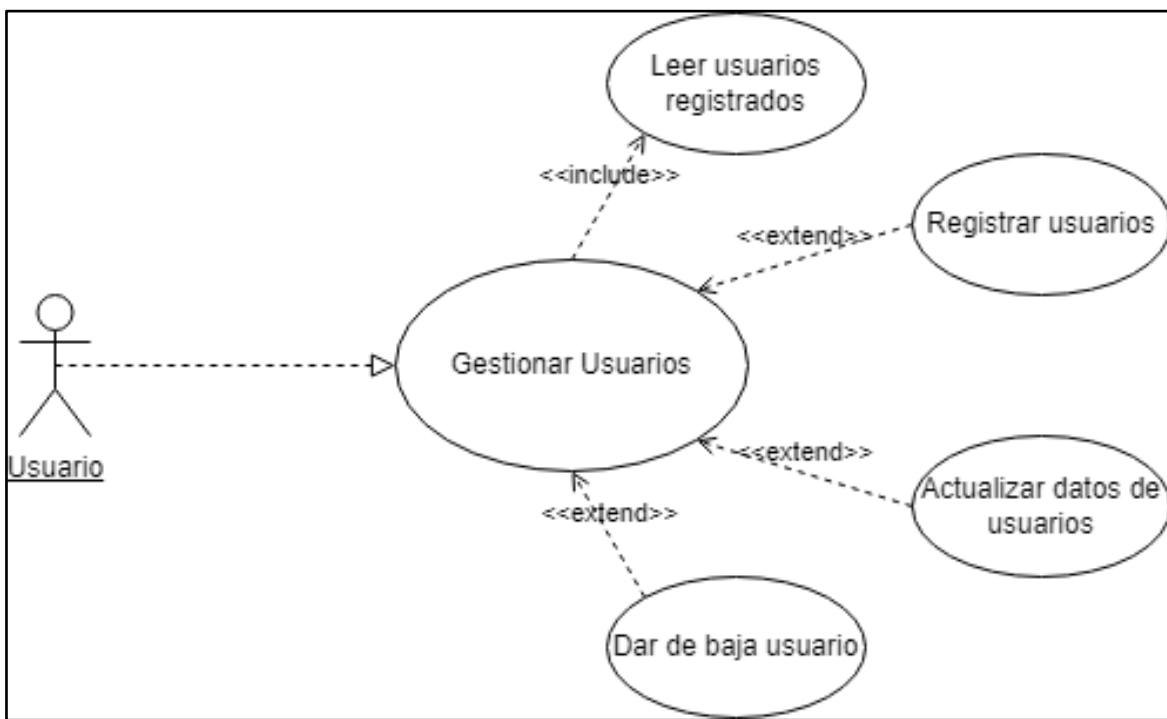


Figura 35: Gestionar Usuarios

Fuente: Elaboración propia

Tabla 3: Descripción Caso de Uso - Gestionar Usuarios

Nombre del caso de uso	Gestionar Usuarios
Descripción	Permite crear, modificar, desactivar registro y consultar los usuarios del sistema.
Actores	Administrador
Escenarios	El administrador tendrá la capacidad de crear un usuario con el rol adecuado, de modificar datos y también desactivar los registros de los usuarios ya registrados.

Fuente: Elaboración propia

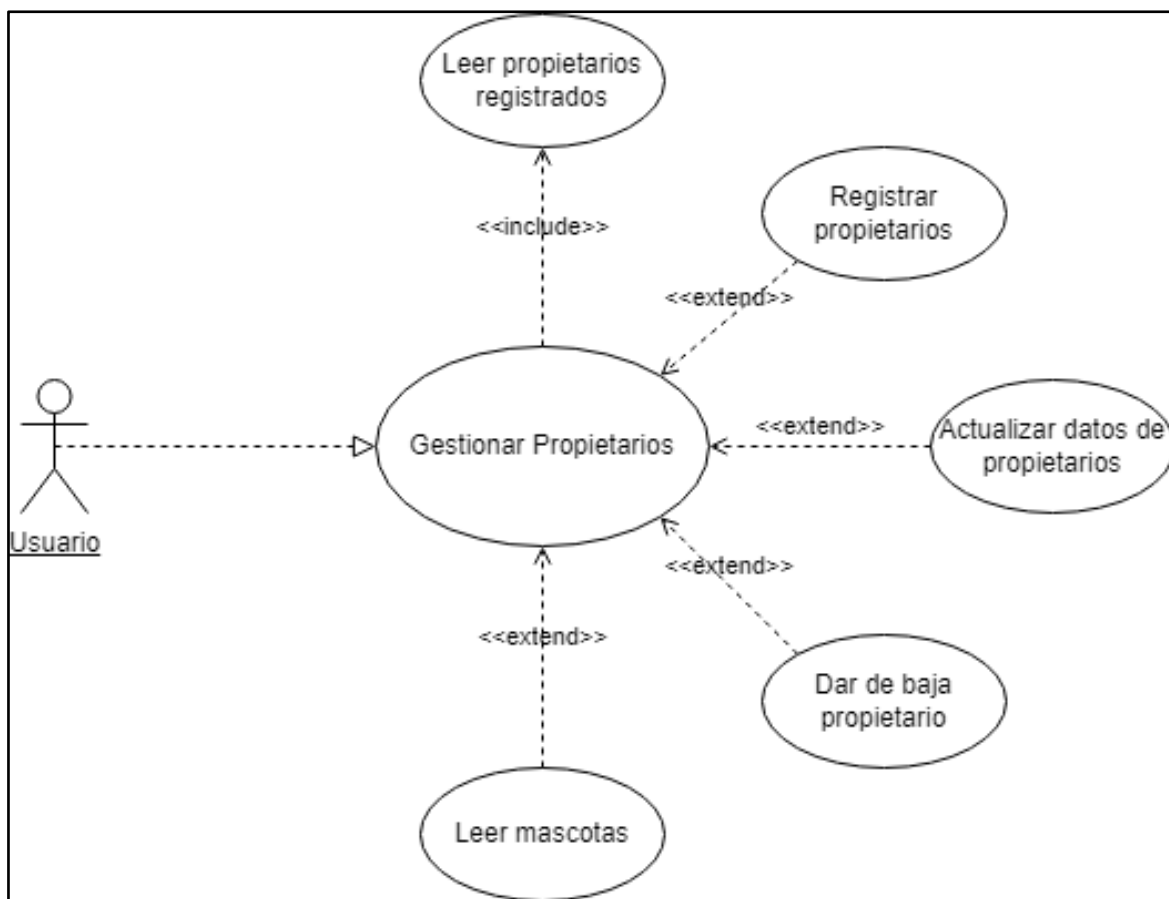


Figura 36: Gestionar Propietarios

Fuente: Elaboración propia

Tabla 4: Descripción Caso de Uso - Gestionar Propietarios

Nombre del caso de uso	Gestionar Propietarios
-------------------------------	------------------------

Descripción	Permite crear, modificar, desactivar registro y consultar los propietarios registrados en el sistema.
Actores	Administrador, Médico
Escenarios	Los actores tendrán la capacidad de crear, modificar datos y también desactivar los registros de los propietarios ya registrados.

Fuente: Elaboración propia

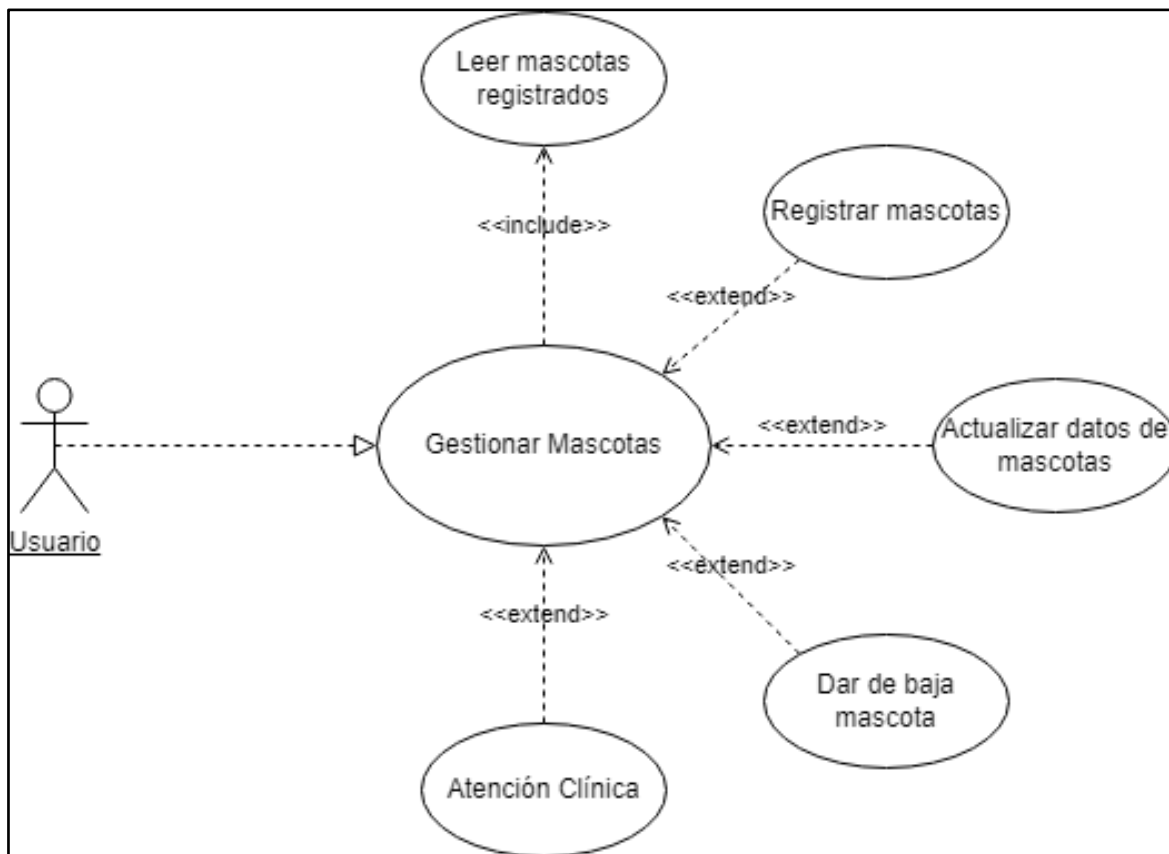


Figura 37: Gestionar Mascotas

Fuente: Elaboración propia

Tabla 5: Descripción Caso de Uso - Gestionar Mascotas

Nombre del caso de uso	Gestionar Mascotas
Descripción	Permite crear, modificar, desactivar registro y consultar las mascotas registradas en el sistema.
Actores	Administrador, Médico

Escenarios	Los actores tendrán la capacidad de crear, modificar datos y también desactivar los registros de las mascotas ya registradas.
-------------------	---

Fuente: Elaboración propia

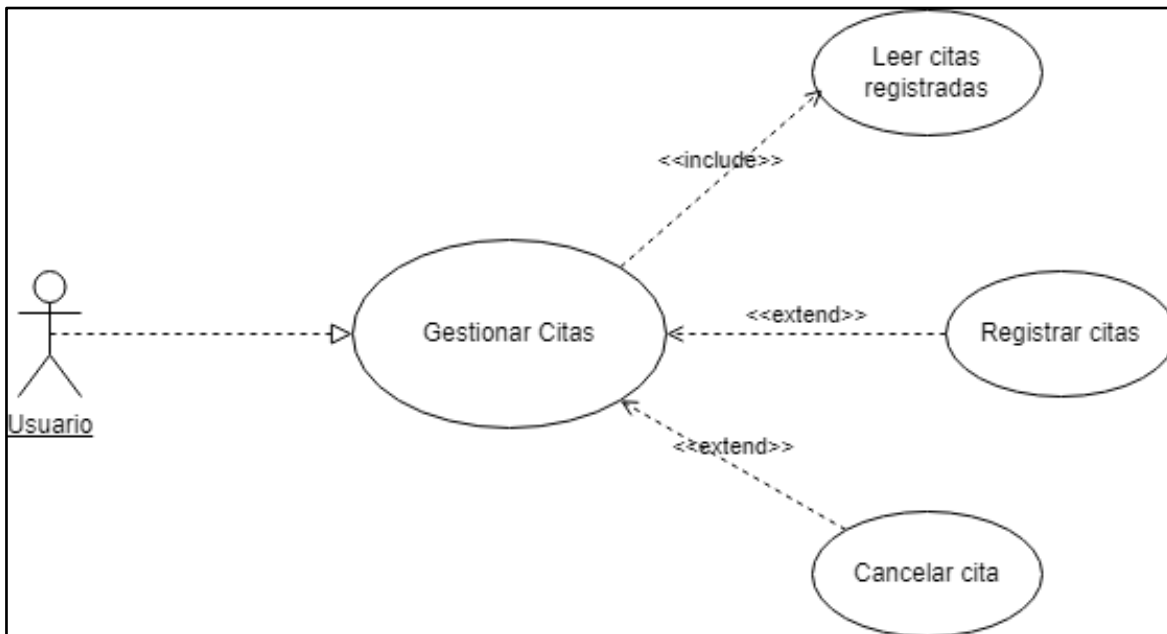


Figura 38: Gestionar Citas

Fuente: Elaboración propia

Tabla 6: Descripción Caso de Uso - Gestionar Citas

Nombre del caso de uso	Gestionar Citas
Descripción	Permite crear, cancelar y consultar las citas agendadas en el sistema.
Actores	Administrador, Médico
Escenarios	Los actores tendrán la capacidad de crear y cancelar citas en el sistema.

Fuente: Elaboración propia

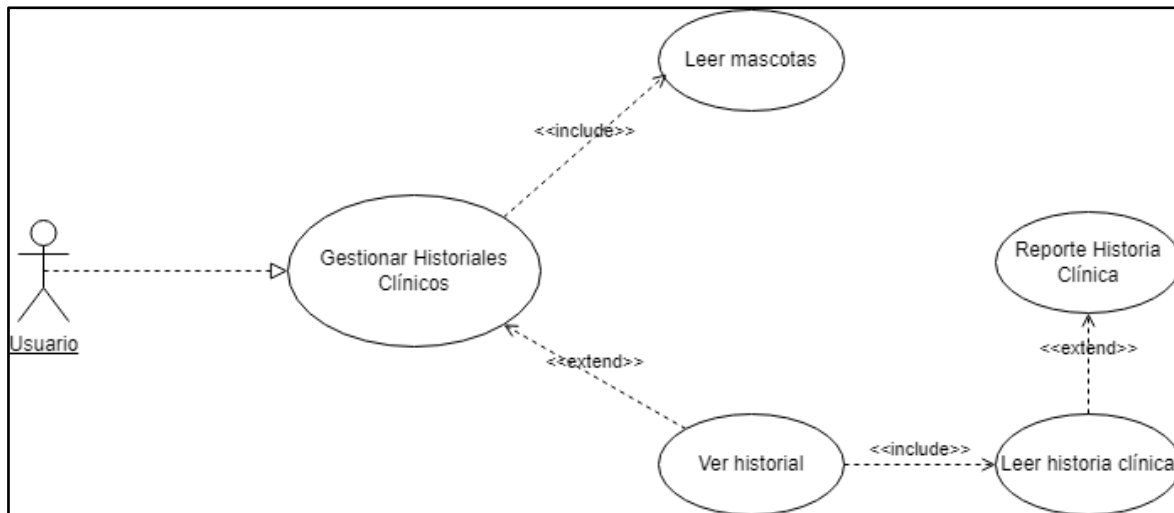


Figura 39: Historiales Clínicos

Fuente: Elaboración propia

Tabla 7: Descripción Caso de Uso - Historiales Clínicos

Nombre del caso de uso	Historiales Clínicos
Descripción	Permite consultar el historial clínico de las mascotas.
Actores	Administrador, Médico
Escenarios	Los actores tendrán la capacidad de consultar el historial clínico de cada una de las mascotas que han sido atendidas en la veterinaria y generar reporte del historial clínico de la mascota según las fechas requeridas.

Fuente: Elaboración propia

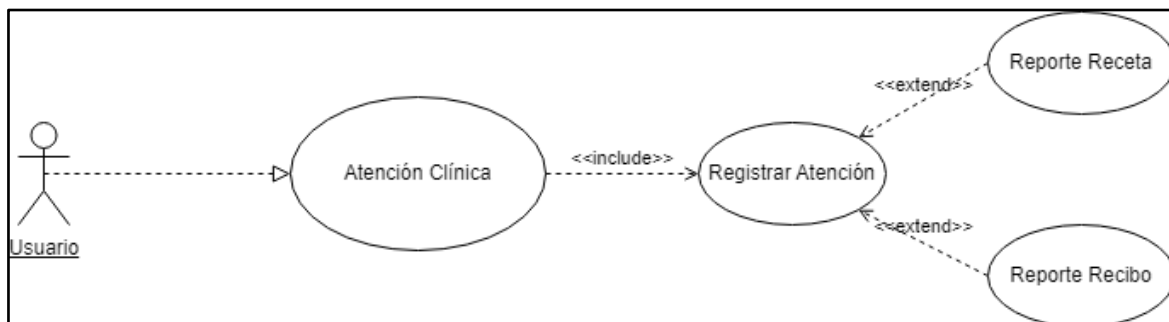


Figura 40: Atención Clínica

Fuente: Elaboración propia

Tabla 8: Descripción Caso de Uso - Atención Clínica

Nombre del caso de uso	Atención Clínica
Descripción	Permite el registro de las atenciones clínicas de acuerdo a la mascota seleccionada.
Actores	Administrador, Médico
Escenarios	Los actores tendrán la capacidad de crear una atención clínica, y generar reportes como receta y recibo e imprimirlos en caso de ser requeridos o guardarlos en PDF.

Fuente: Elaboración propia

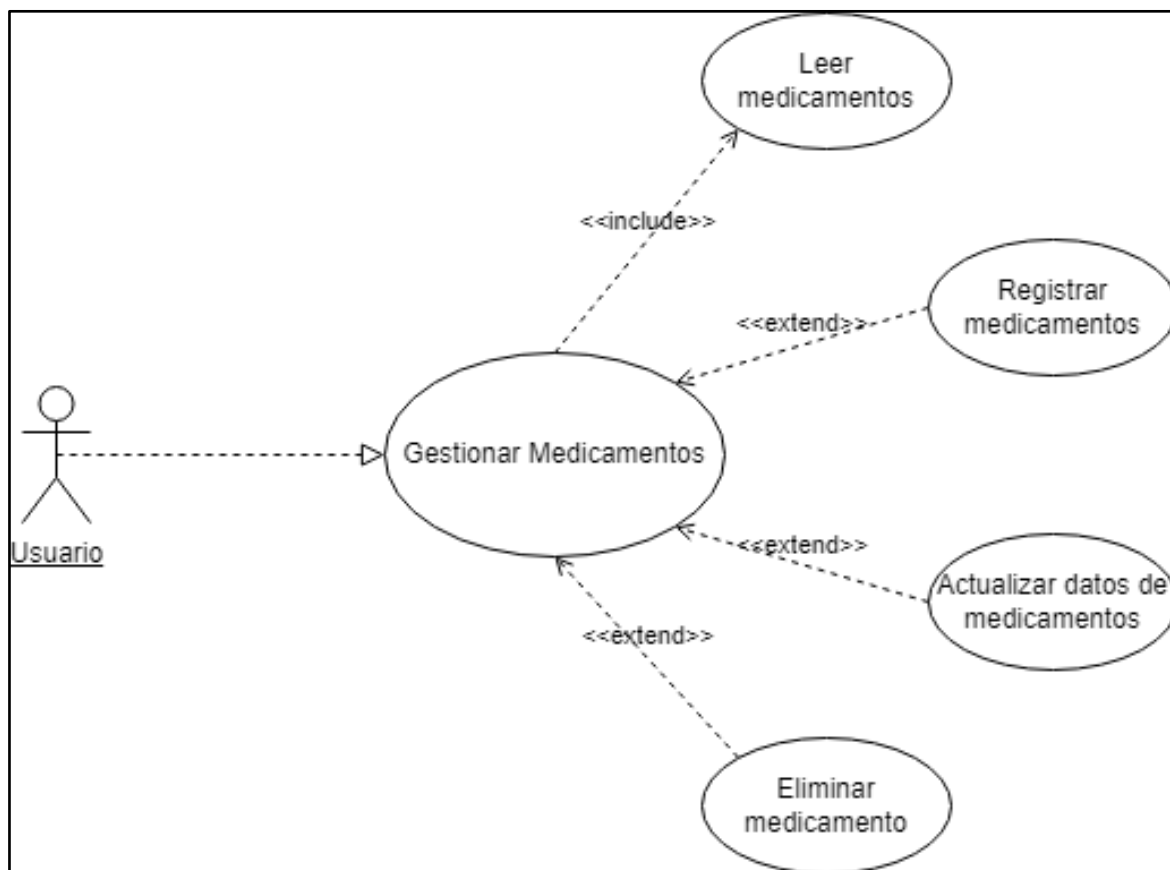


Figura 41: Gestionar Medicamentos

Fuente: Elaboración propia

Tabla 9: Descripción Caso de Uso - Gestionar Medicamentos

Nombre del caso de uso	Gestionar Medicamentos
Descripción	Permite registrar, modificar y consultar el inventario de medicamentos de la veterinaria.
Actores	Administrador

Escenarios	El administrador tendrá la capacidad de registrar y consultar en el sistema el stock de medicamentos en la veterinaria.
-------------------	---

Fuente: Elaboración propia

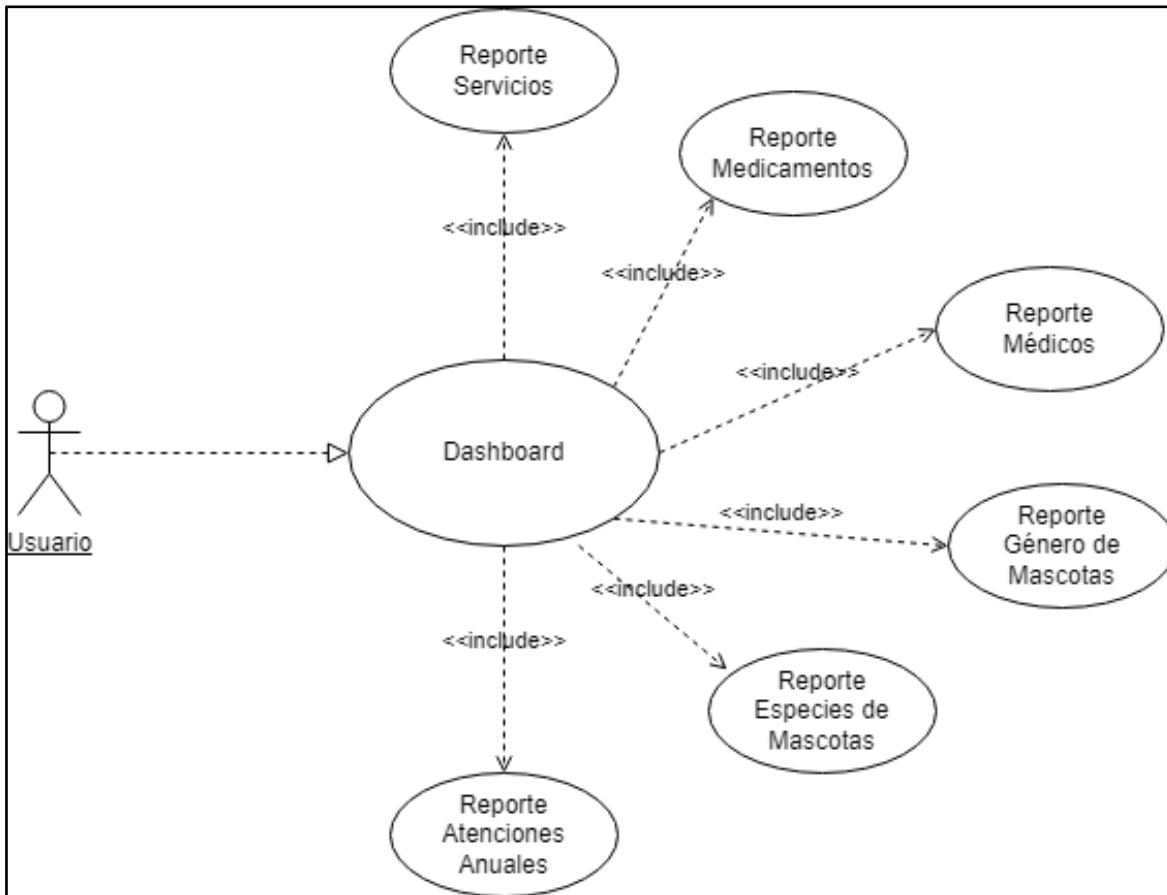


Figura 42: Dashboard (Reportes Gráficos)

Fuente: Elaboración propia

Tabla 10: Descripción Caso de Uso - Dashboard (Reportes Gráficos)

Nombre del caso de uso	Dashboard (Reportes Gráficos)
Descripción	Permite generar los reportes requeridos.
Actores	Administrador
Escenarios	El administrador tendrá la capacidad de generar los reportes previamente requeridos para realizar seguimiento a los procesos de la clínica.

Fuente: Elaboración propia

II.1.1.11 Modelado de datos

II.1.1.11.1 Diagrama de Clases

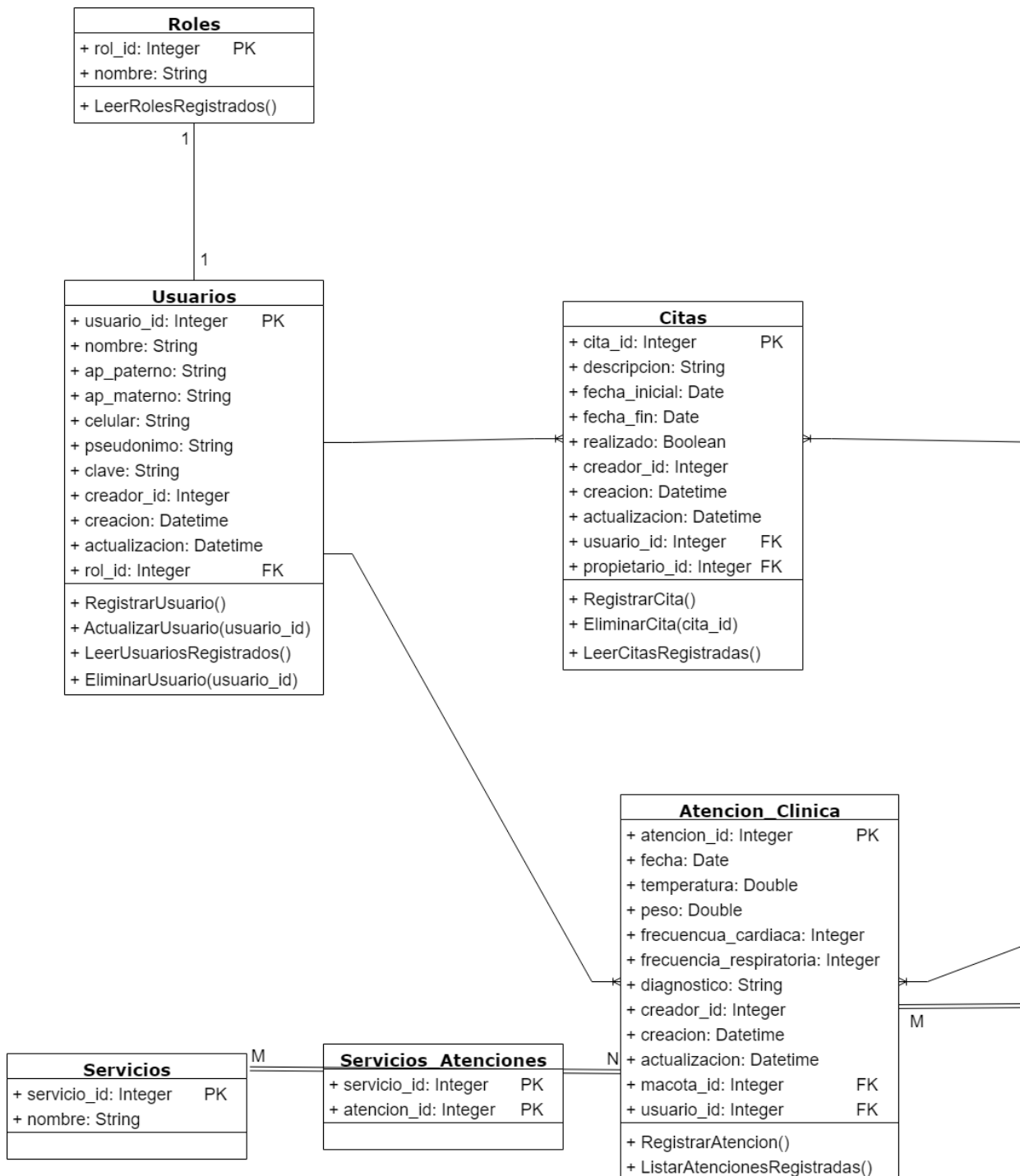
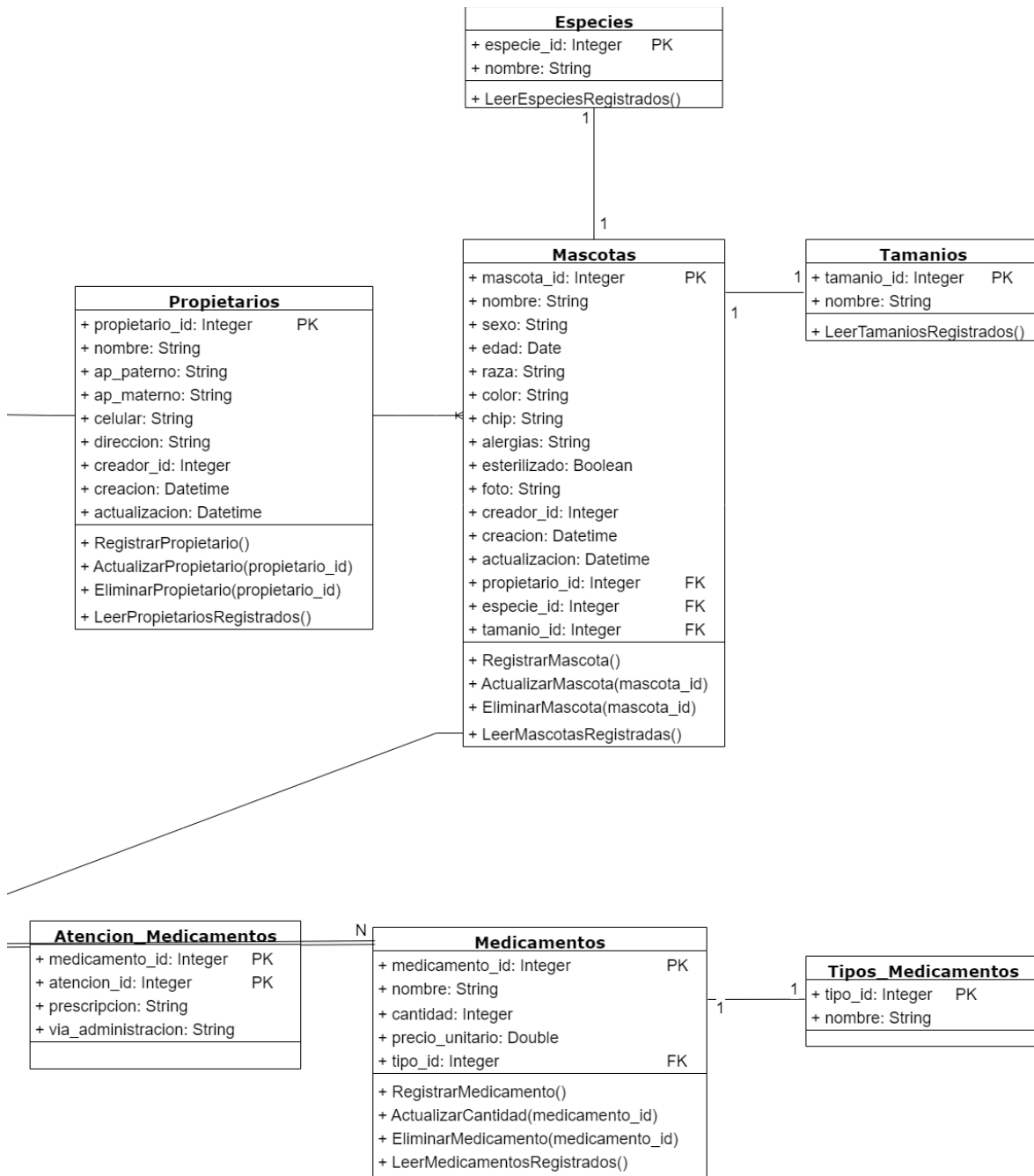


Figura 43: Diagrama de Clases

Fuente: Elaboración propia



II.1.1.11.2 Diagrama Entidad – Relación

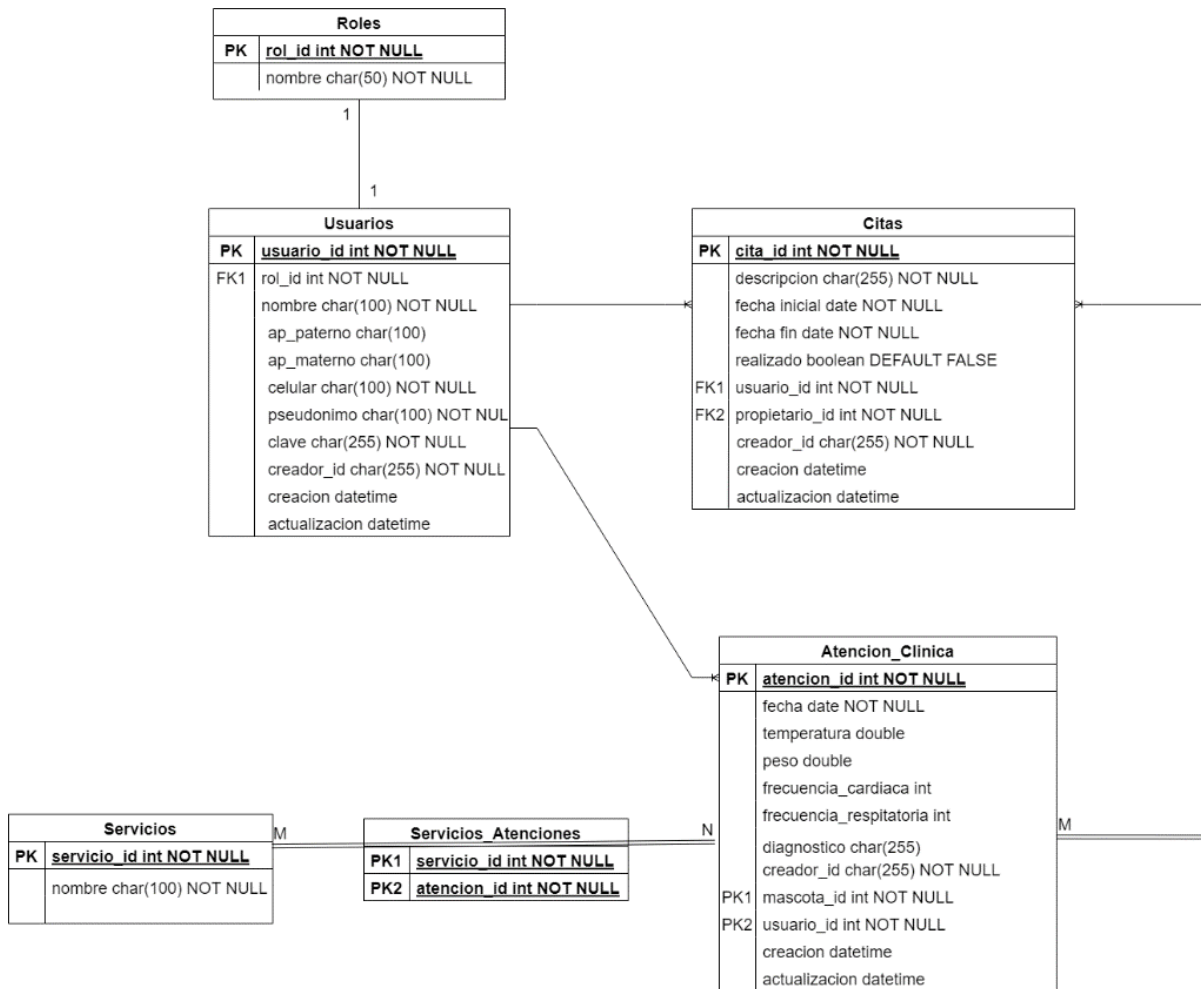
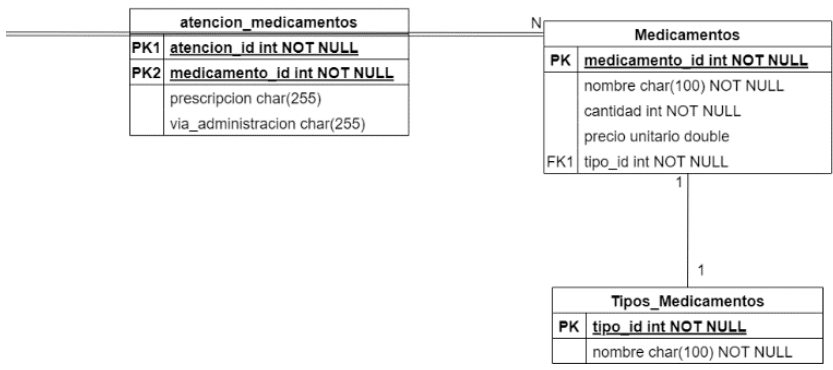
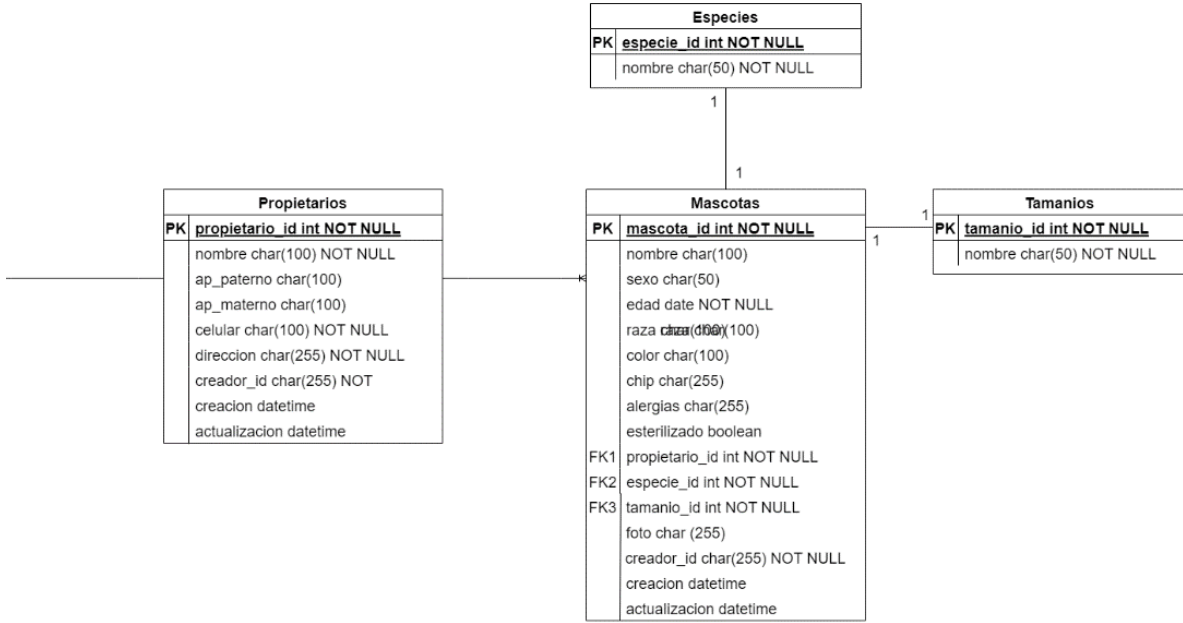


Figura 44: Modelo Entidad Relación

Fuente: Elaboración propia



II.1.1.12 Diccionario de Datos

II.1.1.12.1 Descripción de Tablas

Tabla: roles. - Tabla catálogo de roles con los que cuenta la clínica.

Tabla 11: Roles

ATRIBUTO	TIPO	TAMAÑO	NULO	CLAVE PRINCIPAL	CLAVE FORÁNEA	DESCRIPCIÓN
rol_id	integer	4	No	Si		Identificador del rol
nombre	varchar	50	No			Nombre del rol

Fuente: Elaboración propia

Tabla: especies. – Tabla catálogo de especies de animales.

Tabla 12: Especies

ATRIBUTO	TIPO	TAMAÑO	NULO	CLAVE PRINCIPAL	CLAVE FORÁNEA	DESCRIPCIÓN
especie_id	integer	4	No	Si		Identificador de la especie
nombre	varchar	50	No			Nombre de la especie

Fuente: Elaboración propia

Tabla: tamanios. – Tabla catálogo de tamaños de animales.

Tabla 13: Tamanios

ATRIBUTO	TIPO	TAMAÑO	NULO	CLAVE PRINCIPAL	CLAVE FORÁNEA	DESCRIPCIÓN
tamano_id	integer	4	No	Si		Identificador del tamaño
nombre	varchar	50	No			Nombre del tamaño

Fuente: Elaboración propia

Tabla: tipos_medicamentos. – Tabla catálogo de tipos de medicamentos con los que trabaja la clínica.

Tabla 14: Tipos de Medicamentos

ATRIBUTO	TIPO	TAMAÑO	NULO	CLAVE PRINCIPAL	CLAVE FORÁNEA	DESCRIPCIÓN
tipo_id	integer	4	No	Si		Identificador del tipo de medicamento
nombre	varchar	100	No			Nombre del tipo de medicamento

Fuente: Elaboración propia

Tabla: servicios. – Tabla catálogo de servicios con los que cuenta la clínica.

Tabla 15: Servicios

ATRIBUTO	TIPO	TAMAÑO	NULO	CLAVE PRINCIPAL	CLAVE FORÁNEA	DESCRIPCIÓN
servicio_id	integer	4	No	Si		Identificador del servicio
nombre	varchar	100	No			Nombre del servicio

Fuente: Elaboración propia

Tabla: usuarios. – Almacena datos de los usuarios del sistema.

Tabla 16: Usuarios

ATRIBUTO	TIPO	TAMAÑO	NULO	CLAVE PRINCIPAL	CLAVE FORÁNEA	DESCRIPCIÓN
usuario_id	integer	4	No	Si		Identificador de usuario
rol_id	Integer	4	No		Si	Referencia a la tabla roles
nombre	varchar	100	No			Nombre del usuario
ap_paterno	varchar	100	Si			Apellido Paterno del usuario
ap_materno	varchar	100	Si			Apellido Materno del usuario
celular	varchar	100	No			Número de celular del usuario
pseudónimo	varchar	100	No			Nombre con el que el usuario ingresa al sistema
clave	varchar	100	No			Clave con la que el usuario ingresa al sistema
creador_id	varchar	255	No			Id del usuario que está realizando acciones en el sistema
creacion	datetime		Si			Fecha de creación de alguna acción realizada en el sistema
actualizacion	datetime		Si			Fecha de actualización de alguna acción realizada en el sistema

Fuente: Elaboración propia

Tabla: propietarios. – Almacena datos de los propietarios registrados en el sistema.

Tabla 17: Propietarios

ATRIBUTO	TIPO	TAMAÑO	NULO	CLAVE PRINCIPAL	CLAVE FORÁNEA	DESCRIPCIÓN
propietario_id	integer	4	No	Si		Identificador de propietario
nombre	varchar	100	No			Nombre del propietario
ap_paterno	varchar	100	Si			Apellido Paterno del propietario
ap_materno	varchar	100	Si			Apellido Materno del propietario
celular	varchar	100	No			Número de celular del propietario
direccion	varchar	255	No			Dirección del propietario
creador_id	varchar	255	No			Id del usuario que está realizando acciones en el sistema
creacion	datetime		Si			Fecha de creación de alguna acción realizada en el sistema
actualizacion	datetime		Si			Fecha de actualización de alguna acción realizada en el sistema

Fuente: Elaboración propia

Tabla: mascotas. – Almacena datos de las mascotas registradas en el sistema.

Tabla 18: Mascotas

ATRIBUTO	TIPO	TAMAÑO	NULO	CLAVE PRINCIPAL	CLAVE FORÁNEA	DESCRIPCIÓN
mascota_id	integer	4	No	Si		Identificador de mascota
nombre	varchar	100	Si			Nombre de la mascota
sexo	varchar	50	Si			Sexo de la mascota
edad	date		No			Fecha de nacimiento aproximada de la mascota
raza	varchar	100	Si			Raza de la mascota

color	varchar	100	Si			Color distintivo de la mascota
chip	varchar	255	Si			URL del chip
alergias	varchar	255	Si			Alergias de las mascotas
esterilizado	boolean	1	Si			Identifica si la mascota está esterilizada
foto	varchar	255	Si			URL de la foto que se desea almacenar
propietario_id	integer	4	No		Si	Referencia a la tabla propietarios
especie_id	integer	4	No		Si	Referencia a la tabla especies
tamano_id	integer	4	No		Si	Referencia a la tabla tamanos
creador_id	varchar	255	No			Id del usuario que está realizando acciones en el sistema
creacion	datetime		Si			Fecha de creación de alguna acción realizada en el sistema
actualizacion	datetime		Si			Fecha de actualización de alguna acción realizada en el sistema

Fuente: Elaboración propia

Tabla: citas. – Almacena las citas registradas en el sistema.

Tabla 19: Gestionar Citas

ATRIBUTO	TIPO	TAMAÑO	NULO	CLAVE PRINCIPAL	CLAVE FORÁNEA	DESCRIPCIÓN
cita_id	integer	4	No	Si		Identificador de cita
descripcion	varchar	255	No			Detalle de cita a agendar
fecha_inicial	date		No			Fecha y hora inicial de la cita
fecha_final	date		No			Fecha y hora en que finaliza la cita
realizado	boolean	1	No			Identifica si la cita fue realizada
usuario_id	integer	4	No		Si	Referencia a la tabla usuarios
propietario_id	integer	4	No		Si	Referencia a la tabla propietarios

creador_id	varchar	255	No			Id del usuario que está realizando acciones en el sistema
creacion	datetime		Si			Fecha de creación de alguna acción realizada en el sistema
actualizacion	datetime		Si			Fecha de actualización de alguna acción realizada en el sistema

Fuente: Elaboración propia

Tabla: medicamentos. – Almacena los medicamentos registrados en el sistema, su cantidad y su precio.

Tabla 20: Medicamentos

ATRIBUTO	TIPO	TAMAÑO	NULO	CLAVE PRINCIPAL	CLAVE FORÁNEA	DESCRIPCIÓN
medicamento_id	integer	4	No	Si		Identificador de medicamento
nombre	varchar	100	No			Nombre del medicamento
cantidad	integer		No			Stock del medicamento
precio_unitario	double		Si			Precio del medicamento
tipo_id	integer	4	No		Si	Referencia a la tabla tipo de medicamentos

Fuente: Elaboración propia

Tabla: atenciones_clinicas. – Almacena los datos recogidos en la atención clínica realizada a la mascota.

Tabla 21: Atenciones Clínicas

ATRIBUTO	TIPO	TAMAÑO	NULO	CLAVE PRINCIPAL	CLAVE FORÁNEA	DESCRIPCIÓN
atencion_id	integer	4	No	Si		Identificador de la atención clínica
fecha	date		No			Fecha actual
temperatura	double		Si			Temperatura de la mascota en grados

peso	double		Si			Peso de la mascota en kilogramos
frecuencia_cardiaca	integer		Si			Frecuencia cardiaca de la mascota en latidos por minuto
frecuencia_respiratoria	integer		Si			Frecuencia respiratoria de la mascota en respiraciones por minuto
diagnostico	varchar	255	Si			Descripción del diagnóstico de la mascota
mascota_id	integer		No		Si	Referencia a la tabla mascotas
usuario_id	integer		No		Si	Referencia a la tabla usuarios
creador_id	varchar	255	No			Id del usuario que está realizando acciones en el sistema
creacion	datetime		Si			Fecha de creación de alguna acción realizada en el sistema
actualizacion	datetime		Si			Fecha de actualización de alguna acción realizada en el sistema

Fuente: Elaboración propia

Tabla: atencion_medicamentos. – Tabla resultante de relación muchos a muchos.

Tabla 22: Atención Medicamentos

ATRIBUTO	TIPO	TAMAÑO	NULO	CLAVE PRINCIPAL	CLAVE FORÁNEA	DESCRIPCIÓN
atencion_id	integer	4	No	Si	Si	Referencia a la tabla atención clínica
medicamento_id	Integer	4	No	Si	Si	Referencia a la tabla medicamentos
prescripcion	Varchar	255	Si			Detalle de la receta médica
via_administracion	varchar	255	Si			Vía de administración del medicamento

Fuente: Elaboración propia

Tabla: servicios_atenciones. – Tabla resultante de relación muchos a muchos.

Tabla 23: Servicios Atenciones

ATRIBUTO	TIPO	TAMAÑO	NULO	CLAVE PRINCIPAL	CLAVE FORÁNEA	DESCRIPCIÓN
atencion_id	integer	4	No	Si	Si	Referencia a la tabla atención clínica
servicio_id	integer	4	No	Si	Si	Referencia a la tabla servicios

Fuente: Elaboración propia

II.1.2 Desarrollo del Componente Sistema

II.1.2.1 Fase 1 – Planificación

En esta fase se presentará la planificación general identificando los problemas que se tiene en la Clínica Veterinaria, elaborando el Product Backlog, formando el equipo Scrum e identificando las historias de usuario.

II.1.2.1.1 Identificación de Problemas

- Todos los procesos actualmente son realizados de forma manual, que en consecuencia ocasiona demora en los procesos.
- Debido a los documentos adjuntados manualmente, puede ocurrir una pérdida o algún duplicando perjudicando así al cliente y al personal encargado.

II.1.2.1.2 Planificar el Product Backlog

A continuación, se presenta la planificación del Product Backlog que explica las funciones que tendrá el sistema web. Se agrupan las funcionalidades dentro de Sprints. Para conocer la cantidad de Sprints, se agrupa en cada una, una cierta cantidad de historias de usuario con sus respectivas estimaciones y prioridades.

Cada Sprint tendrá una duración de 1 mes aproximadamente. Las funcionalidades de cada Sprint fueron identificadas en conjunto con el equipo Scrum y la dueña y médico de la Clínica Veterinaria que explicó las necesidades y problemática de la Clínica.

II.1.2.1.2.1 Sprint 1: Base de Datos y Acceso al Sistema

- Diseño de la Base de Datos.
- El usuario podrá acceder al sistema con un usuario y contraseña.

- El usuario podrá cerrar sesión estando dentro del sistema.
- El usuario tendrá acceso la interfaz general del sistema.

II.1.2.1.2.2 Sprint 2: Implementación de los Endpoints y pruebas en Swagger

- Se podrá realizar el registro de un Usuario, Mascota, Propietario y Citas a través de Swagger
- Se podrá realizar la actualización de un Usuario, Mascota, Propietario y Citas a través de Swagger
- Se podrá realizar la desactivación de un Usuario, Mascota, Propietario y Citas a través de Swagger
- Se podrá realizar la consulta de todos los Usuarios, Mascota, Propietario y Citas a través de Swagger
- El usuario podrá tener acceso a la interfaz de gestión Mascotas

II.1.2.1.2.3 Sprint 3: Administración de Usuarios, Mascotas, Propietarios, Citas,

Atenciones, Medicamentos y Reportes

- El usuario podrá agendar una cita en la fecha y hora deseada.
- El usuario podrá revisar las citas agendadas.
- El usuario podrá cancelar las citas que lo requieran.
- El usuario podrá agregar usuarios nuevos.
- El usuario podrá revisar a los usuarios registrados en el sistema.
- El usuario podrá modificar datos de usuarios ya registrados.
- El usuario podrá dar de baja algún usuario que ya no forme parte del equipo de la clínica.
- El usuario podrá agregar un propietario nuevo.
- El usuario podrá modificar los datos de los propietarios.
- El usuario podrá dar de baja a propietarios.
- El usuario podrá listar a todos los propietarios registrados en el sistema.
- El usuario podrá agregar mascotas nuevas a propietarios ya registrados.
- El usuario podrá modificar los datos de las mascotas.
- El usuario podrá dar de baja mascotas.

- El usuario podrá listar a todas las mascotas registradas, según su propietario, en el sistema.
- El usuario podrá listar los historiales clínicos de las mascotas atendidas en la clínica.
- El usuario podrá registrar atenciones clínicas a mascotas registradas en el sistema.
- El usuario podrá entregar una receta impresa con las indicaciones de los medicamentos recetados.
- El usuario podrá recibir un recibo impreso del monto de los medicamentos administrados o recetados.
- El usuario podrá agregar medicamentos nuevos.
- El usuario podrá revisar a los medicamentos registrados en el sistema.
- El usuario podrá modificar algunos datos de medicamentos ya registrados.
- El usuario podrá eliminar algún medicamento que ya no se utilice en la clínica.

II.1.2.1.3 Definir roles y definir el equipo Scrum

En la siguiente tabla se muestra el nombre, teléfono de contacto y rol de cada persona que conformará el equipo Scrum.

Tabla 24: Tabla de asignación de roles para la elaboración del Sistema

Persona	Contacto	Rol
Ing. Silvana Paz		Scrum Master, Product Owner
Est. Rocío Ximena Cruz Málaga	69326862	Equipo Scrum

Fuente: Elaboración propia

II.1.2.1.4 Determinar Requerimientos (Product Backlog)

En la siguiente tabla se muestra el Product Backlog que contiene el código de las historias de usuario y su respectivo nombre.

Tabla 25: Tabla Product Backlog

Nº	BACKLOG ITEM
HU01	Diseño de Base de Datos
HU02	Acceso al sistema
HU03	Registrar usuario

HU04	Leer usuarios registrados
HU05	Actualizar datos de usuarios
HU06	Dar de baja usuario
HU07	Registrar mascota
HU08	Leer mascotas registradas
HU09	Actualizar datos de mascotas
HU10	Dar de baja mascota
HU11	Registrar cita
HU12	Leer citas registradas
HU13	Cancelar/Eliminar cita
HU14	Registrar propietario
HU15	Leer propietarios registrados
HU16	Actualizar datos de propietarios
HU17	Dar de baja propietario
HU18	Leer historiales clínicos
HU19	Reporte historia clínica
HU20	Registrar atención clínica
HU21	Generar receta
HU22	Generar recibo
HU23	Registrar medicamento
HU24	Leer medicamentos registrados
HU25	Actualizar datos de medicamentos
HU26	Eliminar medicamento
HU27	Dashboard

Fuente: Elaboración propia

II.1.2.1.5 Priorizar Requerimientos (Product Backlog)

En la siguiente tabla se muestra la priorización del Product Backlog que contiene el código de las historias de usuario, su respectivo nombre, la estimación y su prioridad. La estimación está en horas que tomará desarrollar dicha historia de usuario y la prioridad está de acuerdo a la complejidad que ésta tenga.

Tabla 26: Tabla de Priorización del Product Backlog

Nº	BACKLOG ITEM	ESTIMACIÓN	PRIORIDAD
HU01	Diseño de base de datos	120	Alto
HU02	Acceso al sistema	120	Alto
HU03	Registrar usuario	120	Alto
HU04	Leer usuarios registrados	72	Medio
HU05	Actualizar datos de usuarios	120	Alto
HU06	Dar de baja usuario	48	Bajo
HU07	Registrar mascota	120	Alto

HU08	Leer mascotas registradas	72	Medio
HU09	Actualizar datos de mascotas	120	Alto
HU10	Dar de baja mascota	72	Medio
HU11	Registrar cita	120	Alto
HU12	Leer citas registradas	72	Medio
HU13	Cancelar/Eliminar cita	48	Bajo
HU14	Registrar propietario	120	Alto
HU15	Leer propietarios registrados	72	Medio
HU16	Actualizar datos de propietarios	120	Alto
HU17	Dar de baja propietario	72	Medio
HU18	Leer historiales clínicos	120	Alto
HU19	Reporte historia clínica	48	Bajo
HU20	Registrar atención clínica	120	Alto
HU21	Reporte de receta impresa	48	Bajo
HU22	Reporte de recibo impreso	48	Bajo
HU23	Registrar medicamento	120	Alto
HU24	Leer medicamentos registrados	72	Medio
HU25	Actualizar datos de medicamentos	120	Alto
HU26	Eliminar medicamento	48	Bajo
HU27	Dashboard	72	Medio

Fuente: Elaboración propia

Tabla 27: Elaboración y estimación de historias de usuario general

Sprint	ENUNCIADO DE LA HISTORIA DE USUARIO						CRITERIOS DE ACEPTACIÓN				
	Código	Historia	Descripción	Prioridad	Estimación	Horas reales	Pasos	Tareas	Entrada de datos	Salida de datos	Resultado / Comportamiento esperado
Sprint 1: Acceso al Sistema	HU01	Diseño de Base de Datos	Historia de usuario del diseño de Base de Datos	Alto	240	235	1. El usuario accede al sistema con una URL proporcionada. 2. El usuario visualiza una interfaz donde podrá ingresar usuario y contraseña. 3. El usuario podrá cerrar sesión en la parte inferior izquierda estando dentro del sistema.	1. Diseño del Modelo Relacional de la Base de Datos e implementación. 2. Crear la interfaz inicial para que el usuario se pueda loguear. 3. Realizar la validación de datos. 3. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos)	Usuario y contraseña. Ninguno		El usuario tendrá los datos necesarios en el sistema.
	HU02	Acceso al sistema	Historia de usuario que permite el acceso al sistema.	Alto	120	115	1. El usuario accede al submenú usuarios. 2. Se muestra una interfaz con el listado con todos los usuarios registrados. 3. El usuario accede a la funcionalidad para registrar. 4. Se muestra la interfaz para ingresar los datos a registrar	1. Crear la interfaz inicial de usuarios. 2. Crear la interfaz de registro de usuario. 3. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos)	Datos del usuario. Pop-up		El usuario podrá registrar un usuario de forma exitosa.
Sprint 2: Implementación de los Endpoints y pruebas en Swagger	HU03	Registrar usuario	Historia de usuario que permite el registro de un usuario nuevo.	Alto	120	125	1. El usuario accede al submenú usuarios. 2. Se muestra una interfaz con el listado con todos los usuarios registrados. 3. El usuario accede a la funcionalidad para registrar. 4. Se muestra la interfaz para ingresar los datos a registrar	1. Crear la interfaz inicial de usuarios. 2. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos)	Datos del usuario. Pop-up		El usuario podrá registrar un usuario de forma exitosa.
	HU04	Leer usuarios registrados	Historia de usuario que permite listar los usuarios registrados.	Medio	72	74	1. El usuario accede al submenú usuarios. 2. Se muestra una interfaz con el listado con todos los usuarios registrados. 3. El usuario puede realizar la búsqueda de otros usuarios a través de un buscador.	1. Crear la interfaz inicial de usuarios. 2. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos)	Datos del usuario. Listado		El usuario podrá listar a los usuarios registrados de forma exitosa.
	HU05	Actualizar datos de usuarios	Historia de usuario que permite la actualización de los datos de los usuarios.	Alto	120	120	1. El usuario accede al submenú usuarios. 2. Se muestra una interfaz con el listado con todos los usuarios registrados. 3. El usuario accede a la funcionalidad para actualizar. 4. Se muestra la interfaz para actualizar los datos	1. Adecuar la interfaz para actualizar registro de usuario. 2. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos)	Datos del usuario. Pop-up		El usuario podrá actualizar un usuario de forma exitosa.
	HU06	Dar de baja usuario	Historia de usuario que permite el dar de baja el registro de un usuario.	Bajo	48	52	1. El usuario accede al submenú de usuarios. 2. Se muestra una interfaz con el listado con todos los usuarios registrados. 3. El usuario accede a la funcionalidad para eliminar. 4. Se muestra un cuadro de confirmación.	1. Crear la interfaz inicial de usuarios. 2. Crear la interfaz de confirmación para dar de baja al usuario. 3. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos)	Datos del usuario. Pop-up		El usuario podrá dar de baja un usuario de forma exitosa.

Sprint	ENUNCIADO DE LA HISTORIA DE USUARIO					CRITERIOS DE ACEPTACIÓN					
	Código	Historia	Descripción	Prioridad	Estimación	Horas reales	Pasos	Tareas	Entrada de datos	Salida de datos	Resultado / Comportamiento esperado
	H U 07	Registrar mascota	Historia de usuario que permite el registro de una mascota nueva.	Alto	120	140	<ol style="list-style-type: none"> 1. El usuario accede al submenú mascotas dentro de propietarios. 2. Se muestra una interfaz con las mascotas registradas del propietario seleccionado. 3. El usuario accede a la funcionalidad para registrar. 4. Se muestra la interfaz para ingresar los datos a registrar. 	<ol style="list-style-type: none"> 1. Crear la interfaz inicial de mascotas. 2. Crear la interfaz de registro de mascotas. 3. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos) 	Datos de la mascota.	Pop-up	El usuario podrá registrar una mascota de forma exitosa
	H U 08	Leer mascotas registradas	Historia de usuario que permite listar las mascotas registradas.	Medio	72	120	<ol style="list-style-type: none"> 1. El usuario accede al submenú mascotas dentro de propietarios. 2. Se muestra una interfaz con el listado con todas las mascotas registradas del propietario seleccionado. 	<ol style="list-style-type: none"> 1. Crear la interfaz inicial de mascotas 2. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos) 	Datos de la mascota.	Listado	El usuario podrá listar las mascotas de cada propietario de forma exitosa
	H U 09	Actualizar datos de mascotas	Historia de usuario que permite la actualización de datos de las mascotas.	Alto	120	120	<ol style="list-style-type: none"> 1. El usuario accede al submenú mascotas dentro de propietarios. 2. Se muestra una interfaz con el listado con todas las mascotas registradas del propietario seleccionado. 3. El usuario accede a la funcionalidad para actualizar. 4. Se muestra la interfaz para actualizar los datos 	<ol style="list-style-type: none"> 1. Adecuar la interfaz para actualizar registro de mascotas. 2. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos) 	Datos de la mascota.	Pop-up	El usuario podrá actualizar una mascota de forma exitosa
	H U 10	Dar de baja mascota	Historia de usuario que permite el dar de baja el registro de una mascota.	Medio	72	78	<ol style="list-style-type: none"> 1. El usuario accede al submenú mascotas dentro de propietarios. 2. Se muestra una interfaz con el listado con todas las mascotas registradas del propietario seleccionado. 3. El usuario accede a la funcionalidad para eliminar mascota. 4. Se muestra un cuadro de confirmación. 	<ol style="list-style-type: none"> 1. Crear la interfaz inicial de mascotas. 2. Crear la interfaz de confirmación para dar de baja a la mascota. 3. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos) 	Datos de la mascota.	Pop-up	El usuario podrá dar de baja una mascota de forma exitosa
	H U 11	Registrar cita	Historia de usuario que permite el registro de una cita nueva.	Alto	120	130	<ol style="list-style-type: none"> 1. El usuario accede al submenú agenda. 2. Se muestra una interfaz con un calendario donde se muestran las citas registradas en el mes. 3. El usuario accede a la funcionalidad para registrar cita presionando en la fecha que desea agendar la cita. 4. Se muestra la interfaz para ingresar los datos a registrar. 	<ol style="list-style-type: none"> 1. Crear la interfaz inicial de agenda. 2. Crear la interfaz de registro de agenda. 3. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos) 	Datos de cita.	Pop-up	El usuario podrá registrar una cita de forma exitosa

Sprint	ENUNCIADO DE LA HISTORIA DE USUARIO					CRITERIOS DE ACEPTACIÓN					
	Código	Historia	Descripción	Prioridad	Estimación	Horas reales	Pasos	Tareas	Entrada de datos	Salida de datos	Resultado / Comportamiento esperado
Sprint 3: Administración y Reportes de Usuarios, Mascotas, Propietarios, Citas y Atenciones	HU 12	Leer citas registradas	Historia de usuario que permite listar las citas registradas en el calendario.	Medio	72	70	1. El usuario accede al submenú agenda. 2. Se muestra una interfaz con un calendario donde se muestran las citas registradas en el mes.	1. Crear la interfaz inicial de agenda. 2. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos)	Dato de cita.	Listado	El usuario podrá leer las citas registradas de forma extensa
	HU 13	Cancelar/eliminar cita	Historia de usuario que permite la cancelación de una cita seleccionada.	Bajo	48	55	1. El usuario accede al submenú agenda. 2. Se muestra una interfaz con un calendario donde se muestran las citas registradas en el mes. 3. El usuario accede a la funcionalidad para eliminar cita. 4. Se muestra un cuadro de confirmación.	1. Crear la interfaz inicial de agenda. 2. Crear la interfaz de confirmación para cancelar una cita. 3. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos)	Dato de cita.	Pop-up	El usuario podrá cancelar una cita de forma extensa
	HU 14	Registrar propietario	Historia de usuario que permite el registro de un propietario nuevo.	Alto	120	115	1. El usuario accede al submenú propietarios. 2. Se muestra una interfaz con el listado con todos los propietarios registrados. 3. El usuario accede a la funcionalidad para registrar. 4. Se muestra la interfaz para ingresar los datos a registrar	1. Crear la interfaz inicial de propietarios. 2. Crear la interfaz de registro de propietario. 3. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos)	Datos de propietario.	Pop-up	El usuario podrá registrar un propietario de forma extensa
	HU 15	Leer propietarios registrados	Historia de usuario que permite listar los propietarios registrados.	Medio	72	72	1. El usuario accede al submenú propietarios. 2. Se muestra una interfaz con el listado con todos los propietarios registrados. 3. El usuario puede realizar la búsqueda de cualquier propietario a través de un buscador.	1. Crear la interfaz inicial de propietarios. 2. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos)	Dato de propietario.	Listado	El usuario podrá leer los propietarios registrados de forma extensa
	HU 16	Actualizar datos de propietarios	Historia de usuario que permite la actualización de los datos de los propietarios.	Alto	120	120	1. El usuario accede al submenú propietarios. 2. Se muestra una interfaz con el listado con todos los propietarios registrados. 3. El usuario accede a la funcionalidad para actualizar. 4. Se muestra la interfaz para actualizar los datos	1. Adecuar la interfaz para actualizar registro de propietarios. 2. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos)	Dato de propietario.	Pop-up	El usuario podrá actualizar un propietario de forma extensa
	HU 17	Dar de baja propietario	Historia de usuario que permite el dar de baja el registro de un propietario.	Medio	72	70	1. El usuario accede al submenú de propietarios. 2. Se muestra una interfaz con el listado con todos los usuarios propietarios registrados. 3. El usuario accede a la funcionalidad para eliminar. 4. Se muestra un cuadro de confirmación.	1. Crear la interfaz inicial de propietarios. 2. Crear la interfaz de confirmación para dar de baja al propietario. 3. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos)	Dato de propietario.	Pop-up	El usuario podrá dar de baja un propietario de forma extensa

ENUNCIADO DE LA HISTORIA DE USUARIO		CRITERIOS DE ACEPTACIÓN									
Sprint	Código	Historia	Descripción	Prioridad	Estimación	Horas reales	Pasos	Tareas	Entrada de datos	Salida de datos	Resultado / Comportamiento esperado
	HU 18	Leer historiales clínicos	Historia de usuario que permite listar a todas las mascotas registradas en el sistema para poder acceder al historial de cada uno de ellos.	Alto	120	119	<ol style="list-style-type: none"> 1. El usuario accede al submenú de historias clínicas. 2. Se muestra una interfaz con el listado con todas las mascotas registradas. 3. El usuario accede a la funcionalidad para ver el historial seleccionando una mascota. 4. Se muestra una interfaz con una lista de las atenciones clínicas de la mascota seleccionada. 	<ol style="list-style-type: none"> 1. Crear la interfaz inicial de historiales clínicos. 2. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos) 	Datos de mascota.	Listado	El usuario podrá listar los historiales clínicos de las mascotas registradas de forma exitosa
	HU 19	Reporte Historia Clínica	Historia de usuario que permite generar un reporte impreso de la historia clínica de cualquier paciente.	Medio	72	68	<ol style="list-style-type: none"> 1. El usuario accede al submenú de historias clínicas. 2. Se muestra una interfaz con el listado con todas las mascotas registradas. 3. El usuario accede a la funcionalidad para ver el historial seleccionando una mascota. 4. Se muestra una interfaz con una lista de las atenciones clínicas de la mascota seleccionada. 5. Se selecciona el ícono de imprimir para generar el reporte. 	<ol style="list-style-type: none"> 1. Crear la interfaz con los datos de las historias clínicas. 2. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos) 	Datos de mascota y de la atención clínica.	Ficha de información	El usuario podrá generar el reporte de la mascota que elija
	HU 20	Registrar atención clínica	Historia de usuario que permite registrar las atenciones clínicas de las mascotas registradas.	Alto	120	150	<ol style="list-style-type: none"> 1. El usuario accede al submenú mascotas dentro de propietarios. 2. Se muestra una interfaz con el listado con todas las mascotas registradas. 3. El usuario accede a la funcionalidad para ver registrar atención clínica. 4. Se muestra una interfaz para registrar una nueva atención. 	<ol style="list-style-type: none"> 1. Crear la interfaz de registro de atención clínica. 2. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos) 	Datos de atención clínica.	Pop-up	El usuario podrá registrar una atención clínica de forma exitosa
	HU 21	Reporte de receta impresa	Historia de usuario que permite generar el reporte impreso de la prescripción médica.	Bajo	48	50	<ol style="list-style-type: none"> 1. Justo al finalizar el llenado del registro de una atención clínica, se puede generar la receta. 2. Se procede a imprimir la receta. 	<ol style="list-style-type: none"> 1. Crear la interfaz de generado de receta. 2. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos) 	Datos de atención clínica.	Reporte pdf o impreso	El usuario podrá generar una receta y tendrá la posibilidad de imprimirla de forma exitosa
	HU 22	Reporte de recibo impreso	Historia de usuario que permite generar el reporte impreso de un recibo.	Bajo	48	64	<ol style="list-style-type: none"> 1. Justo al finalizar el llenado del registro de una atención clínica, se puede generar un recibo con el costo de los medicamentos administrados. 2. Se procede a imprimir el recibo en caso de ser requerido. 	<ol style="list-style-type: none"> 1. Crear la interfaz de registro de generado de recibo. 2. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos) 	Datos de atención clínica.	Reporte pdf o impreso	El usuario podrá generar un recibo y tendrá la posibilidad de imprimirlo de forma exitosa

Sprint	ENUNCIADO DE LA HISTORIA DE USUARIO					CRITERIOS DE ACEPTACIÓN					
	Código	Historia	Descripción	Prioridad	Estimación	Horas reales	Pasos	Tareas	Entrada de datos	Salida de datos	Resultado / Comportamiento esperado
	H U23	Registrar medicamento	Historia de usuario que permite el registro de un medicamento nuevo.	Alto	120	125	<ol style="list-style-type: none"> 1. El usuario accede al submenú inventario. 2. Se muestra una interfaz con el listado con todos los medicamentos registrados. 3. El usuario accede a la funcionalidad para registrar. 4. Se muestra la interfaz para ingresar los datos a registrar 	<ol style="list-style-type: none"> 1. Crear la interfaz inicial de medicamentos. 2. Crear la interfaz de registro de medicamento. 3. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos) 	Datos del medicamento.	Pop-up	El usuario podrá registrar un medicamento de forma exitosa.
	H U24	Leer medicamentos registrados	Historia de usuario que permite listar los medicamentos registrados.	Medio	72	74	<ol style="list-style-type: none"> 1. El usuario accede al submenú inventario. 2. Se muestra una interfaz con el listado con todos los medicamentos registrados. 3. El usuario puede realizar la búsqueda de otros medicamentos a tra vez de un buscador. 	<ol style="list-style-type: none"> 1. Crear la interfaz inicial de medicamentos. 2. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos) 	Datos del medicamentos	Listado	El usuario podrá listar a los medicamentos registrados de forma exitosa.
	H U25	Actualizar datos de medicamentos	Historia de usuario que permite la actualización de los datos de los medicamentos.	Alto	120	120	<ol style="list-style-type: none"> 1. El usuario accede al submenú inventario. 2. Se muestra una interfaz con el listado con todos los medicamentos registrados. 3. El usuario accede a la funcionalidad para actualizar. 4. Se muestra la interfaz para actualizar los datos 	<ol style="list-style-type: none"> 1. Adecuar la interfaz para actualizar registro de medicamento 2. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos) 	Datos del medicamento.	Pop-up	El usuario podrá actualizar un medicamento de forma exitosa.
	H U26	Eliminar medicamento	Historia de usuario que permite eliminar el registro de un medicamento.	Bajo	48	52	<ol style="list-style-type: none"> 1. El usuario accede al submenú de inventario. 2. Se muestra una interfaz con el listado con todos los medicamentos registrados. 3. El usuario accede a la funcionalidad para eliminar. 4. Se muestra un cuadro de confirmación. 	<ol style="list-style-type: none"> 1. Crear la interfaz inicial de medicamentos. 2. Crear la interfaz de confirmación para eliminar un medicamento. 3. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos) 	Datos del medicamento.	Pop-up	El usuario podrá eliminar un medicamento de forma exitosa
	H U27	Dashboard (Reportes)	Historia de usuario que permite generar reportes de varios datos generales del sistema.	Medio	72	80	<ol style="list-style-type: none"> 1. El usuario accede al submenú dashboard. 2. Se muestra una interfaz con el listado (gráfico) con los reportes. 3. El usuario puede navegar por la interfaz para observar los reportes generados. 	<ol style="list-style-type: none"> 1. Crear la interfaz de reportes (dashboard). 2. Desarrollo funcional (Análisis, diseño, codificación, pruebas, solución de defectos) 	Datos generales de la base de datos.	Reporte gráfico, pdf o impreso	El usuario podrá generar un listado de los reportes generales del sistema y tendrá la posibilidad de imprimirlo de forma exitosa

Fuente: Elaboración propia

II.1.2.2 Fase 2 – Desarrollo

En esta fase se comenzará con el desarrollo de la solución dando uso a los Sprints que se planificaron previamente; la estimación del Sprint Backlog; análisis y diseño, realización de interfaces, gráfico burndown; revisión del Sprint y resultado retrospectiva.

Se utilizará las siguientes tecnologías para el desarrollo del sistema: Python como lenguaje de programación principal; FastAPI como framework debido a su rapidez y gran uso en la actualidad y SQLAlchemy como gestor de Base de Datos.

II.1.2.2.1 Sprint 1: Base de Datos y Acceso al Sistema

II.1.2.2.1.1 Planificación del Sprint

El desarrollo del primer Sprint se llevó a cabo desde el 03/07/2023 al 28/07/2023. Con el objetivo de cumplir con la primera evaluación de la materia de Taller 3.

Comenzamos planificando el Sprint con la estimación de las historias de usuario que conforman el mismo. Seguidamente se detallan las tareas a realizar para el despliegue de cada historia de usuario, los criterios que se usan para la usabilidad y los criterios de aceptación que serán parte fundamental para cumplir con los objetivos planteados en este sprint. Posteriormente se procede con el desarrollo de las actividades planteadas en la planificación para implementar de forma exitosa las historias de usuario.

II.1.2.2.1.2 Estimación del Sprint Backlog

En la tabla a continuación se visualiza las historias de usuario que confirman el Sprint 1 y que fueron definidas dentro del Sprint Planning. Se establecen las prioridades, estimación y horas reales que serán utilizadas en el desarrollo de cada historia de usuario.

Tabla 28: Tabla de estimación del Sprint Backlog del Sprint 1

Código	Historia	Prioridad	Estimación	Horas reales
HU00	Diseño de Base de Datos	Alto	240	235
HU01	Acceso al sistema	Alto	120	115

Fuente: Elaboración propia

Detalle de cada historia de usuario:

HU01 – Diseño de Base de Datos	
Descripción: Se diseñó el modelo relacional de la Base de Datos pensando en cumplir todos los requisitos, haciendo uso de la teoría de normalización,	
Prioridad: Alto	Estimación: 240
Criterios de aceptación:	
<ul style="list-style-type: none"> ✓ DADO QUE deseo manejar los datos de la clínica CUANDO tengo acceso al sistema ENTONCES el sistema tiene que contener los datos en un formato correcto. 	
Tareas:	
<ul style="list-style-type: none"> ✓ Realizar el análisis, entendimiento y diseño de la historia de usuario. ✓ Realizar el modelo de datos. ✓ Diseñar los prototipos correspondientes. ✓ Realizar el desarrollo de la historia de usuario. ✓ Diseñar casos de prueba. ✓ Ejecutar los casos de prueba planteados. ✓ Solucionar los defectos encontrados. ✓ Desplegar la historia de usuario. 	
Definiciones adicionales:	
<ul style="list-style-type: none"> ✓ Utilizará SQLAlchemy para el manejo de datos en este proyecto. 	

Fuente: Elaboración propia

HU02 – Acceso al sistema	
Descripción: La historia de usuario describe la forma en la que el usuario ingresa al sistema de administración de datos de la Clínica Veterinaria Málaga. El ingreso será controlado por un nombre de usuario y una contraseña. También contará con la opción de cerrar sesión.	
Prioridad: Alto	Estimación: 120
Criterios de aceptación:	
<ul style="list-style-type: none"> ✓ DADO QUE soy un usuario encargado que desea ingresar al sistema de administración de datos de la Clínica CUANDO ingreso mi usuario y contraseña y presiono el botón de iniciar sesión ENTONCES el sistema me muestra la interfaz principal con los todos los menús ✓ DADO QUE soy un usuario encargado que desea ingresar al sistema de administración de datos de la Clínica, y deseo cerrar mi sesión CUANDO presiono el botón de cerrar de sesión 	

<p>ENTONCES el sistema me cierra la sesión y me manda la interfaz inicial de inicio de sesión.</p>
<p>Tareas:</p> <ul style="list-style-type: none"> ✓ Realizar el análisis, entendimiento y diseño de la historia de usuario. ✓ Realizar el modelo de datos. ✓ Diseñar el prototipo de acceso al sistema. ✓ Realizar el desarrollo de la historia de usuario. ✓ Diseñar casos de prueba. ✓ Ejecutar los casos de prueba planteados. ✓ Solucionar los defectos encontrados. ✓ Desplegar la historia de usuario
<p>Definiciones adicionales:</p> <ul style="list-style-type: none"> ✓ Al iniciar sesión el sistema redirige a la interfaz principal del sistema. Y mostrará el calendario de citas, denominado agenda, posicionado en el mes actual donde se visualizarán las citas ya registradas en sus respectivas fechas. ✓ Cuando el usuario o contraseña no son las correctas entonces el sistema me muestra una alerta indicando que existe algún error.

Fuente: Elaboración propia

II.1.2.2.1.3 Análisis y Diseño

A continuación, se presenta los diagramas de actividad de las historias descritas en el Backlog para el presente Sprint. No se detallará a profundidad los diagramas por ser una metodología ágil, lo mencionado anteriormente se encuentra dentro de la descripción de cada historia de usuario.

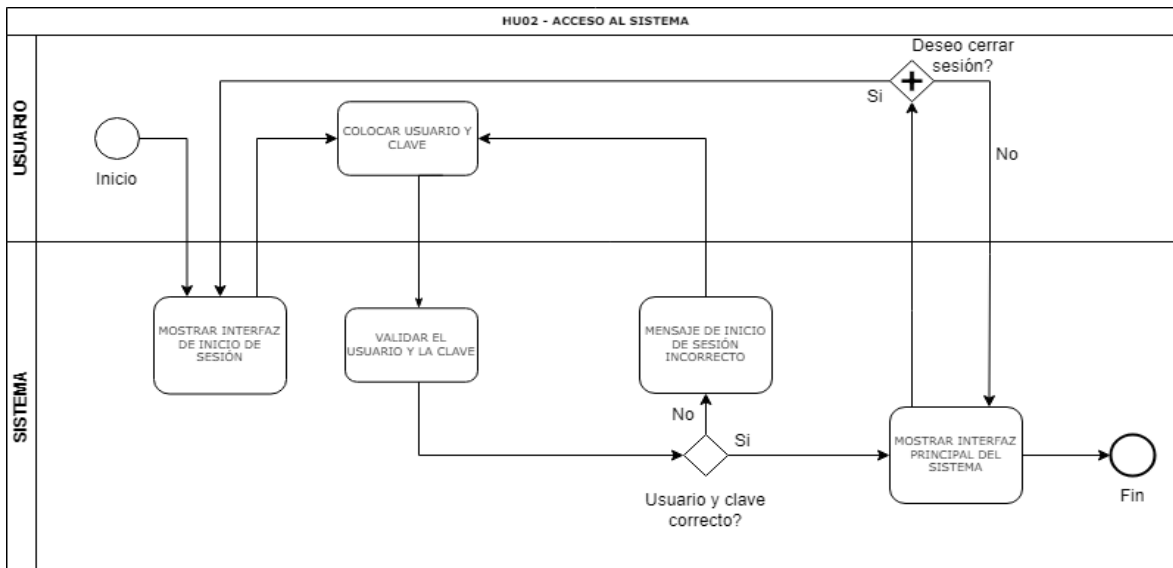


Figura 45: Diagrama de actividad HU02 – Acceso al sistema

Fuente: Elaboración propia

A continuación, se presenta los diagramas de secuencia de las historias descritas en el Backlog para el presente Sprint. No se detallará a profundidad los diagramas por ser una metodología ágil, lo mencionado anteriormente se encuentra dentro de la descripción de cada historia de usuario.

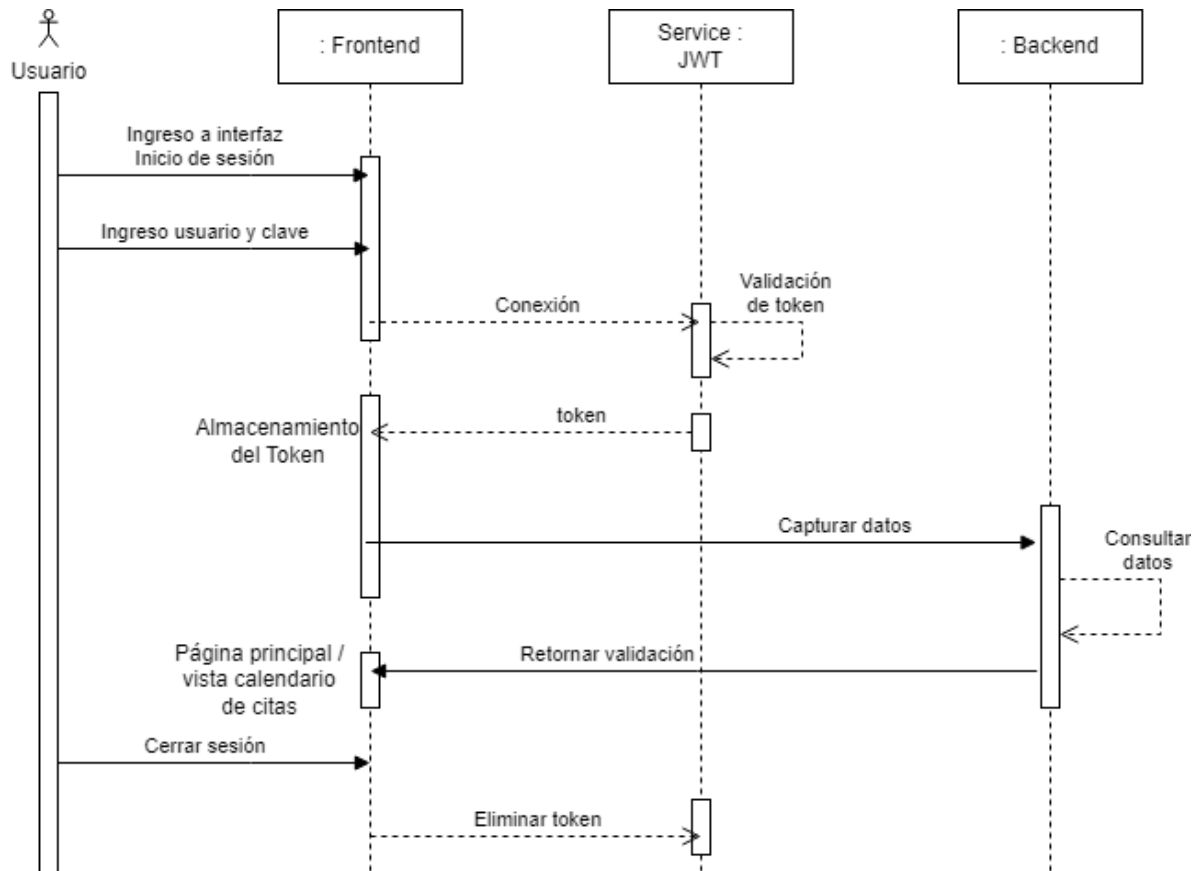


Figura 46: Diagrama de secuencia HU02 – Acceso al sistema

Fuente: Elaboración propia

II.1.2.2.1.4 Interfaces

A continuación, se muestran las interfaces de las historias de usuario para este Sprint 1:

HU01 – Base de Datos

En la siguiente imagen se muestra la herramienta Swagger, que nos permitió realizar las pruebas de los endpoints antes de realizar la conexión con el frontend.

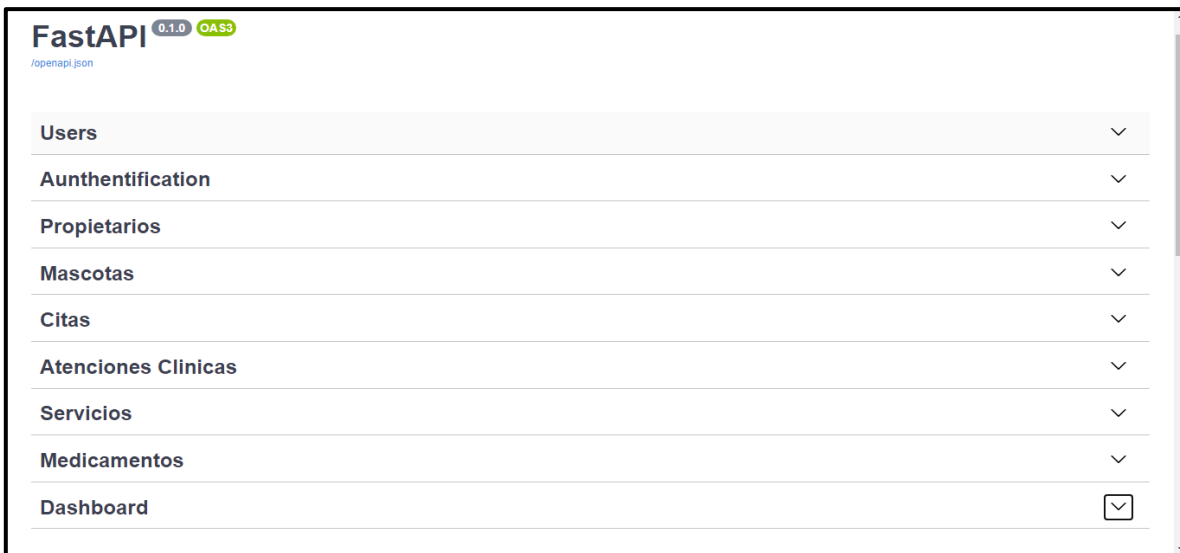


Figura 47: Endpoints en Swagger

Fuente: Elaboración propia

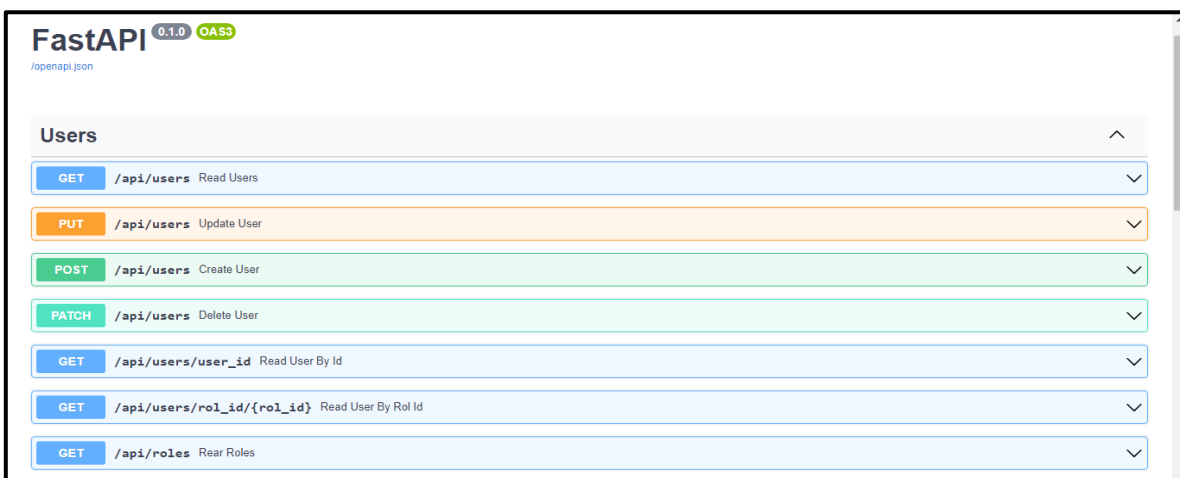


Figura 48: Usuarios Endpoints

Fuente: Elaboración propia

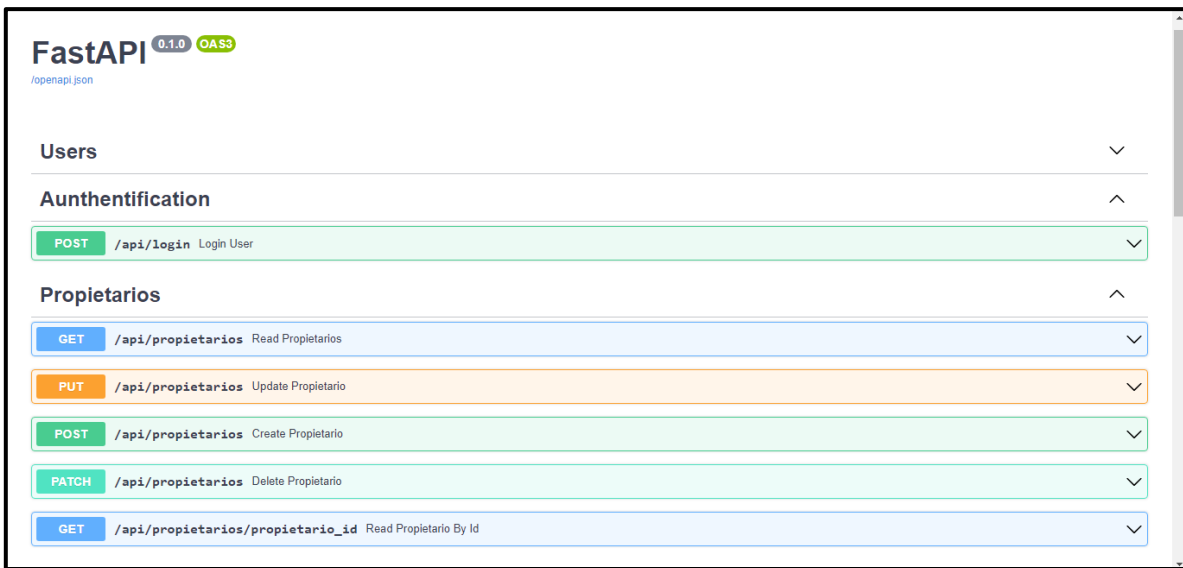


Figura 49: Autenticación y Propietarios Endpoints

Fuente: Elaboración propia

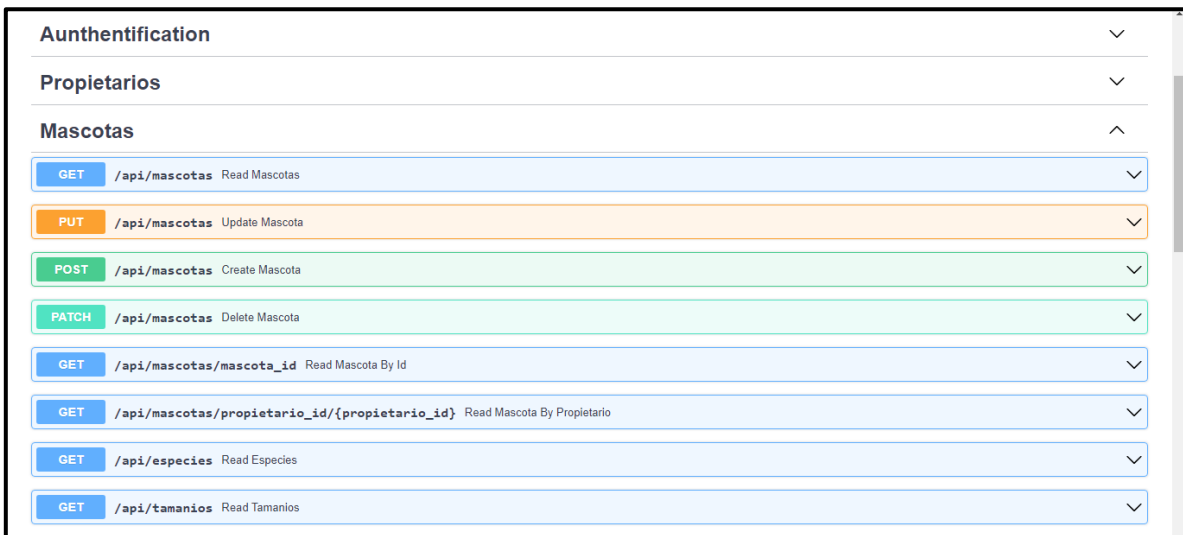


Figura 50: Mascotas Endpoints

Fuente: Elaboración propia

Citas	
GET	/api/citas Read Citas
POST	/api/citas Create Cita
PATCH	/api/citas Delete Cita
GET	/api/citas/cita_id Read Cita By Id
Atenciones Clínicas	
GET	/api/atenciones/atencion_id Read Atencion By Atencion Id
GET	/api/atenciones/{mascota_id} Historial By Mascota Id
POST	/api/atenciones Create Atencion
Servicios	
GET	/api/servicios Read Servicios

Figura 51: Citas, Atenciones Clínicas y Servicios Endpoints

Fuente: Elaboración propia

Medicamentos	
GET	/api/medicamentos Read Medicamentos
Dashboard	
GET	/api/dashboard/medicamentos/{mes} Total Medicamentos Mes
GET	/api/dashboard/servicios/{mes} Total Servicios Mes
GET	/api/dashboard/medicos/{mes} Total Atenciones Medicos Mes
GET	/api/dashboard/generos/{mes} Distribucion Generos Mes
GET	/api/dashboard/especies/{mes} Distribucion Especies Mes
GET	/api/dashboard/gestion/{fecha} Atenciones Por Gestion

Figura 52: Medicamentos y Dashboard Endpoints

Fuente: Elaboración propia

HU02 - Acceso al Sistema

En la siguiente imagen se muestra el formulario de inicio de sesión:

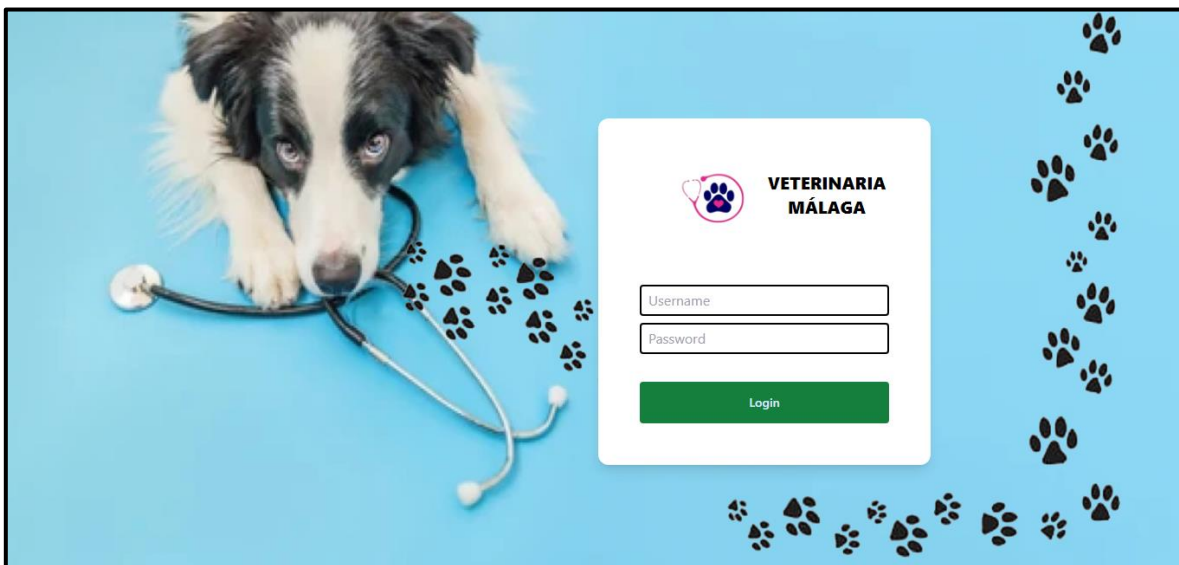


Figura 53: Interfaz HU02 – Acceso al sistema inicial

Fuente: Elaboración propia

En la siguiente imagen se muestra la validación de usuario y contraseña:

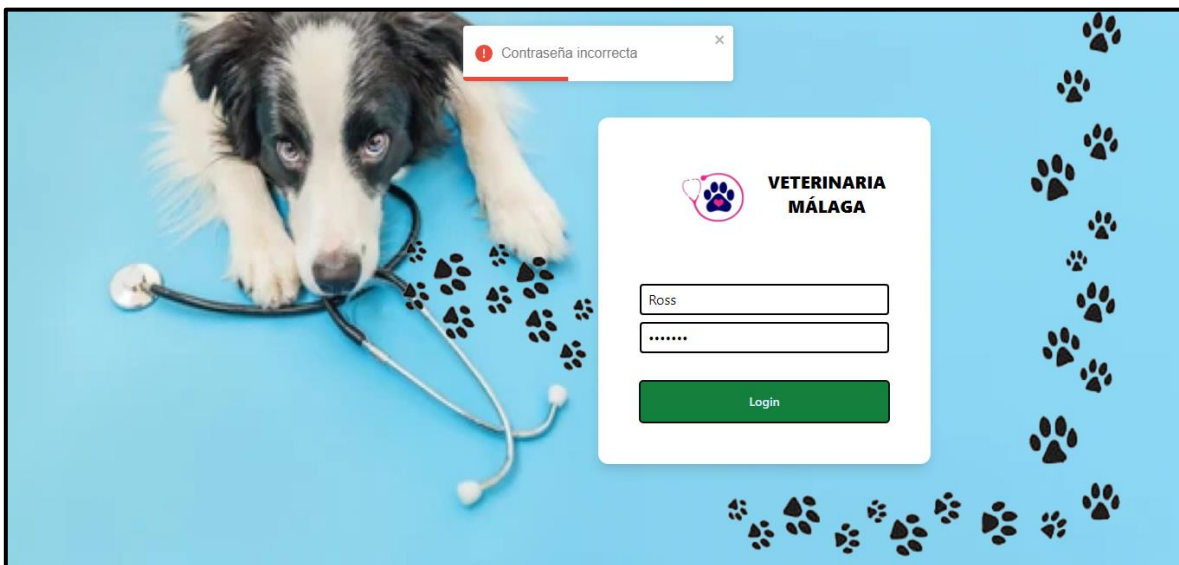


Figura 54: Interfaz HU02 – Validaciones de acceso al sistema

Fuente: Elaboración propia

En la siguiente imagen se muestra la interfaz principal del sistema luego de validar el inicio de sesión:



Figura 55: Interfaz HU02 – Ventana principal

Fuente: Elaboración propia

En la siguiente imagen se muestra la forma cómo el usuario cierra la sesión:

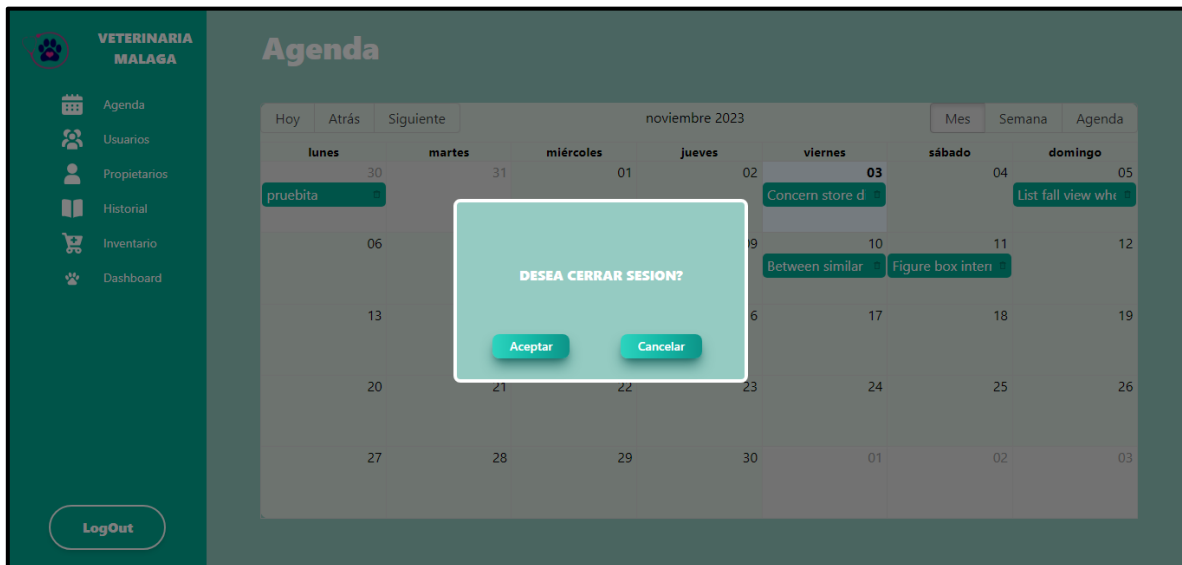


Figura 56: Interfaz HU02 – Cerrar sesión

Fuente: Elaboración propia

II.1.2.2.1.5 Gráfico Burn Down

En la siguiente gráfica se puede observar que en los primeros días del sprint las estimaciones iban de acuerdo al tiempo estimado. Durante la etapa intermedia del sprint se observa un desnivel positivo porque las historias se terminaban mucho más rápido que lo estimado. Sin embargo, en los últimos días del sprint se casi niveló con lo estimado.

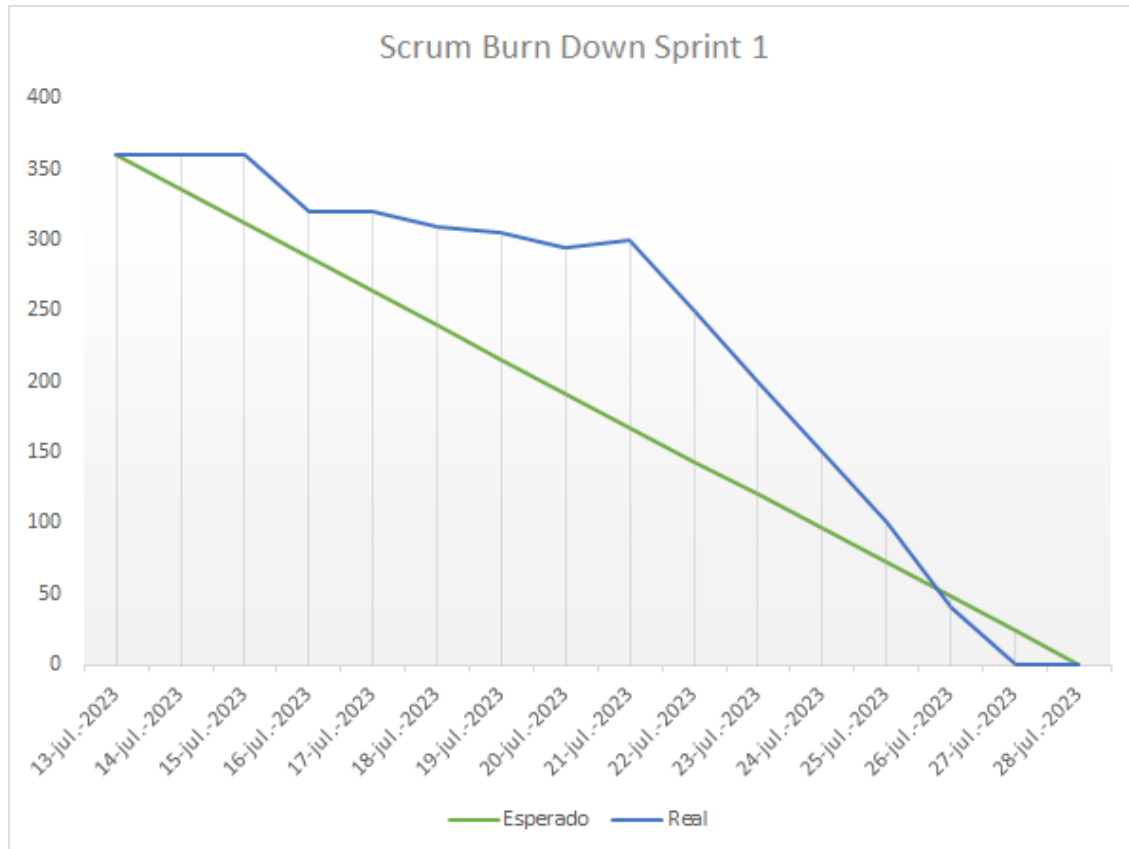


Figura 57: Gráfico Burn Down Sprint 1

Fuente: Elaboración propia

II.1.2.2.1.6 Revisión del Sprint (Sprint Review)

En la siguiente tabla se muestra el resultado que se obtuvo durante este primer sprint, de un total de dos historias implementadas, se aprobaron todas.

Tabla 29: Sprint Review

Código	Historia de usuario	Aprobada	Rechazada	Detalles
HU01	Diseño de Base de datos	X		(OK)
HU02	Acceso al sistema	X		(OK)

Fuente: Elaboración propia

II.1.2.2.2 Sprint 2: Implementación de los Endpoints y pruebas en Swagger

II.1.2.2.2.1 Planificación del Sprint

El desarrollo del segundo Sprint se llevó a cabo desde el 18/08/2023 al 08/09/2023. Con el objetivo de cumplir con la segunda evaluación de la materia de Taller 3.

Comenzamos planificando el Sprint con la estimación de las historias de usuario que conforman el mismo. Seguidamente se detallan las tareas a realizar para el despliegue de cada historia de usuario, los criterios que se usan para la usabilidad y los criterios de aceptación que serán parte fundamental para cumplir con los objetivos planteados en este sprint. Posteriormente se procede con el desarrollo de las actividades planteadas en la planificación para implementar de forma exitosa las historias de usuario.

II.1.2.2.2.2 Estimación del Sprint Backlog

En la tabla a continuación se visualiza las historias de usuario que conforman el Sprint 2 y que fueron definidas dentro del Sprint Planning. Se establecen las prioridades, estimación y horas reales que serán utilizadas en el desarrollo de cada historia de usuario.

Tabla 30: Tabla de estimación del Sprint Backlog del Sprint 2

Código	Historia	Prioridad	Estimación	Horas reales
HU03	Registrar usuario	Alto	120	125
HU04	Leer usuarios registrados	Medio	72	74
HU05	Actualizar datos de usuarios	Alto	120	120

HU06	Dar de baja usuario	Bajo	48	52
------	---------------------	------	----	----

Fuente: Elaboración propia

Detalle de cada historia de usuario:

HU03 – Registrar usuario	
Descripción: La historia de usuario describe la forma en el que usuario encargado pueda registrar a un usuario nuevo que forma parte de la clínica.	
Prioridad: Alto	Estimación: 120
Criterios de aceptación:	
<ul style="list-style-type: none"> ✓ DADO QUE soy un usuario encargado y quiero registrar a un usuario nuevo CUANDO ingreso a la interfaz de usuarios, presiono en el botón agregar usuario, lleno el formulario presentado con los datos obligatorios ENTONCES se registra en el sistema al usuario nuevo y se muestra automáticamente en el listado general. ✓ DADO QUE soy un usuario encargado y quiero cancelar el proceso de registro de un usuario nuevo. CUANDO ingreso a la interfaz de registro y presiono en la esquina superior derecha la X para cerrar el formulario ENTONCES el sistema me cancela el proceso de registro y me redirige a la interfaz del listado general de usuarios. 	
Tareas:	
<ul style="list-style-type: none"> ✓ Realizar el análisis, entendimiento y diseño de la historia de usuario. ✓ Realizar el modelo de datos. ✓ Diseñar los prototipos correspondientes. ✓ Realizar el desarrollo de la historia de usuario. ✓ Diseñar casos de prueba. ✓ Ejecutar los casos de prueba planteados. ✓ Solucionar los defectos encontrados. ✓ Desplegar la historia de usuario. 	
Definiciones adicionales:	
<ul style="list-style-type: none"> ✓ Los campos del formulario de registro son: Nombre (obligatorio), Apellido Paterno (opcional), Apellido Materno (opcional), Celular (obligatorio), Rol (obligatorio), Pseudónimo (obligatorio), Clave (obligatorio). ✓ Al saltarse un campo obligatorio sin llenar el sistema mostrará una alerta indicando que se deben completar. ✓ Al registrar un usuario nuevo, el sistema muestra un toastify con registro exitoso y redirige automáticamente 	

a la interfaz principal con el registro nuevo dentro del listado general.

Fuente: Elaboración propia

HU04 – Leer usuarios registrados	
Descripción: La historia de usuario describe la forma en la que el usuario encargado visualiza a los usuarios registrados en el sistema.	
Prioridad: Medio	Estimación: 72
Criterios de aceptación:	
<ul style="list-style-type: none"> ✓ DADO QUE soy un usuario encargado y quiero visualizar todos los usuarios registrados en el sistema CUANDO ingreso a la interfaz de usuarios ENTONCES el sistema me muestra el listado general con los usuarios registrados en la base de datos. ✓ DADO QUE deseo filtrar registros CUANDO digito dentro de un pequeño buscador en la interfaz principal de usuarios. ENTONCES el sistema filtrará mediante cada columna en tiempo real. 	
Tareas:	
<ul style="list-style-type: none"> ✓ Realizar el análisis, entendimiento y diseño de la historia de usuario. ✓ Realizar el modelo de datos. ✓ Diseñar el prototipo de acceso al sistema. ✓ Realizar el desarrollo de la historia de usuario. ✓ Diseñar casos de prueba. ✓ Ejecutar los casos de prueba planteados. ✓ Solucionar los defectos encontrados. ✓ Desplegar la historia de usuario 	

Fuente: Elaboración propia

HU05 – Actualizar datos de usuarios	
Descripción: La historia de usuario describe la forma en que un usuario actualiza los datos de un registro.	
Prioridad: Alto	Estimación: 120
<p>Criterios de aceptación:</p> <ul style="list-style-type: none"> ✓ DADO QUE deseo actualizar los datos de un registro CUANDO selecciono un registro dentro del listado general Y corrijo los datos correspondientes en el formulario ENTONCES el sistema me actualiza el registro con los datos que corregí. ✓ DADO QUE deseo cancelar la actualización de un registro CUANDO seleccione la X de la parte superior derecha del formulario ENTONCES el sistema cancela la desactivación del registro. 	
<p>Tareas:</p> <ul style="list-style-type: none"> ✓ Realizar el análisis, entendimiento y diseño de la historia de usuario. ✓ Realizar el modelo de datos. ✓ Diseñar los prototipos correspondientes. ✓ Realizar el desarrollo de la historia de usuario. ✓ Diseñar casos de prueba. ✓ Ejecutar los casos de prueba planteados. ✓ Solucionar los defectos encontrados. ✓ Desplegar la historia de usuario. 	
<p>Definiciones adicionales:</p> <ul style="list-style-type: none"> ✓ Se creará un botón en cada fila del listado general para poder actualizar el registro que me redirigirá al formulario correspondiente. ✓ Los campos del formulario son: Nombre (obligatorio), Apellido Paterno (opcional), Apellido Materno (opcional), Celular (obligatorio), Rol (obligatorio), Pseudónimo (obligatorio), Clave (obligatorio). ✓ Al saltarse un cambio obligatorio sin llenar el sistema mostrará una alerta indicando que se deben completar. ✓ Al guardar un usuario con los campos a actualizar, el sistema muestra un toastify con registro exitoso y redirige automáticamente a la interfaz principal con el registro actualizado dentro del listado general. 	

Fuente: Elaboración propia

HU06 – Dar de baja usuario	
Descripción: La historia de usuario describe la forma en que un usuario desactiva el registro de un usuario seleccionado.	
Prioridad: Bajo	Estimación: 48
Criterios de aceptación:	
<ul style="list-style-type: none"> ✓ DADO QUE deseo desactivar un registro CUANDO seleccione el registro dentro del listado general ENTONCES el sistema desactiva el registro. ✓ DADO QUE deseo cancelar la desactivación del registro CUANDO seleccione la opción CANCELAR de la parte inferior de la ventana de confirmación ✓ ENTONCES el sistema cancela la desactivación del registro. 	
Tareas:	
<ul style="list-style-type: none"> ✓ Realizar el análisis, entendimiento y diseño de la historia de usuario. ✓ Realizar el modelo de datos. ✓ Diseñar los prototipos correspondientes. ✓ Realizar el desarrollo de la historia de usuario. ✓ Diseñar casos de prueba. ✓ Ejecutar los casos de prueba planteados. ✓ Solucionar los defectos encontrados. ✓ Desplegar la historia de usuario. 	
Definiciones adicionales:	
<ul style="list-style-type: none"> ✓ Se creará un botón en cada fila del listado general para poder dar de baja el registro. ✓ Al momento de seleccionar el registro a desactivar, aparecerá un pop-up para confirmar la desactivación del registro. ✓ Al confirmar la desactivación el sistema muestra un toastify con mensaje de desactivado exitoso y redirige automáticamente a la interfaz principal con el registro eliminado del listado general. 	

Fuente: Elaboración propia

II.1.2.2.3 Análisis y Diseño

A continuación, se presenta los diagramas de actividad de las historias descritas en el Backlog para el presente Sprint. No se detallará a profundidad los diagramas por ser una metodología ágil, lo mencionado anteriormente se encuentra dentro de la descripción de cada historia de usuario.

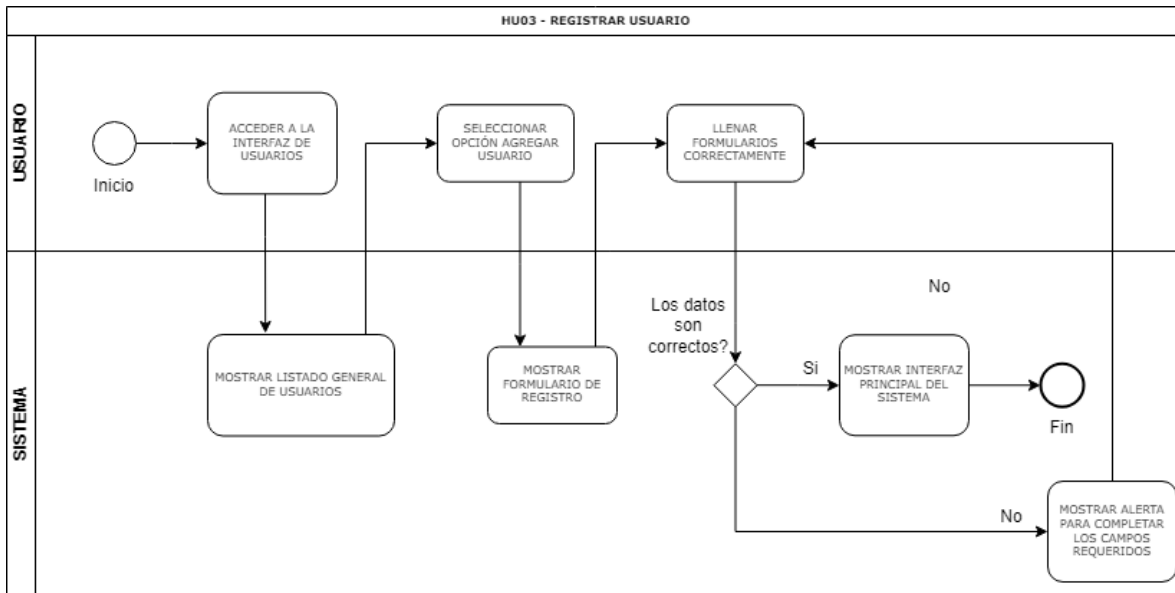


Figura 58: Diagrama de actividad HU03 – Registrar usuario

Fuente: Elaboración propia

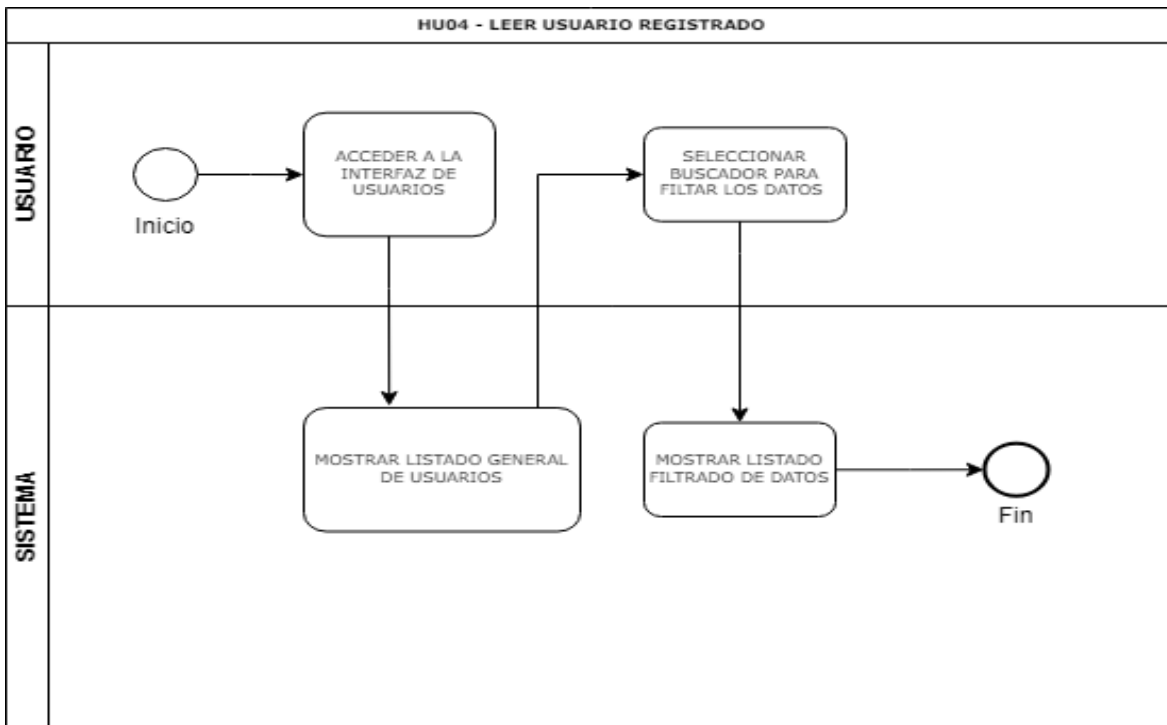


Figura 59: Diagrama de actividad HU04 – Leer usuario registrado

Fuente: Elaboración propia

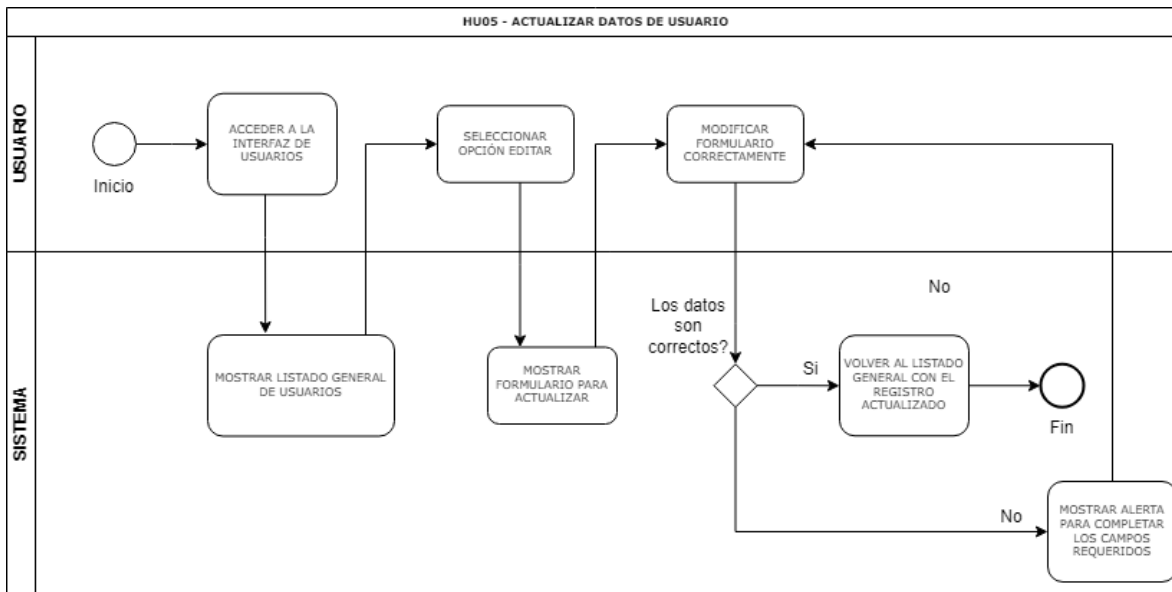


Figura 60: Diagrama de actividad HU05 – Actualizar datos de usuario

Fuente: Elaboración propia

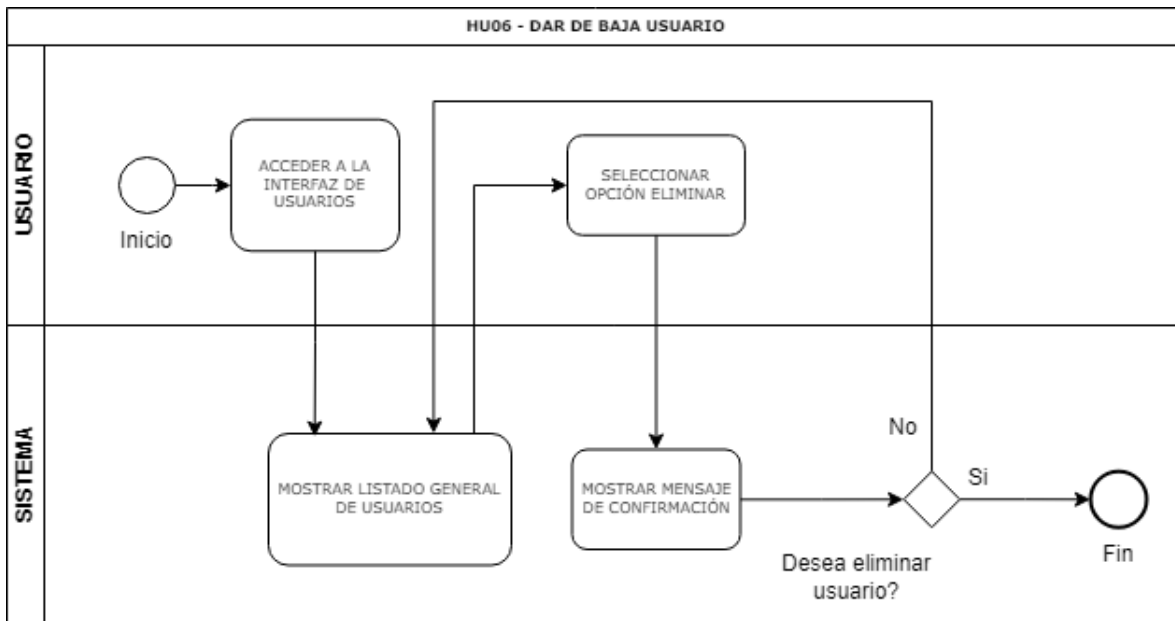


Figura 61: Diagrama de actividad HU06 – Dar de baja usuario

Fuente: Elaboración propia

A continuación, se presenta los diagramas de secuencia de las historias descritas en el Backlog para el presente Sprint. No se detallará a profundidad los diagramas por ser una metodología ágil, lo mencionado anteriormente se encuentra dentro de la descripción de cada historia de usuario.

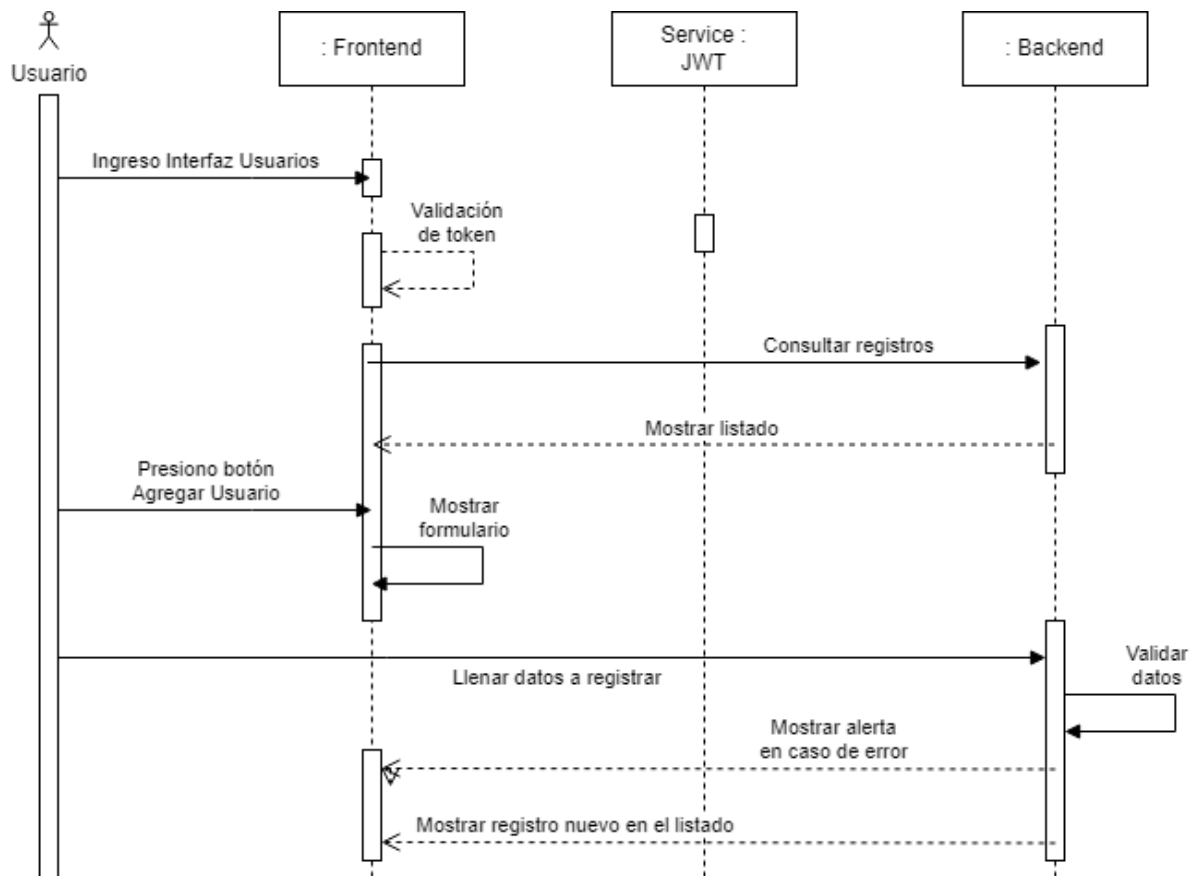


Figura 62: Diagrama de secuencia HU03 – Registrar usuario

Fuente: Elaboración propia

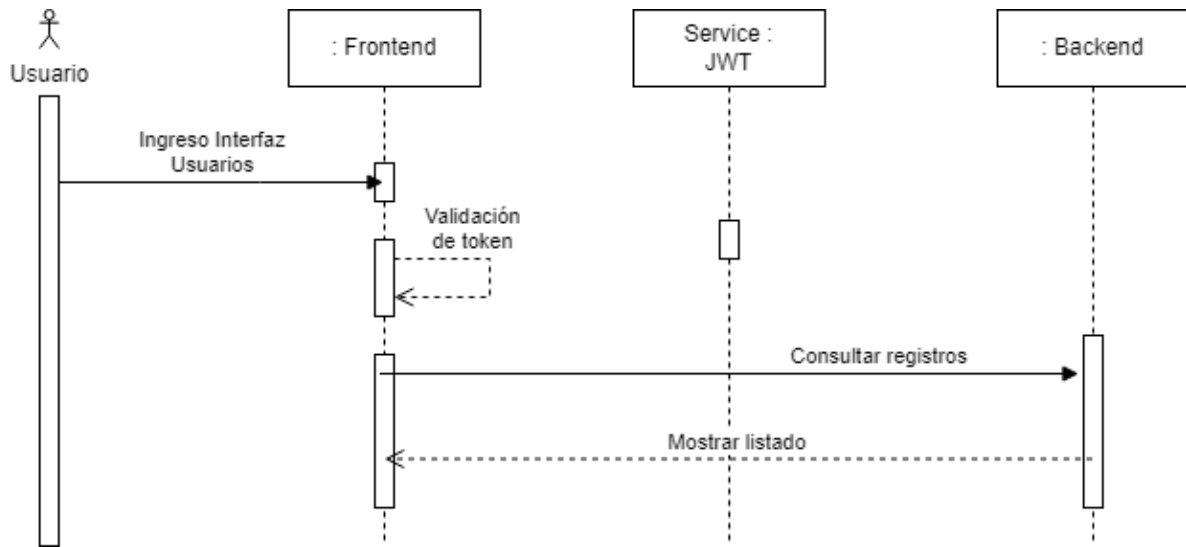


Figura 63: Diagrama de secuencia HU04 – Leer usuario registrado

Fuente: Elaboración propia

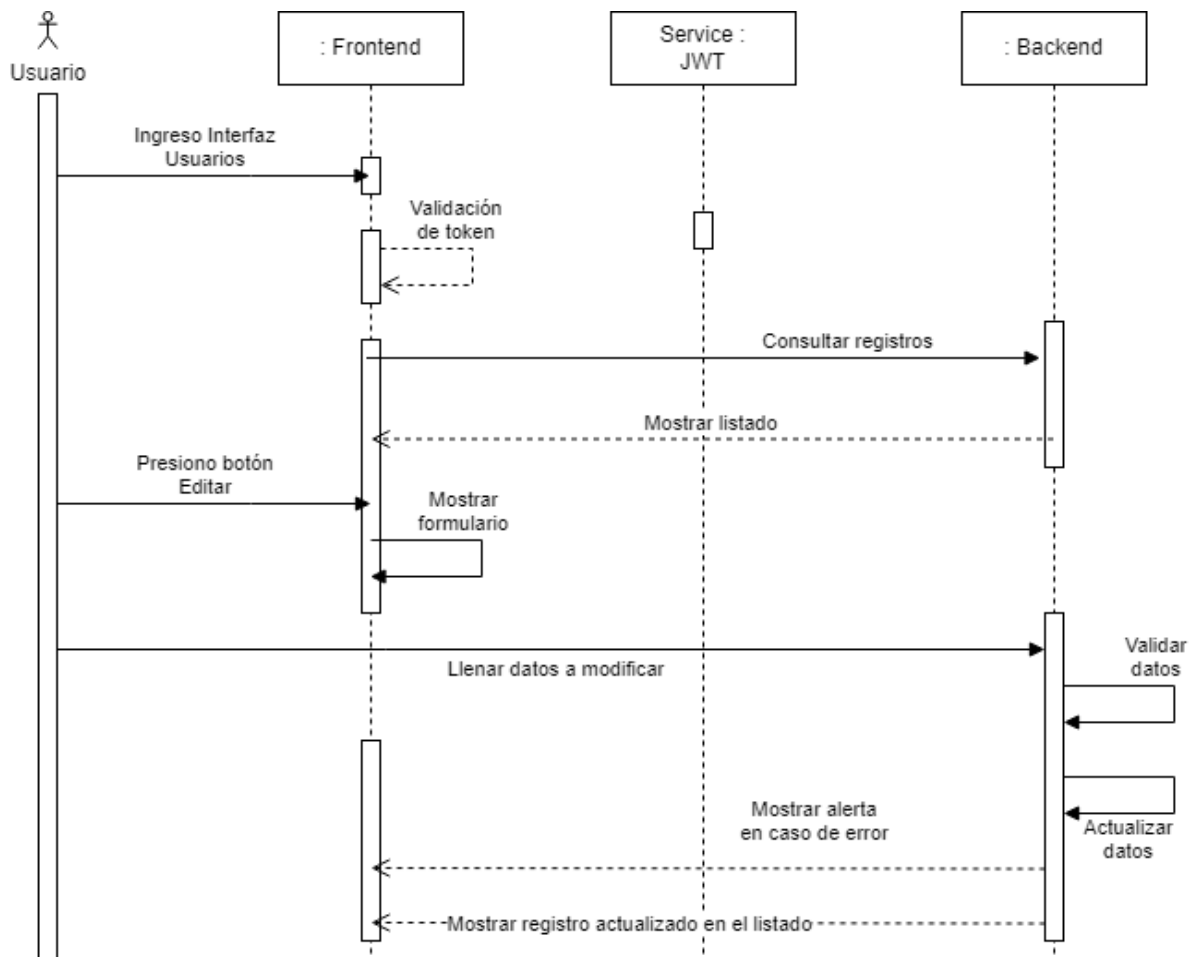


Figura 64: Diagrama de secuencia HU05 – Actualizar datos de usuario

Fuente: Elaboración propia

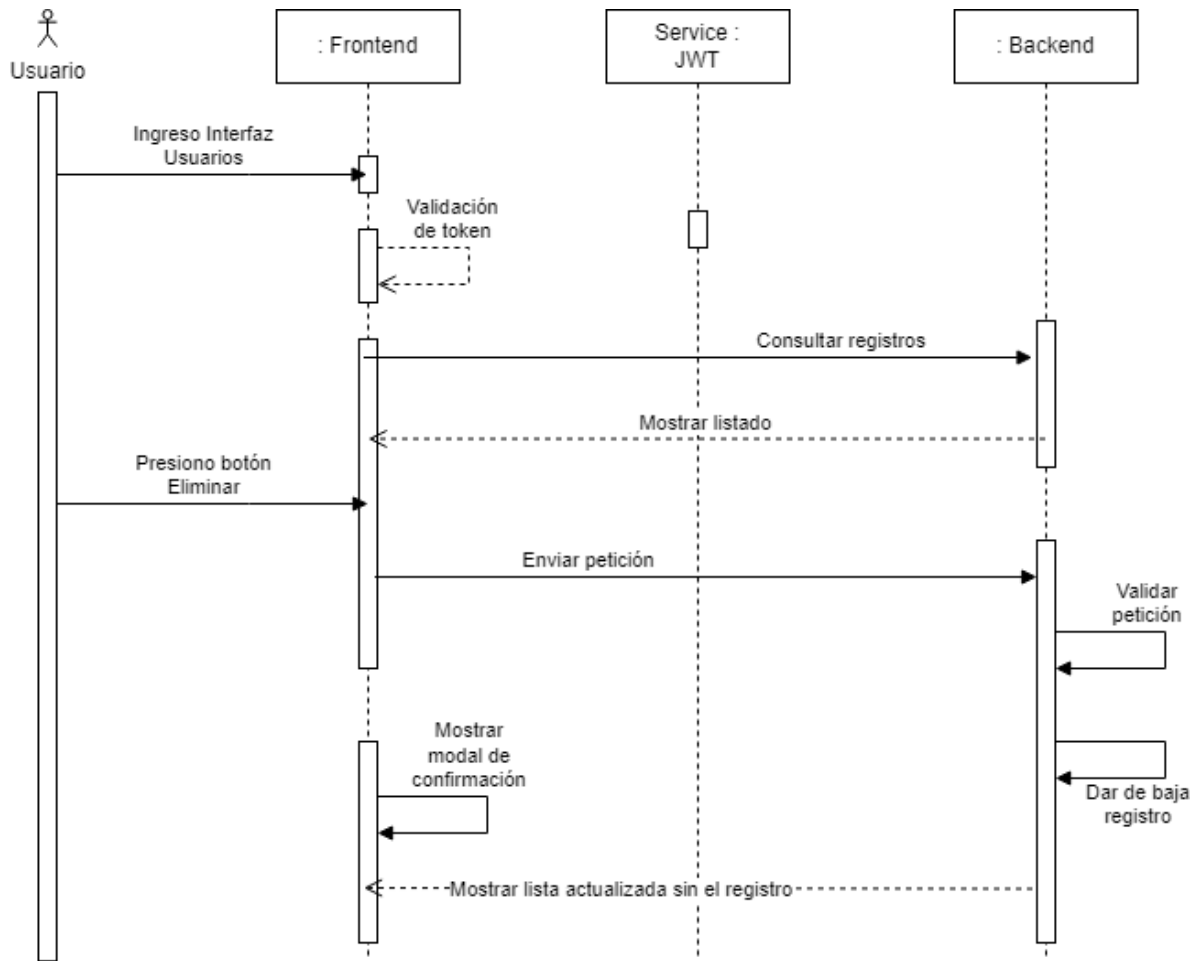


Figura 65: Diagrama de secuencia HU06 – Dar de baja usuario

Fuente: Elaboración propia

II.1.2.2.4 Interfaces

A continuación, se muestran las interfaces de las historias de usuario para este Sprint 2:

HU03 – Registrar Usuario

En la siguiente imagen se muestra el formulario de registro de usuario:

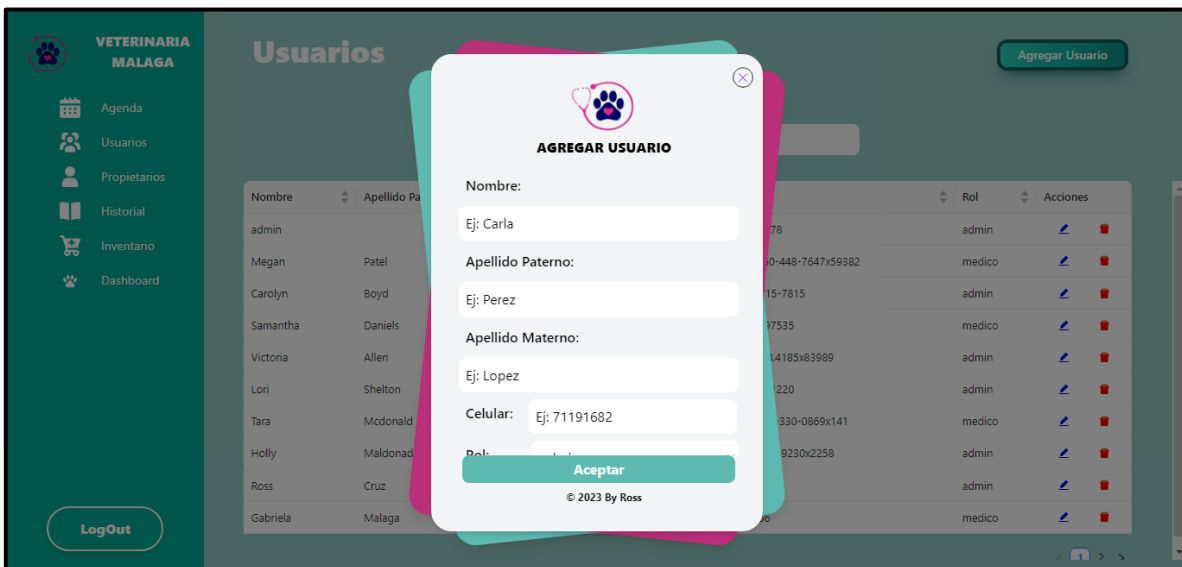


Figura 66: Interfaz HU03 – Registrar Usuario

Fuente: Elaboración propia

HU04 – Leer Usuarios Registrados

En la siguiente imagen se muestra la interfaz principal de usuarios, un listado de todos los usuarios registrados:

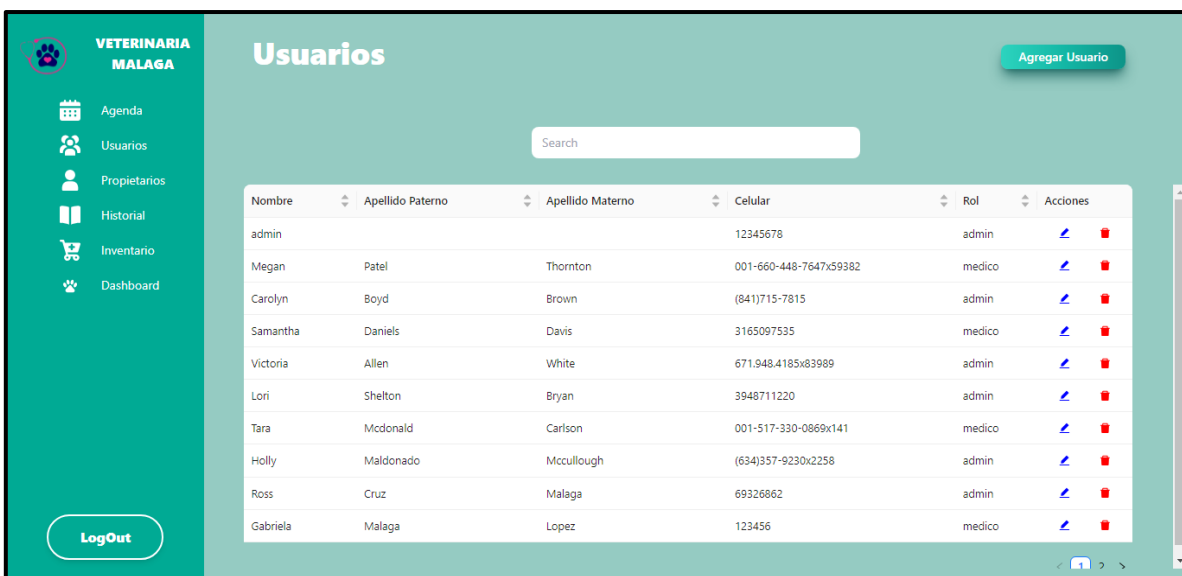


Figura 67: Interfaz HU04 – Leer Usuarios Registrados

Fuente: Elaboración propia

HU05 – Actualizar datos de Usuarios

En la siguiente imagen se muestra el formulario para editar un usuario:

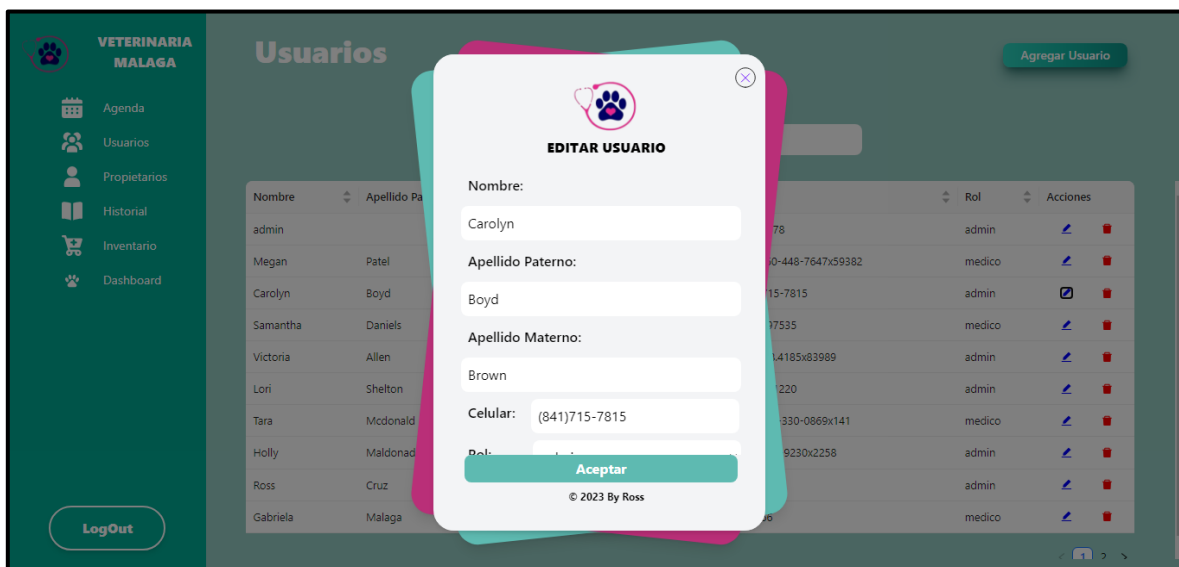


Figura 68: Interfaz HU05 – Actualizar datos de Usuarios

Fuente: Elaboración propia

HU06 – Dar de Baja Usuario

En la siguiente imagen se muestra la validación para eliminar un usuario:

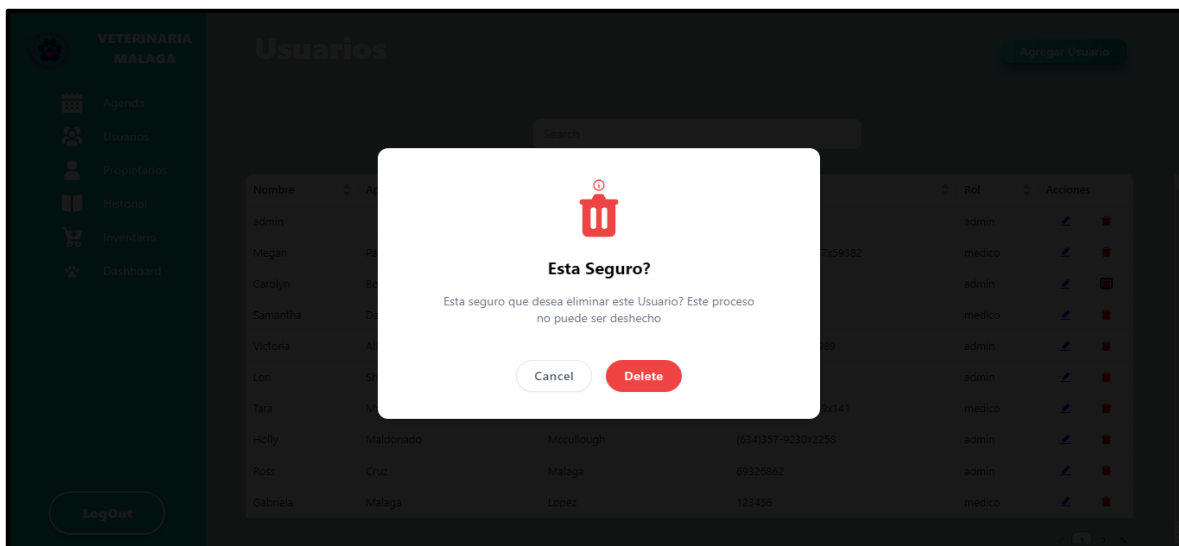


Figura 69: Interfaz HU06 – Dar de Baja Usuario

Fuente: Elaboración propia

II.1.2.2.2.5 Gráfico Burn Down

En la siguiente gráfica se puede observar que en los primeros días del sprint las estimaciones iban un poco desniveladas. Durante la etapa intermedia y final del sprint se observa una nivelación acorde al tiempo estimado.

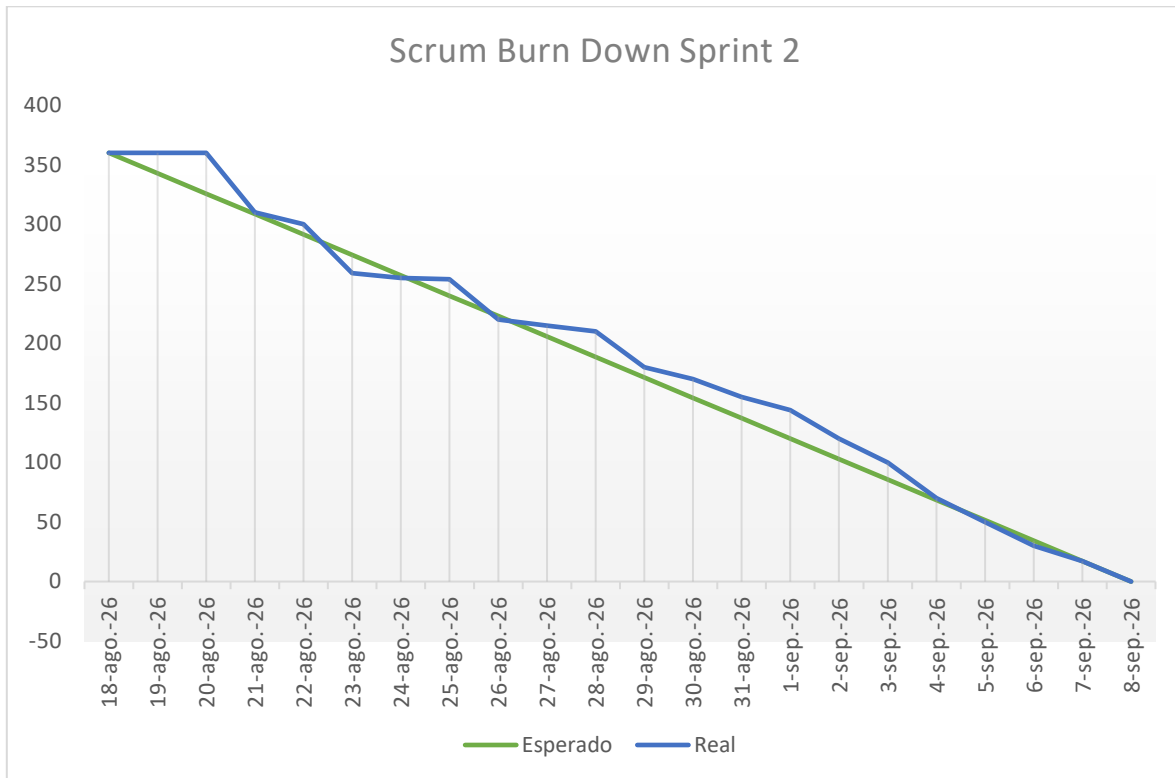


Figura 70: Gráfico Burn Down Sprint 2

Fuente: Elaboración propia

II.1.2.2.2.6 Revisión del Sprint (Sprint Review)

En la siguiente tabla se muestra el resultado que se obtuvo durante este segundo sprint, de un total de cuatro historias implementadas, se aprobaron todas.

Tabla 31: Sprint Review

Código	Historia de usuario	Aprobada	Rechazada	Detalles
HU03	Registrar usuario	X		(OK)
HU04	Leer usuarios registrados	X		(OK)
HU05	Actualizar datos de usuarios	X		(OK)
HU06	Dar de baja usuario	X		(OK)

Fuente: Elaboración propia

II.1.2.2.3 Sprint 3: Administración y Reportes de Usuarios, Mascotas, Propietarios, Citas y Atenciones

II.1.2.2.3.1 Planificación del Sprint

El desarrollo del tercer Sprint se llevó a cabo desde el 09/09/2023 al 06/11/2023. Con el objetivo de cumplir con la tercera evaluación de la materia de Taller 3 y entregar al cliente las historias de usuario que permitirán manejar la administración de los datos de los clientes de la Clínica Veterinaria Málaga. (13 semanas)

Comenzamos planificando el Sprint con la estimación de las historias de usuario que conforman el mismo. Seguidamente se detallan las tareas a realizar para el despliegue de cada historia de usuario, los criterios que se usan para la usabilidad y los criterios de aceptación que serán parte fundamental para cumplir con los objetivos planteados en este sprint. Posteriormente se procede con el desarrollo de las actividades planteadas en la planificación para implementar de forma exitosa las historias de usuario.

II.1.2.2.3.2 Estimación del Sprint Backlog

En la tabla a continuación se visualiza las historias de usuario que conforman el Sprint 3 y que fueron definidas dentro del Sprint Planning. Se establecen las prioridades, estimación y horas reales que serán utilizadas en el desarrollo de cada historia de usuario.

Tabla 32: Tabla de estimación del Sprint Backlog del Sprint 3

Código	Historia	Prioridad	Estimación	Horas reales
HU07	Registrar mascota	Alto	120	140
HU08	Leer mascotas registradas	Medio	72	80
HU09	Actualizar datos de mascotas	Alto	120	120
HU10	Dar de baja mascota	Medio	72	78
HU11	Registrar cita	Alto	120	130
HU12	Leer citas registradas	Medio	72	70
HU13	Cancelar/Eliminar cita	Bajo	48	55
HU14	Registrar propietario	Alto	120	115
HU15	Leer propietarios registrados	Medio	72	72
HU16	Actualizar datos de propietarios	Alto	120	120
HU17	Dar de baja propietario	Medio	72	70
HU18	Leer historiales clínicos	Alto	120	119
HU19	Reporte historia clínica	Bajo	48	50
HU20	Registrar atención clínica	Alto	120	150
HU21	Generar receta	Bajo	48	50
HU22	Generar recibo	Bajo	48	64
HU23	Registrar medicamento	Alto	120	115
HU24	Leer medicamentos registrados	Medio	72	72
HU25	Actualizar datos de medicamentos	Alto	120	120
HU26	Eliminar medicamento	Medio	72	70
HU27	Dashboard	Medio	72	80

Fuente: Elaboración propia

Detalle de cada historia de usuario:

HU07 – Registrar mascota	
Descripción: La historia de usuario describe la forma en el que usuario encargado pueda registrar a una nueva mascota que formará parte de la clínica.	
Prioridad: Alto	Estimación: 120
<p>Criterios de aceptación:</p> <ul style="list-style-type: none"> ✓ DADO QUE soy un usuario encargado y quiero registrar a una mascota nueva CUANDO ingreso a la interfaz de mascotas, presiono en el botón agregar mascota, lleno el formulario presentado con los datos obligatorios ENTONCES se registra en el sistema a la mascota nueva y se muestra automáticamente en el listado general de mascotas del propietario correspondiente. ✓ DADO QUE soy un usuario encargado y quiero cancelar el proceso de registro de una mascota nueva. CUANDO ingreso a la interfaz de registro y presiono en la esquina superior derecha la X para cerrar el formulario ENTONCES el sistema me cancela el proceso de registro y me redirige a la interfaz del listado general de mascotas. 	
<p>Tareas:</p> <ul style="list-style-type: none"> ✓ Realizar el análisis, entendimiento y diseño de la historia de usuario. ✓ Realizar el modelo de datos. ✓ Diseñar los prototipos correspondientes. ✓ Realizar el desarrollo de la historia de usuario. ✓ Diseñar casos de prueba. ✓ Ejecutar los casos de prueba planteados. ✓ Solucionar los defectos encontrados. ✓ Desplegar la historia de usuario. 	
<p>Definiciones adicionales:</p> <ul style="list-style-type: none"> ✓ Los campos del formulario de registro son: Nombre (obligatorio), Sexo (obligatorio), Fecha de Nacimiento (obligatorio), Raza (obligatorio), Color (obligatorio), Chip (opcional), Alergias (obligatorio), Esterilizado (obligatorio), Foto (opcional), Especie (obligatorio), Tamaño (obligatorio). ✓ Al saltarse un campo obligatorio sin llenar el sistema mostrará una alerta indicando que se deben completar. ✓ Al registrar una mascota nueva, el sistema muestra un toastify con registro exitoso y redirige automáticamente 	

a la interfaz principal con el registro nuevo dentro del listado general.

Fuente: Elaboración propia

HU08 – Leer mascotas registradas

Descripción: La historia de usuario describe la forma en la que el usuario encargado visualiza a las mascotas registradas en el sistema, de acuerdo al propietario.

Prioridad: Medio

Estimación: 72

Criterios de aceptación:

- ✓ DADO QUE soy un usuario encargado y quiero visualizar todas las mascotas registradas en el sistema dependiendo su propietario
- CUANDO ingreso a la interfaz de mascotas
- ENTONCES el sistema me muestra el listado general con las mascotas registradas en la base de datos según su respectivo propietario.

Tareas:

- ✓ Realizar el análisis, entendimiento y diseño de la historia de usuario.
- ✓ Realizar el modelo de datos.
- ✓ Diseñar el prototipo de acceso al sistema.
- ✓ Realizar el desarrollo de la historia de usuario.
- ✓ Diseñar casos de prueba.
- ✓ Ejecutar los casos de prueba planteados.
- ✓ Solucionar los defectos encontrados.
- ✓ Desplegar la historia de usuario

Definiciones adicionales:

- ✓ Los datos de cada propietario se encuentran al pie de la página.

Fuente: Elaboración propia

HU09 – Actualizar datos de mascotas	
Descripción: La historia de usuario describe la forma en que un usuario actualiza los datos de un registro.	
Prioridad: Alto	Estimación: 120
<p>Criterios de aceptación:</p> <ul style="list-style-type: none"> ✓ DADO QUE deseo actualizar los datos de un registro CUANDO selecciono un registro dentro del listado general Y corrijo los datos correspondientes en el formulario ENTONCES el sistema me actualiza el registro con los datos que corregí. ✓ DADO QUE deseo cancelar la actualización de un registro CUANDO seleccione la X de la parte superior derecha del formulario ENTONCES el sistema cancela la desactivación del registro. 	
<p>Tareas:</p> <ul style="list-style-type: none"> ✓ Realizar el análisis, entendimiento y diseño de la historia de usuario. ✓ Realizar el modelo de datos. ✓ Diseñar los prototipos correspondientes. ✓ Realizar el desarrollo de la historia de usuario. ✓ Diseñar casos de prueba. ✓ Ejecutar los casos de prueba planteados. ✓ Solucionar los defectos encontrados. ✓ Desplegar la historia de usuario. 	
<p>Definiciones adicionales:</p> <ul style="list-style-type: none"> ✓ Se creará un botón en cada tarjeta del listado general para poder actualizar el registro que me redirigirá al formulario correspondiente. ✓ Los campos del formulario de registro son: Nombre (obligatorio), Sexo (obligatorio), Fecha de Nacimiento (obligatorio), Raza (obligatorio), Color (obligatorio), Chip (opcional), Alergias (obligatorio), Esterilizado (obligatorio), Foto (opcional), Especie (obligatorio), Tamaño (obligatorio). ✓ Al saltarse un cambio obligatorio sin llenar el sistema mostrará una alerta indicando que se deben completar. ✓ Al guardar un usuario con los campos a actualizar, el sistema muestra un toastify con registro exitoso y redirige automáticamente a la interfaz principal con el registro actualizado dentro del listado general. 	

Fuente: Elaboración propia

HU10 – Dar de baja mascota	
Descripción: La historia de usuario describe la forma en que un usuario desactiva el registro de una mascota seleccionada.	
Prioridad: Medio	Estimación: 72
Criterios de aceptación:	
<ul style="list-style-type: none"> ✓ DADO QUE deseo desactivar un registro CUANDO seleccione el registro dentro del listado general ENTONCES el sistema desactiva el registro. ✓ DADO QUE deseo cancelar la desactivación del registro CUANDO seleccione la opción CANCELAR de la parte inferior de la ventana de confirmación ✓ ENTONCES el sistema cancela la desactivación del registro. 	
Tareas:	
<ul style="list-style-type: none"> ✓ Realizar el análisis, entendimiento y diseño de la historia de usuario. ✓ Realizar el modelo de datos. ✓ Diseñar los prototipos correspondientes. ✓ Realizar el desarrollo de la historia de usuario. ✓ Diseñar casos de prueba. ✓ Ejecutar los casos de prueba planteados. ✓ Solucionar los defectos encontrados. ✓ Desplegar la historia de usuario. 	
Definiciones adicionales:	
<ul style="list-style-type: none"> ✓ Se creará un botón en cada tarjeta del listado general para poder dar de baja el registro. ✓ Al momento de seleccionar el registro a desactivar, aparecerá un pop-up para confirmar la desactivación del registro. ✓ Al confirmar la desactivación el sistema muestra un toastify con mensaje de desactivado exitoso y redirige automáticamente a la interfaz principal con el registro eliminado del listado general. 	

Fuente: Elaboración propia

HU11 – Registrar cita	
Descripción: La historia de usuario describe la forma en el que usuario encargado pueda registrar a una cita nueva en la agenda mensual.	
Prioridad: Alto	Estimación: 120
<p>Criterios de aceptación:</p> <ul style="list-style-type: none"> ✓ DADO QUE soy un usuario encargado y quiero registrar a una cita nueva CUANDO ingreso a la interfaz de agenda, presiono en la fecha que deseo registrar la cita, lleno el formulario presentado con los datos obligatorios ENTONCES se registra en el sistema la cita nueva y se muestra automáticamente en el calendario de citas. ✓ DADO QUE soy un usuario encargado y quiero cancelar el proceso de registro de una cita nueva. CUANDO ingreso a la interfaz de registro y presiono en la esquina superior derecha la X para cerrar el formulario ENTONCES el sistema me cancela el proceso de registro y me redirige a la interfaz del calendario de agenda. 	
<p>Tareas:</p> <ul style="list-style-type: none"> ✓ Realizar el análisis, entendimiento y diseño de la historia de usuario. ✓ Realizar el modelo de datos. ✓ Diseñar los prototipos correspondientes. ✓ Realizar el desarrollo de la historia de usuario. ✓ Diseñar casos de prueba. ✓ Ejecutar los casos de prueba planteados. ✓ Solucionar los defectos encontrados. ✓ Desplegar la historia de usuario. 	
<p>Definiciones adicionales:</p> <ul style="list-style-type: none"> ✓ Los campos del formulario de registro son: Descripción (obligatorio), Fecha Inicial (obligatorio), Médico (obligatorio), Propietario (obligatorio). ✓ Al saltarse un cambio obligatorio sin llenar el sistema mostrará una alerta indicando que se deben completar. ✓ Al agendar una cita nueva, el sistema muestra un toastify con registro exitoso y redirige automáticamente a la interfaz principal con el registro nuevo dentro del listado general en el calendario. 	

Fuente: Elaboración propia

HU12 – Leer citas registradas	
Descripción: La historia de usuario describe la forma en la que el usuario encargado visualiza en el calendario las citas agendadas.	
Prioridad: Medio	Estimación: 72
Criterios de aceptación: <ul style="list-style-type: none"> ✓ DADO QUE soy un usuario encargado y quiero visualizar todas las citas agendadas en el mes seleccionado CUANDO ingreso a la interfaz de agenda ENTONCES el sistema me muestra el listado general en el calendario con las citas agendadas. 	
Tareas: <ul style="list-style-type: none"> ✓ Realizar el análisis, entendimiento y diseño de la historia de usuario. ✓ Realizar el modelo de datos. ✓ Diseñar el prototipo de acceso al sistema. ✓ Realizar el desarrollo de la historia de usuario. ✓ Diseñar casos de prueba. ✓ Ejecutar los casos de prueba planteados. ✓ Solucionar los defectos encontrados. ✓ Desplegar la historia de usuario 	
Definiciones adicionales: <ul style="list-style-type: none"> ✓ La agenda puede mostrar las citas agendadas según mes, semana o día, de acuerdo a elección del usuario. 	

Fuente: Elaboración propia

HU13 – Cancelar cita	
Descripción: La historia de usuario describe la forma en que un usuario desactiva el registro de una cita agendada.	
Prioridad: Bajo	Estimación: 48
Criterios de aceptación:	
<ul style="list-style-type: none"> ✓ DADO QUE deseo desactivar un registro CUANDO seleccione el registro dentro del calendario de citas ENTONCES el sistema desactiva el registro. ✓ DADO QUE deseo cancelar la desactivación del registro CUANDO seleccione la opción CANCELAR de la parte inferior de la ventana de confirmación ✓ ENTONCES el sistema cancela la desactivación del registro. 	
Tareas:	
<ul style="list-style-type: none"> ✓ Realizar el análisis, entendimiento y diseño de la historia de usuario. ✓ Realizar el modelo de datos. ✓ Diseñar los prototipos correspondientes. ✓ Realizar el desarrollo de la historia de usuario. ✓ Diseñar casos de prueba. ✓ Ejecutar los casos de prueba planteados. ✓ Solucionar los defectos encontrados. ✓ Desplegar la historia de usuario. 	
Definiciones adicionales:	
<ul style="list-style-type: none"> ✓ Se creará un botón en cada recuadro de fecha del calendario para poder dar de baja el registro. ✓ Al momento de seleccionar el registro a desactivar, aparecerá un pop-up para confirmar la desactivación del registro. ✓ Al confirmar la desactivación el sistema muestra un toastify con mensaje de desactivado exitoso y redirige automáticamente a la interfaz principal con el registro eliminado del calendario. 	

Fuente: Elaboración propia

HU14 – Registrar propietario	
Descripción: La historia de usuario describe la forma en el que usuario encargado pueda registrar a un propietario nuevo.	
Prioridad: Alto	Estimación: 120
<p>Criterios de aceptación:</p> <ul style="list-style-type: none"> ✓ DADO QUE soy un usuario encargado y quiero registrar a un propietario nuevo CUANDO ingreso a la interfaz de propietarios, presiono en el botón agregar propietario, lleno el formulario presentado con los datos obligatorios ENTONCES se registra en el sistema al propietario nuevo y se muestra automáticamente en el listado general. ✓ DADO QUE soy un usuario encargado y quiero cancelar el proceso de registro de un propietario nuevo CUANDO ingreso a la interfaz de registro y presiono en la esquina superior derecha la X para cerrar el formulario ENTONCES el sistema me cancela el proceso de registro y me redirige a la interfaz del listado general de propietarios. 	
<p>Tareas:</p> <ul style="list-style-type: none"> ✓ Realizar el análisis, entendimiento y diseño de la historia de usuario. ✓ Realizar el modelo de datos. ✓ Diseñar los prototipos correspondientes. ✓ Realizar el desarrollo de la historia de usuario. ✓ Diseñar casos de prueba. ✓ Ejecutar los casos de prueba planteados. ✓ Solucionar los defectos encontrados. ✓ Desplegar la historia de usuario. 	
<p>Definiciones adicionales:</p> <ul style="list-style-type: none"> ✓ Los campos del formulario de registro son: Nombre (obligatorio), Apellido Paterno (opcional), Apellido Materno (opcional), Celular (obligatorio), Dirección (obligatorio). ✓ Al saltarse un cambio obligatorio sin llenar el sistema mostrará una alerta indicando que se deben completar. ✓ Al registrar un propietario nuevo, el sistema muestra un toastify con registro exitoso y redirige automáticamente a la interfaz principal con el registro nuevo dentro del listado general. 	

Fuente: Elaboración propia

HU15 – Leer propietarios registrados	
Descripción: La historia de usuario describe la forma en la que el usuario encargado visualiza a los propietarios registrados en el sistema.	
Prioridad: Medio	Estimación: 72
Criterios de aceptación:	
<ul style="list-style-type: none"> ✓ DADO QUE soy un usuario encargado y quiero visualizar todos los propietarios registrados en el sistema CUANDO ingreso a la interfaz de propietarios ENTONCES el sistema me muestra el listado general con los propietarios registrados en la base de datos. ✓ DADO QUE deseo filtrar registros CUANDO digito dentro de un pequeño buscador en la interfaz principal de propietarios ENTONCES el sistema filtrará mediante cada columna en tiempo real. 	
Tareas:	
<ul style="list-style-type: none"> ✓ Realizar el análisis, entendimiento y diseño de la historia de usuario. ✓ Realizar el modelo de datos. ✓ Diseñar el prototipo de acceso al sistema. ✓ Realizar el desarrollo de la historia de usuario. ✓ Diseñar casos de prueba. ✓ Ejecutar los casos de prueba planteados. ✓ Solucionar los defectos encontrados. ✓ Desplegar la historia de usuario 	

Fuente: Elaboración propia

HU16 – Actualizar datos de propietarios	
Descripción: La historia de usuario describe la forma en que un usuario actualiza los datos de un registro.	
Prioridad: Alto	Estimación: 120
<p>Criterios de aceptación:</p> <ul style="list-style-type: none"> ✓ DADO QUE deseo actualizar los datos de un registro CUANDO selecciono un registro dentro del listado general Y corrijo los datos correspondientes en el formulario ENTONCES el sistema me actualiza el registro con los datos que corregí. ✓ DADO QUE deseo cancelar la actualización de un registro CUANDO seleccione la X de la parte superior derecha del formulario ENTONCES el sistema cancela la desactivación del registro. 	
<p>Tareas:</p> <ul style="list-style-type: none"> ✓ Realizar el análisis, entendimiento y diseño de la historia de usuario. ✓ Realizar el modelo de datos. ✓ Diseñar los prototipos correspondientes. ✓ Realizar el desarrollo de la historia de usuario. ✓ Diseñar casos de prueba. ✓ Ejecutar los casos de prueba planteados. ✓ Solucionar los defectos encontrados. ✓ Desplegar la historia de usuario. 	
<p>Definiciones adicionales:</p> <ul style="list-style-type: none"> ✓ Se creará un botón en cada fila del listado general para poder actualizar el registro que me redirigirá al formulario correspondiente. ✓ Los campos del formulario de registro son: Nombre (obligatorio), Apellido Paterno (opcional), Apellido Materno (opcional), Celular (obligatorio), Dirección (obligatorio). ✓ Al saltarse un cambio obligatorio sin llenar el sistema mostrará una alerta indicando que se deben completar. ✓ Al guardar un propietario con los campos a actualizar, el sistema muestra un toastify con registro exitoso y redirige automáticamente a la interfaz principal con el registro actualizado dentro del listado general. 	

Fuente: Elaboración propia

HU17 – Dar de baja propietario	
Descripción: La historia de usuario describe la forma en que un usuario desactiva el registro de un propietario seleccionado.	
Prioridad: Medio	Estimación: 72
Criterios de aceptación: <ul style="list-style-type: none"> ✓ DADO QUE deseo desactivar un registro CUANDO seleccione el registro dentro del listado general ENTONCES el sistema desactiva el registro. ✓ DADO QUE deseo cancelar la desactivación del registro CUANDO seleccione la opción CANCELAR de la parte inferior de la ventana de confirmación ✓ ENTONCES el sistema cancela la desactivación del registro. 	
Tareas: <ul style="list-style-type: none"> ✓ Realizar el análisis, entendimiento y diseño de la historia de usuario. ✓ Realizar el modelo de datos. ✓ Diseñar los prototipos correspondientes. ✓ Realizar el desarrollo de la historia de usuario. ✓ Diseñar casos de prueba. ✓ Ejecutar los casos de prueba planteados. ✓ Solucionar los defectos encontrados. ✓ Desplegar la historia de usuario. 	
Definiciones adicionales: <ul style="list-style-type: none"> ✓ Se creará un botón en cada fila del listado general para poder dar de baja el registro. ✓ Al momento de seleccionar el registro a desactivar, aparecerá un pop-up para confirmar la desactivación del registro. ✓ Al confirmar la desactivación el sistema muestra un toastify con mensaje de desactivado exitoso y redirige automáticamente a la interfaz principal con el registro eliminado del listado general. 	

Fuente: Elaboración propia

HU18 – Leer historiales clínicos	
Descripción: La historia de usuario describe la forma en la que el usuario encargado visualiza los historiales clínicos de las mascotas registradas en el sistema.	
Prioridad: Alto	Estimación: 120
Criterios de aceptación: <ul style="list-style-type: none"> ✓ DADO QUE soy un usuario encargado y quiero visualizar todos los historiales clínicos registrados en el sistema CUANDO ingreso a la interfaz de historiales clínicos ENTONCES el sistema me muestra el listado general de las mascotas registradas en el sistema con sus respectivos historiales. ✓ DADO QUE deseo filtrar historiales clínicos CUANDO digito dentro de un pequeño buscador en la interfaz donde se visualizan los historiales clínicos de cada mascota ENTONCES el sistema filtrará mediante la fecha en tiempo real. 	
Tareas: <ul style="list-style-type: none"> ✓ Realizar el análisis, entendimiento y diseño de la historia de usuario. ✓ Realizar el modelo de datos. ✓ Diseñar el prototipo de acceso al sistema. ✓ Realizar el desarrollo de la historia de usuario. ✓ Diseñar casos de prueba. ✓ Ejecutar los casos de prueba planteados. ✓ Solucionar los defectos encontrados. ✓ Desplegar la historia de usuario 	

Fuente: Elaboración propia

HU19 – Reporte historia clínica	
Descripción: La historia de usuario describe la forma en que el usuario encargado pueda generar un reporte de la historia clínica de la mascota seleccionada en la fecha seleccionada impresa o en PDF.	
Prioridad: Medio	Estimación: 68
Criterios de aceptación: <ul style="list-style-type: none"> ✓ DADO QUE soy un usuario encargado y quiero generar un reporte de una historia clínica CUANDO ingreso a la interfaz de historiales clínicos ENTONCES el sistema me muestra el listado general de las mascotas registradas en el sistema con sus respectivos historiales. ✓ DADO QUE deseo filtrar historiales clínicos 	

<p>CUANDO digito dentro de un pequeño buscador en la interfaz donde se visualizan los historiales clínicos de cada mascota ENTONCES el sistema filtrará mediante la fecha en tiempo real.</p> <ul style="list-style-type: none">✓ DADO QUE deseo generar la ficha clínica CUANDO presiono en el icono de impresora ubicado a la derecha inferior del campo seleccionado ENTONCES el sistema generará una interfaz con los datos de la atención clínica seleccionada.
<p>Tareas:</p> <ul style="list-style-type: none">✓ Realizar el análisis, entendimiento y diseño de la historia de usuario.✓ Realizar el modelo de datos.✓ Diseñar los prototipos correspondientes.✓ Realizar el desarrollo de la historia de usuario.✓ Diseñar casos de prueba.✓ Ejecutar los casos de prueba planteados.✓ Solucionar los defectos encontrados.✓ Desplegar la historia de usuario.
<p>Definiciones adicionales:</p> <ul style="list-style-type: none">✓ Al momento de presionar el botón IMPRIMIR se abre una nueva ventana para dar las configuraciones necesarias para la impresión junto con la vista previa de la ficha clínica, en caso de no querer la ficha impresa está la opción de guardar en PDF.

Fuente: Elaboración propia

HU20 – Registrar atención clínica	
Descripción: La historia de usuario describe la forma en el que usuario encargado pueda registrar una nueva atención clínica para una respectiva mascota previamente registrada.	
Prioridad: Alto	Estimación: 120
<p>Criterios de aceptación:</p> <ul style="list-style-type: none"> ✓ DADO QUE soy un usuario encargado y quiero registrar a una atención clínica nueva CUANDO ingreso a la interfaz de mascotas, presiono en el botón atención clínica, lleno el formulario presentado con los datos obligatorios ENTONCES se registra en el sistema la atención clínica realizada y se agrega a la lista de historiales clínicos de la mascota seleccionada. ✓ DADO QUE soy un usuario encargado y quiero cancelar el proceso de registro de atención clínica CUANDO ingreso a la interfaz de registro y presiono en la esquina superior derecha la X para cerrar el formulario ENTONCES el sistema me cancela el proceso de registro y me redirige a la interfaz del listado general de mascotas. 	
<p>Tareas:</p> <ul style="list-style-type: none"> ✓ Realizar el análisis, entendimiento y diseño de la historia de usuario. ✓ Realizar el modelo de datos. ✓ Diseñar los prototipos correspondientes. ✓ Realizar el desarrollo de la historia de usuario. ✓ Diseñar casos de prueba. ✓ Ejecutar los casos de prueba planteados. ✓ Solucionar los defectos encontrados. ✓ Desplegar la historia de usuario. 	
<p>Definiciones adicionales:</p> <ul style="list-style-type: none"> ✓ Los campos del formulario de registro son: Médico (obligatorio), Nombre de la mascota (obligatorio), Peso (obligatorio), Raza (obligatorio), Temperatura (obligatorio), Frecuencia Cardíaca (opcional), Frecuencia Respiratoria (obligatorio), Diagnóstico (obligatorio), Servicios (obligatorio), Medicamentos (obligatorio), Vía de administración (obligatorio), Prescripción (obligatorio). ✓ Al saltarse un campo obligatorio sin llenar el sistema mostrará una alerta indicando que se deben completar. 	

- ✓ Al registrar una atención clínica nueva, el sistema muestra un toastify con registro exitoso y redirige automáticamente a la interfaz principal de mascotas.

Fuente: Elaboración propia

HU21 – Generar receta	
Descripción: La historia de usuario describe la forma en que el usuario encargado pueda generar una receta impresa o en PDF con la prescripción de sus medicamentos.	
Prioridad: Bajo	Estimación: 48
Criterios de aceptación:	
<ul style="list-style-type: none"> ✓ DADO QUE soy un usuario encargado y quiero generar una receta con una prescripción CUANDO ingreso a la interfaz de mascotas, presiono en el botón atención clínica, lleno el formulario presentado con los datos obligatorios y presiono el botón para generar Receta ENTONCES se genera una interfaz con los datos de la prescripción y te proporciona la opción de imprimir o guardar en PDF. ✓ DADO QUE soy un usuario encargado y quiero cancelar el proceso de generar receta CUANDO ingreso a la interfaz de receta y presiono el botón Cancelar para cerrar el formulario ENTONCES el sistema me cancela el proceso de impresión o guardado de receta. 	
Tareas:	
<ul style="list-style-type: none"> ✓ Realizar el análisis, entendimiento y diseño de la historia de usuario. ✓ Realizar el modelo de datos. ✓ Diseñar los prototipos correspondientes. ✓ Realizar el desarrollo de la historia de usuario. ✓ Diseñar casos de prueba. ✓ Ejecutar los casos de prueba planteados. ✓ Solucionar los defectos encontrados. ✓ Desplegar la historia de usuario. 	
Definiciones adicionales:	
<ul style="list-style-type: none"> ✓ Al momento de presionar el botón IMPRIMIR se abre una nueva ventana para dar las configuraciones necesarias para la impresión junto con la vista previa de la receta, en caso de no querer la receta impresa está la opción de guardar en PDF. 	

Fuente: Elaboración propia

HU22 – Generar recibo	
Descripción: La historia de usuario describe la forma en el que el usuario encargado pueda generar un recibo impreso o guardarlo en PDF con el monto del costo de los medicamentos administrados en la atención clínica.	
Prioridad: Bajo	Estimación: 48
<p>Criterios de aceptación:</p> <ul style="list-style-type: none"> ✓ DADO QUE soy un usuario encargado y quiero generar un recibo con el monto de los medicamentos administrados CUANDO ingreso a la interfaz de mascotas, presiono en el botón atención clínica, lleno el formulario presentado con los datos obligatorios y presiono el botón para generar Recibo ENTONCES se genera una interfaz con los datos de los medicamentos utilizados y te proporciona la opción de imprimir o guardar en PDF. ✓ DADO QUE soy un usuario encargado y quiero cancelar el proceso de generar Recibo CUANDO ingreso a la interfaz de recibo y presiono el botón Cancelar para cerrar el formulario ENTONCES el sistema me cancela el proceso de impresión o guardado de recibo. 	
<p>Tareas:</p> <ul style="list-style-type: none"> ✓ Realizar el análisis, entendimiento y diseño de la historia de usuario. ✓ Realizar el modelo de datos. ✓ Diseñar los prototipos correspondientes. ✓ Realizar el desarrollo de la historia de usuario. ✓ Diseñar casos de prueba. ✓ Ejecutar los casos de prueba planteados. ✓ Solucionar los defectos encontrados. ✓ Desplegar la historia de usuario. 	
<p>Definiciones adicionales:</p> <ul style="list-style-type: none"> ✓ Al momento de presionar el botón IMPRIMIR se abre una nueva ventana para dar las configuraciones necesarias para la impresión junto con la vista previa de la receta, en caso de no querer la receta impresa está la opción de guardar en PDF. ✓ Es posible modificar las cantidades de los medicamentos administrados con el fin de generar un total de los costos por medicamento. 	

Fuente: Elaboración propia

HU23 – Registrar Medicamentos	
Descripción: La historia de usuario describe la forma en el que usuario encargado pueda registrar un medicamento nuevo.	
Prioridad: Alto	Estimación: 120
<p>Criterios de aceptación:</p> <ul style="list-style-type: none"> ✓ DADO QUE soy un usuario encargado y quiero registrar un medicamento nuevo CUANDO ingreso a la interfaz de inventario, presiono en el botón agregar medicamento, lleno el formulario presentado con los datos obligatorios ENTONCES se registra en el sistema el medicamento nuevo y se muestra automáticamente en el listado general. ✓ DADO QUE soy un usuario encargado y quiero cancelar el proceso de registro de un medicamento nuevo CUANDO ingreso a la interfaz de registro y presiono en la esquina superior derecha la X para cerrar el formulario ENTONCES el sistema me cancela el proceso de registro y me redirige a la interfaz del listado general de medicamentos. 	
<p>Tareas:</p> <ul style="list-style-type: none"> ✓ Realizar el análisis, entendimiento y diseño de la historia de usuario. ✓ Realizar el modelo de datos. ✓ Diseñar el prototipo de acceso al sistema. ✓ Realizar el desarrollo de la historia de usuario. ✓ Diseñar casos de prueba. ✓ Ejecutar los casos de prueba planteados. ✓ Solucionar los defectos encontrados. ✓ Desplegar la historia de usuario 	
<p>Definiciones adicionales:</p> <ul style="list-style-type: none"> ✓ Los campos del formulario de registro son: Nombre (obligatorio), Apellido Paterno (opcional), Apellido Materno (opcional), Celular (obligatorio), Dirección (obligatorio). ✓ Al saltarse un campo obligatorio sin llenar el sistema mostrará una alerta indicando que se deben completar. ✓ Al registrar un medicamento nuevo, el sistema muestra un toastify con registro exitoso y redirige automáticamente a la interfaz principal con el registro nuevo dentro del listado general. 	

Fuente: Elaboración propia

HU24 – Leer Medicamentos Registrados	
Descripción: La historia de usuario describe la forma en la que el usuario encargado visualiza los medicamentos registrados en el sistema.	
Prioridad: Medio	Estimación: 72
Criterios de aceptación:	
<ul style="list-style-type: none"> ✓ DADO QUE soy un usuario encargado y quiero visualizar todos los medicamentos registrados en el sistema CUANDO ingreso a la interfaz de inventario ENTONCES el sistema me muestra el listado general con los medicamentos registrados en la base de datos. ✓ DADO QUE deseo filtrar registros CUANDO digito dentro de un pequeño buscador en la interfaz principal de medicamentos ENTONCES el sistema filtrará mediante cada columna en tiempo real. 	
Tareas:	
<ul style="list-style-type: none"> ✓ Realizar el análisis, entendimiento y diseño de la historia de usuario. ✓ Realizar el modelo de datos. ✓ Diseñar el prototipo de acceso al sistema. ✓ Realizar el desarrollo de la historia de usuario. ✓ Diseñar casos de prueba. ✓ Ejecutar los casos de prueba planteados. ✓ Solucionar los defectos encontrados. ✓ Desplegar la historia de usuario 	

Fuente: Elaboración propia

HU25 – Actualizar datos de Medicamentos	
Descripción: La historia de usuario describe la forma en que un usuario actualiza los datos de un registro.	
Prioridad: Alto	Estimación: 120
<p>Criterios de aceptación:</p> <ul style="list-style-type: none"> ✓ DADO QUE deseo actualizar los datos de un registro CUANDO selecciono un registro dentro del listado general Y corrijo los datos correspondientes en el formulario ENTONCES el sistema me actualiza el registro con los datos que corregí. ✓ DADO QUE deseo cancelar la actualización de un registro CUANDO seleccione la X de la parte superior derecha del formulario ENTONCES el sistema cancela la desactivación del registro. 	
<p>Tareas:</p> <ul style="list-style-type: none"> ✓ Realizar el análisis, entendimiento y diseño de la historia de usuario. ✓ Realizar el modelo de datos. ✓ Diseñar el prototipo de acceso al sistema. ✓ Realizar el desarrollo de la historia de usuario. ✓ Diseñar casos de prueba. ✓ Ejecutar los casos de prueba planteados. ✓ Solucionar los defectos encontrados. ✓ Desplegar la historia de usuario 	
<p>Definiciones adicionales:</p> <ul style="list-style-type: none"> ✓ Se creará un botón en cada fila del listado general para poder actualizar el registro que me redirigirá al formulario correspondiente. ✓ Los campos del formulario de registro son: Nombre (obligatorio), Cantidad (obligatorio), Precio (obligatorio), Tipo de Medicamento (obligatorio). ✓ Al saltarse un cambio obligatorio sin llenar el sistema mostrará una alerta indicando que se deben completar. ✓ Al guardar un medicamento con los campos a actualizar, el sistema muestra un toastify con registro exitoso y redirige automáticamente a la interfaz principal con el registro actualizado dentro del listado general. 	

Fuente: Elaboración propia

HU26 – Eliminar Medicamento	
Descripción: La historia de usuario describe la forma en que un usuario elimina el registro de un medicamento seleccionado.	
Prioridad: Medio	Estimación: 72
Criterios de aceptación: <ul style="list-style-type: none"> ✓ DADO QUE deseo eliminar un registro CUANDO seleccione el registro dentro del listado general ENTONCES el sistema desactiva el registro. ✓ DADO QUE deseo cancelar la eliminación del registro CUANDO seleccione la opción CANCELAR de la parte inferior de la ventana de confirmación ENTONCES el sistema cancela la eliminación del registro 	
Tareas: <ul style="list-style-type: none"> ✓ Realizar el análisis, entendimiento y diseño de la historia de usuario. ✓ Realizar el modelo de datos. ✓ Diseñar el prototipo de acceso al sistema. ✓ Realizar el desarrollo de la historia de usuario. ✓ Diseñar casos de prueba. ✓ Ejecutar los casos de prueba planteados. ✓ Solucionar los defectos encontrados. ✓ Desplegar la historia de usuario 	
Definiciones adicionales: <ul style="list-style-type: none"> ✓ Se creará un botón en cada fila del listado general para poder dar de baja el registro. ✓ Al momento de seleccionar el registro a desactivar, aparecerá un pop-up para confirmar la desactivación del registro. ✓ Al confirmar la desactivación el sistema muestra un toastify con mensaje de desactivado exitoso y redirige automáticamente a la interfaz principal con el registro eliminado del listado general. 	

Fuente: Elaboración propia

HU27 – Dashboard (Reportes)	
Descripción: La historia de usuario describe la forma en la que el usuario encargado visualiza los Reportes relevantes del sistema.	
Prioridad: Alto	Estimación: 120
Criterios de aceptación: <ul style="list-style-type: none"> ✓ DADO QUE soy un usuario encargado y quiero visualizar todos los reportes del sistema CUANDO ingreso a la interfaz dashboard 	

ENTONCES el sistema me muestra los gráficos de los siguientes reportes: servicios más realizados por mes, top medicamentos administrados por mes, médicos que realizaron más atenciones por mes, mascotas atendidas según su género y especie por mes, y un reporte general por año de atenciones clínicas.

Tareas:

- ✓ Realizar el análisis, entendimiento y diseño de la historia de usuario.
- ✓ Realizar el modelo de datos.
- ✓ Diseñar el prototipo de acceso al sistema.
- ✓ Realizar el desarrollo de la historia de usuario.
- ✓ Diseñar casos de prueba.
- ✓ Ejecutar los casos de prueba planteados.
- ✓ Solucionar los defectos encontrados.
- ✓ Desplegar la historia de usuario

Fuente: Elaboración propia

II.1.2.2.3.3 Análisis y Diseño

A continuación, se presenta los diagramas de actividad de las historias descritas en el Backlog para el presente Sprint. No se detallará a profundidad los diagramas por ser una metodología ágil, lo mencionado anteriormente se encuentra dentro de la descripción de cada historia de usuario.

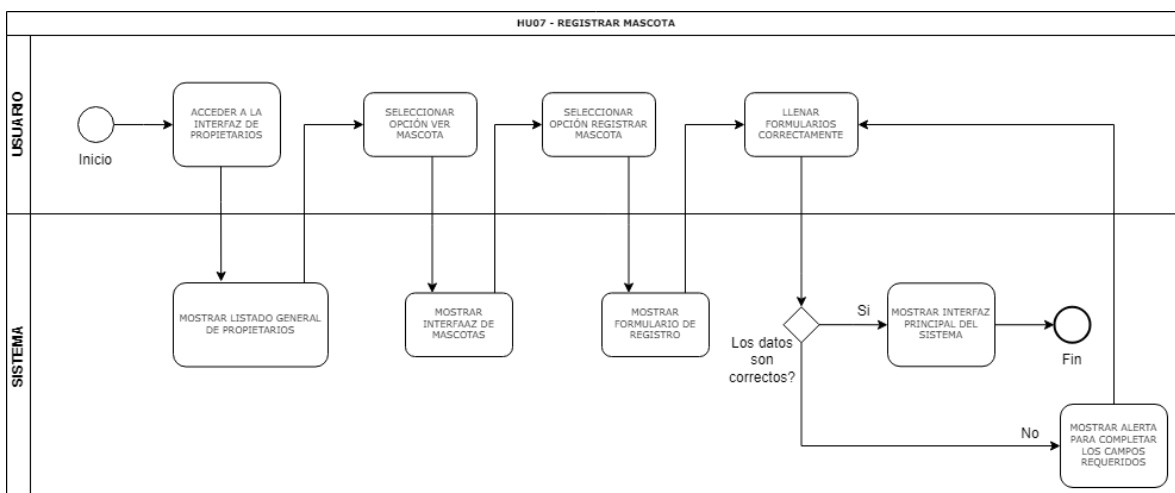


Figura 71: Diagrama de actividad HU07 – Registrar Mascotas

Fuente: Elaboración propia

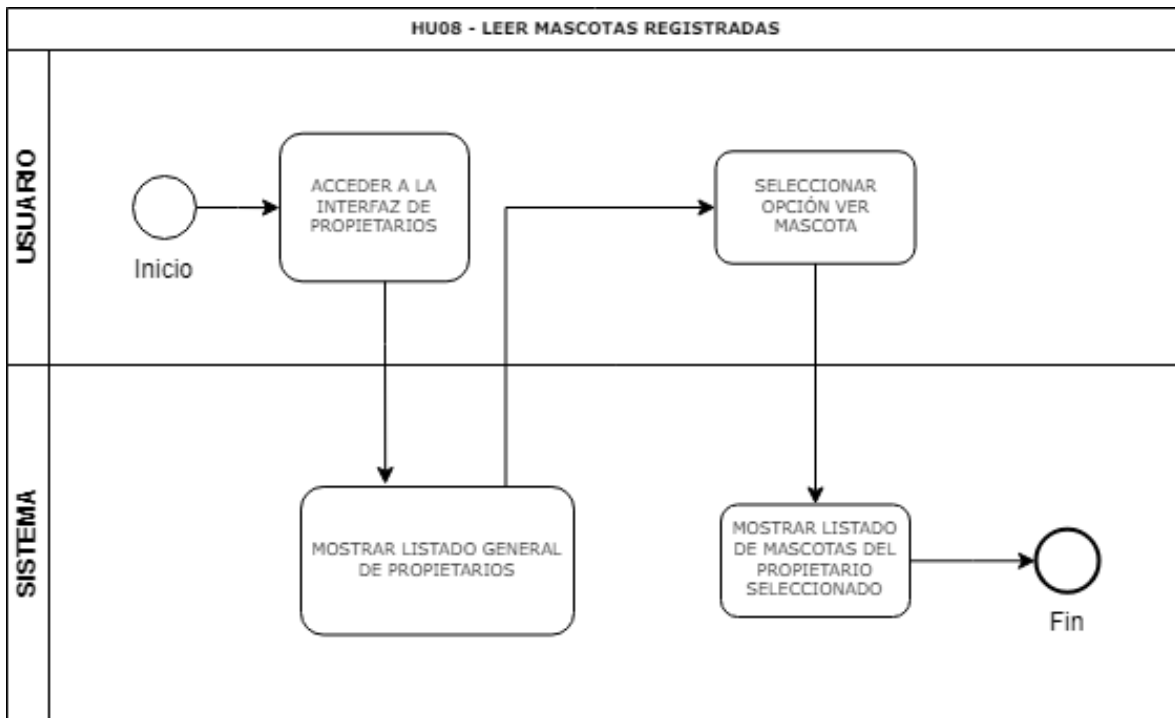


Figura 72: Diagrama de actividad HU08 – Leer mascotas registradas

Fuente: Elaboración propia

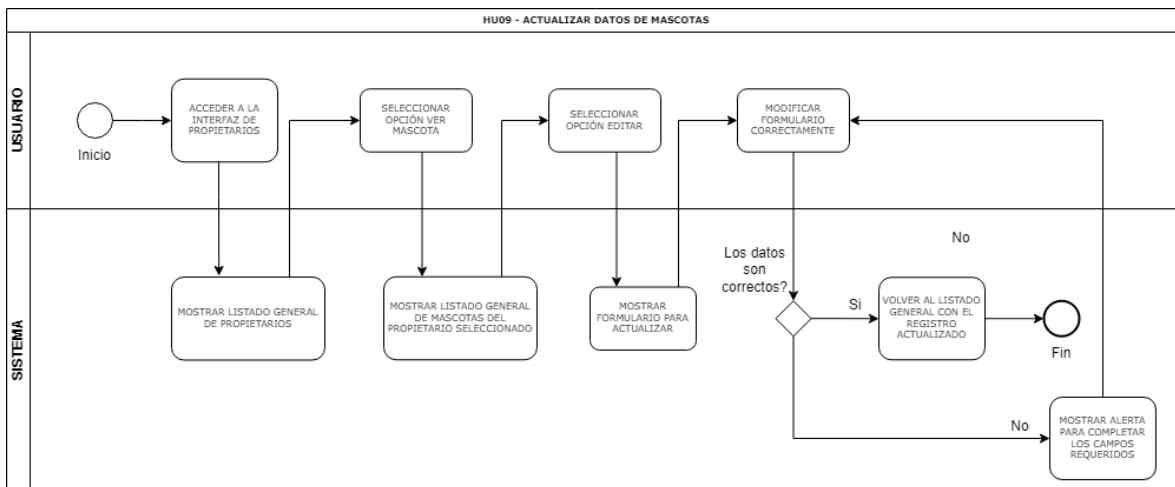


Figura 73: Diagrama de actividad HU09 – Actualizar datos de mascotas

Fuente: Elaboración propia

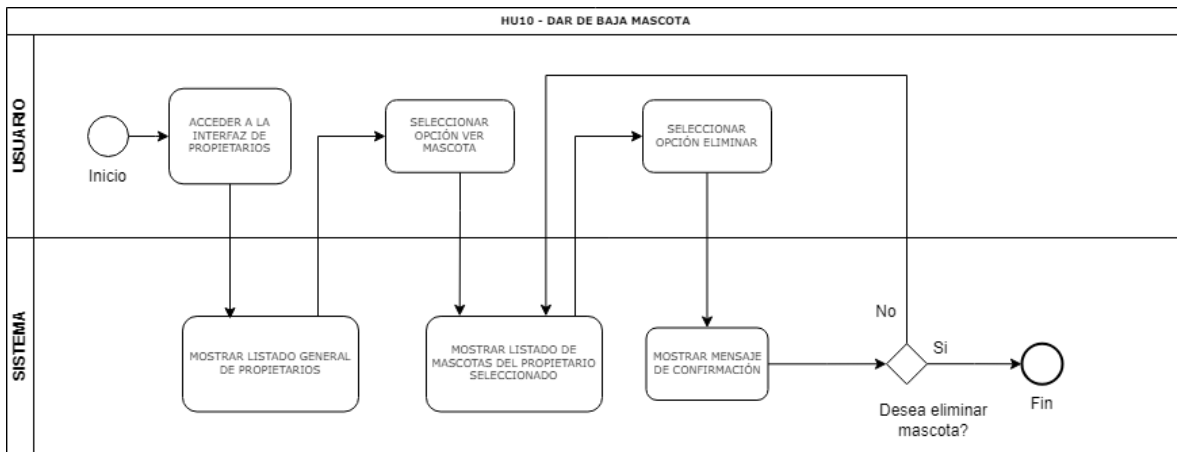


Figura 74: Diagrama de actividad HU10 – Dar de baja mascota

Fuente: Elaboración propia

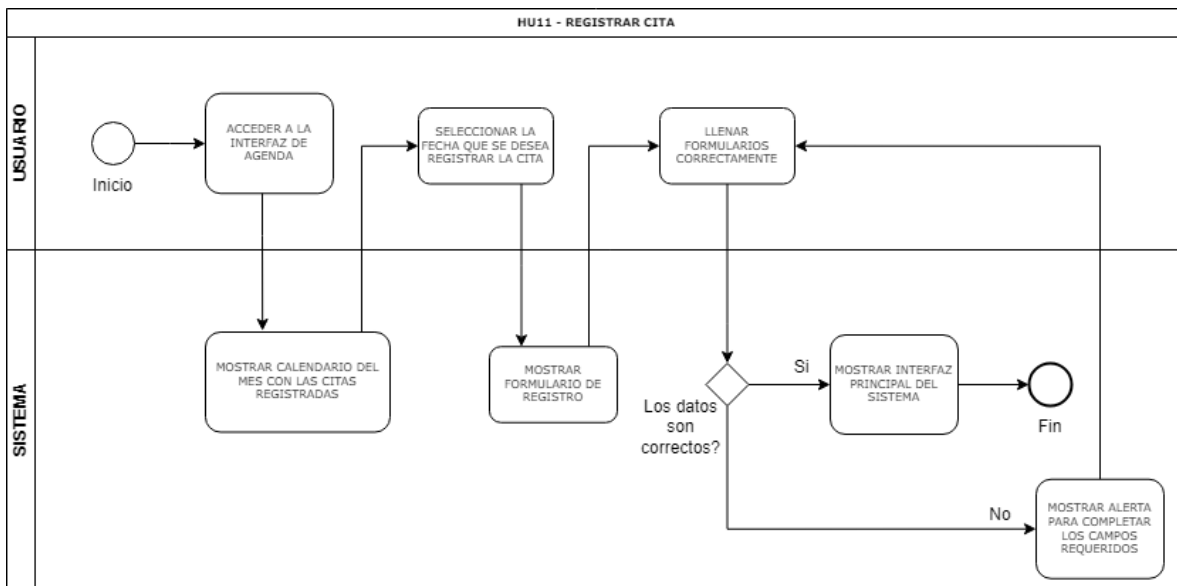


Figura 75: Diagrama de actividad HU11 – Registrar cita

Fuente: Elaboración propia

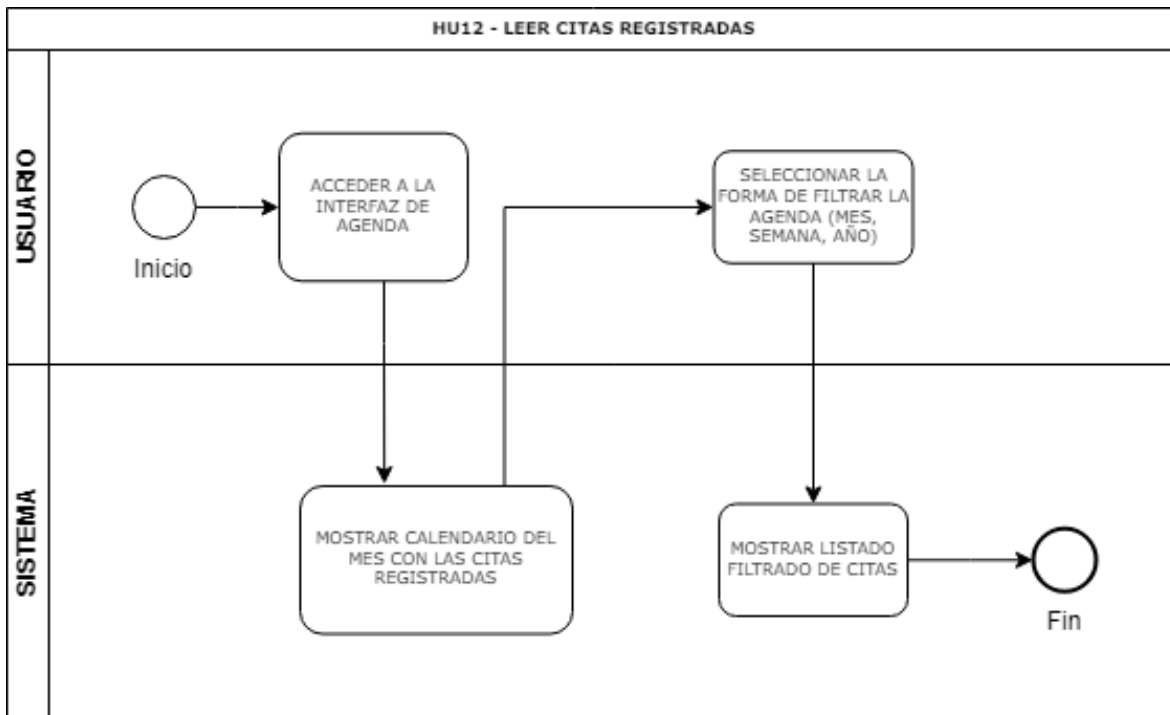


Figura 76: Diagrama de actividad HU12 – Leer citas registradas

Fuente: Elaboración propia

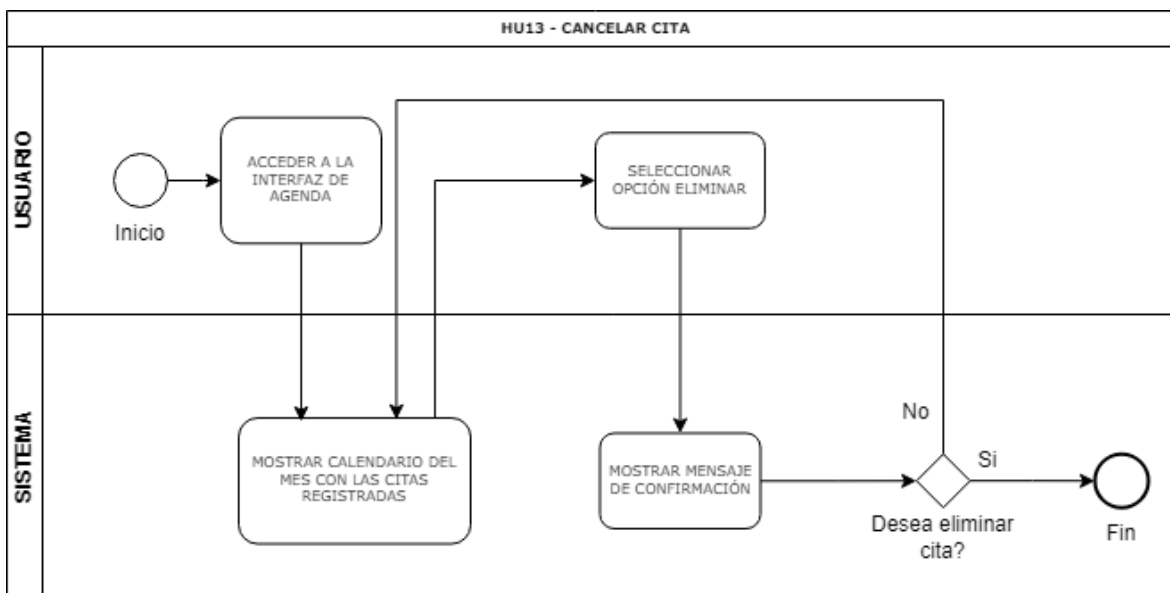


Figura 77: Diagrama de actividad HU13 – Cancelar cita

Fuente: Elaboración propia

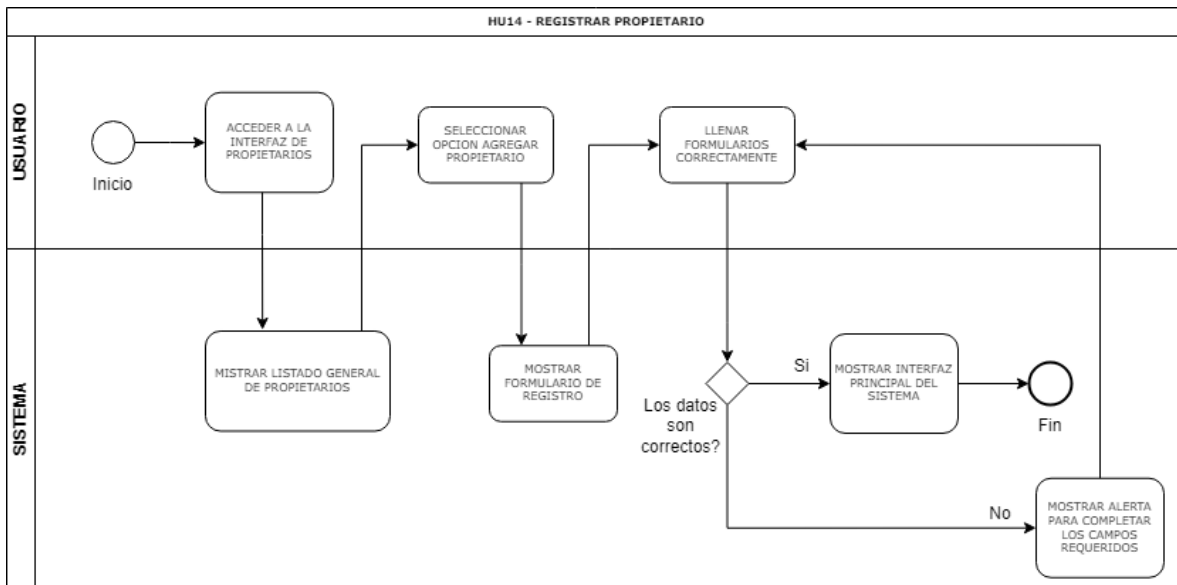


Figura 78: Diagrama de actividad HU14 – Registrar propietario

Fuente: Elaboración propia

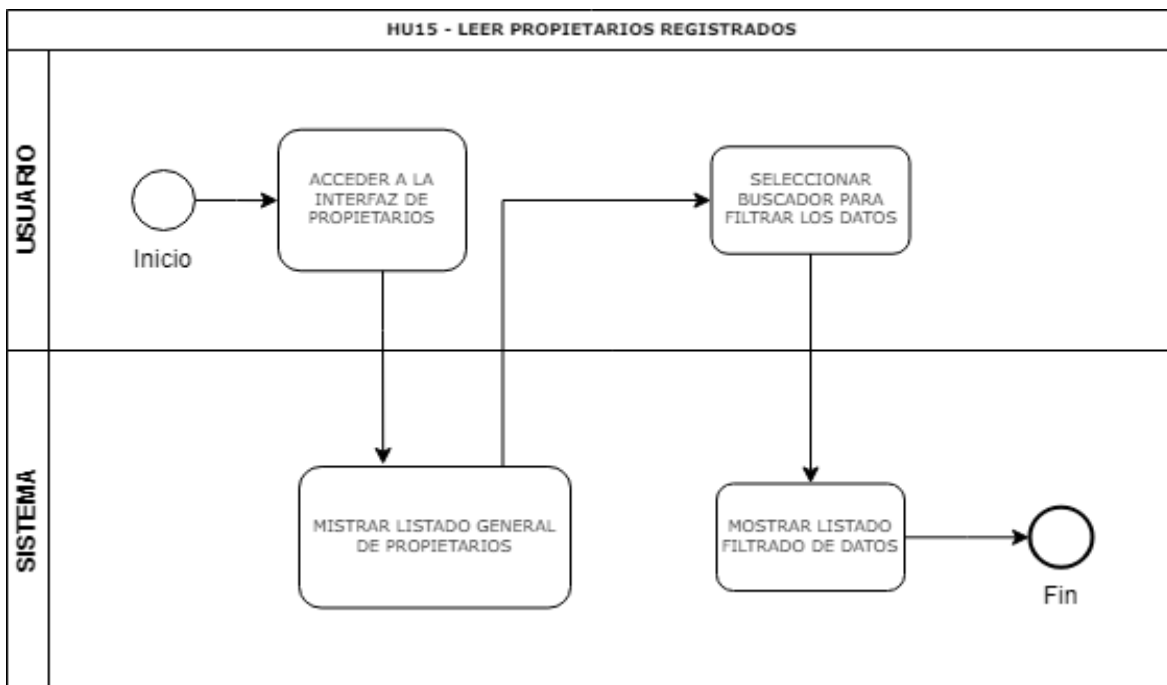


Figura 79: Diagrama de actividad HU15 – Leer propietarios registrados

Fuente: Elaboración propia

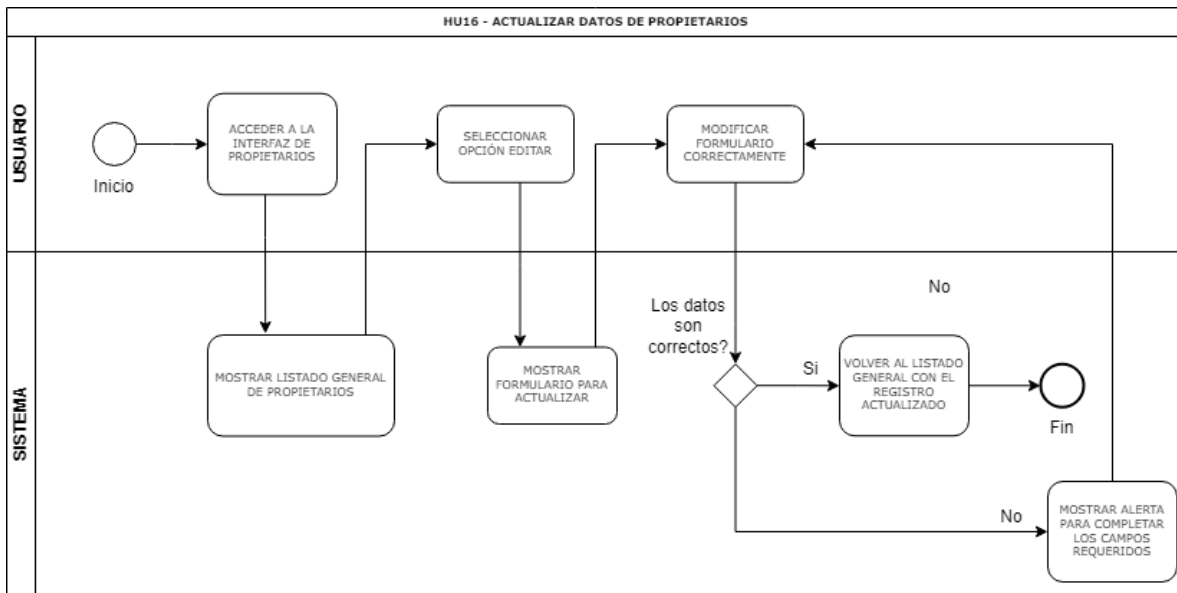


Figura 80: Diagrama de actividad HU16 – Actualizar datos de propietarios

Fuente: Elaboración propia

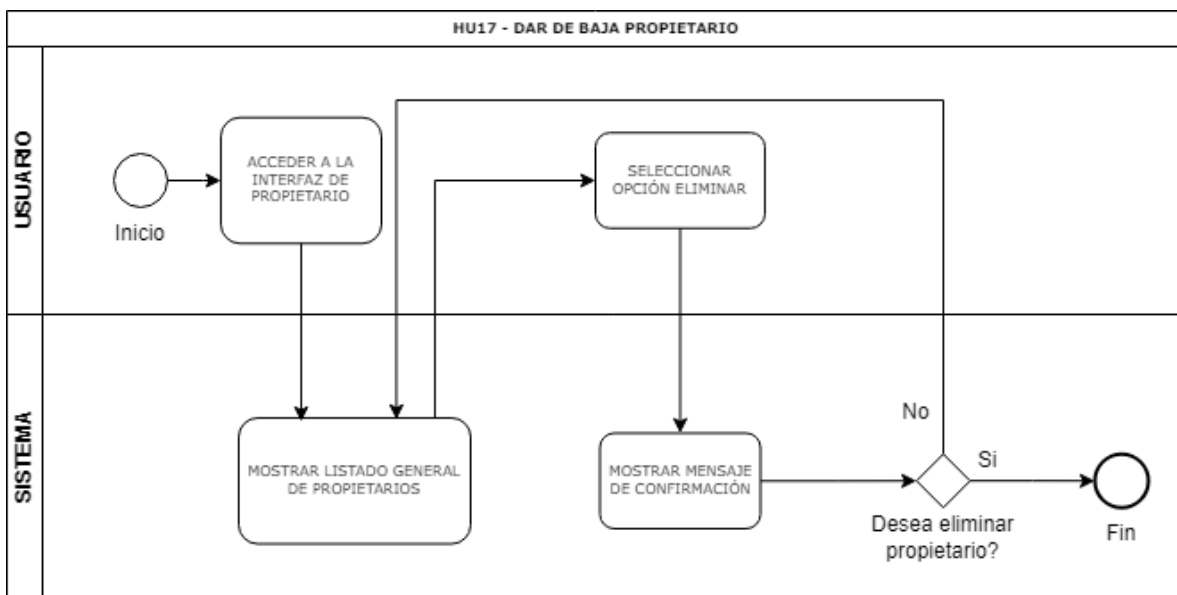


Figura 81: Diagrama de actividad HU17 – Dar de baja propietario

Fuente: Elaboración propia

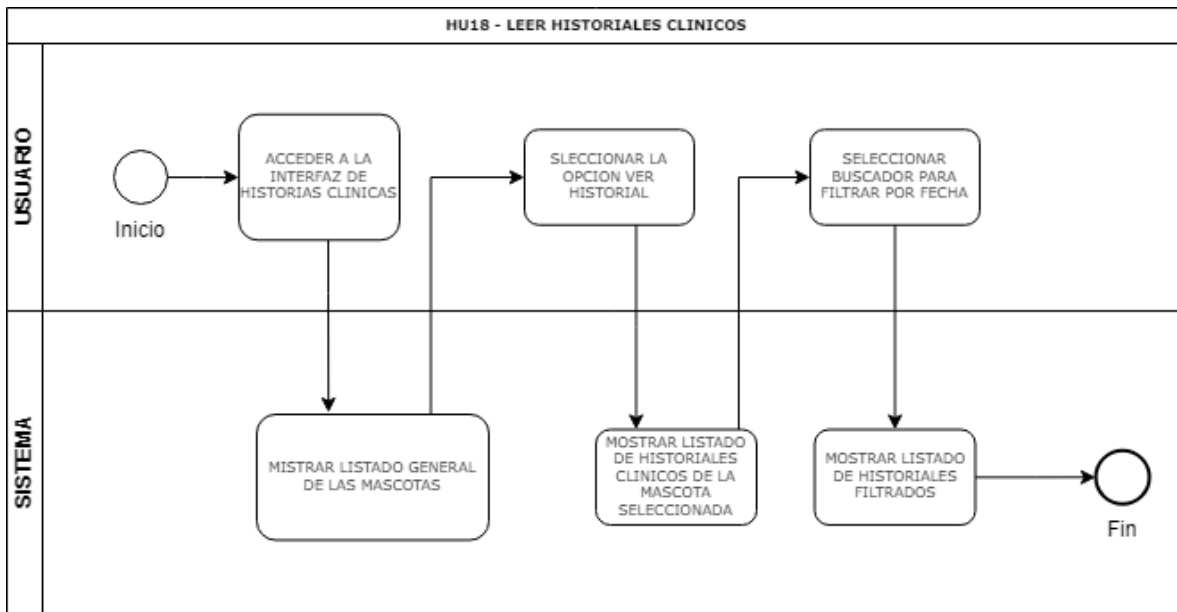


Figura 82: Diagrama de actividad HU18 – Leer historiales clínicos

Fuente: Elaboración propia

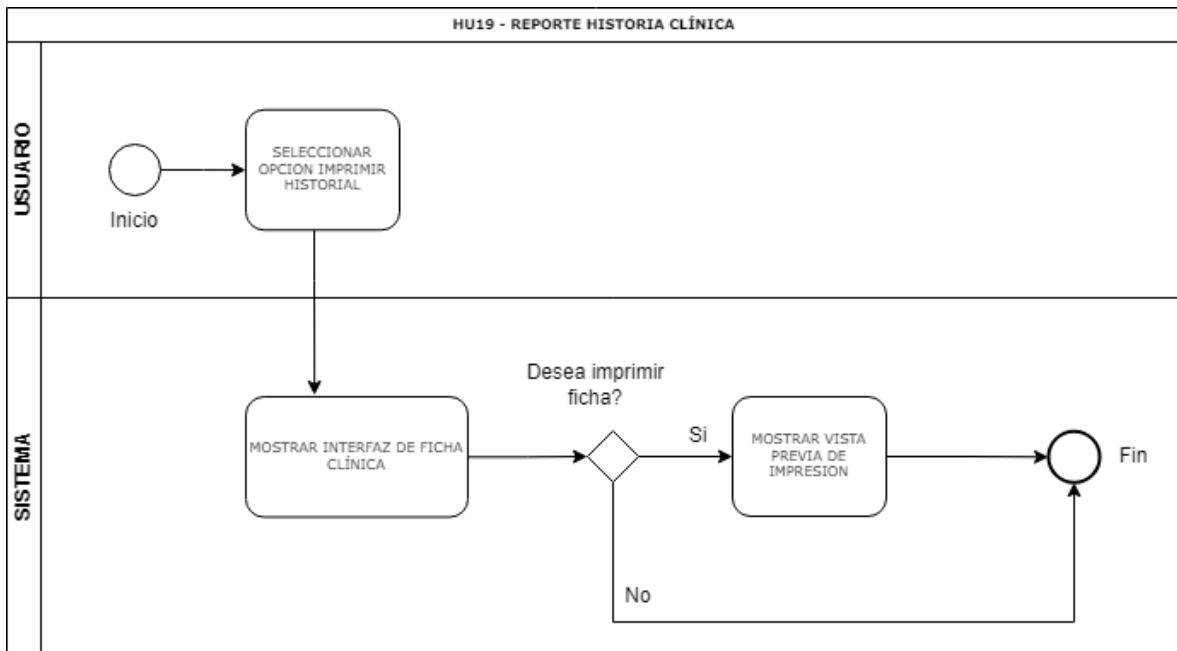


Figura 83: Diagrama de actividad HU19 - Reporte Historia Clínica

Fuente: Elaboración propia

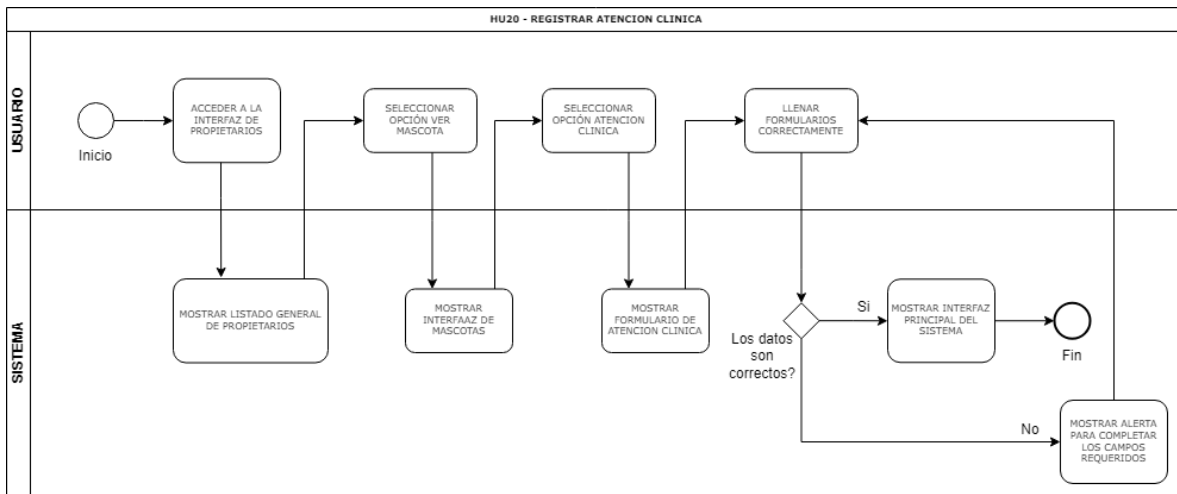


Figura 84: Diagrama de actividad HU20 – Registrar atención clínica

Fuente: Elaboración propia

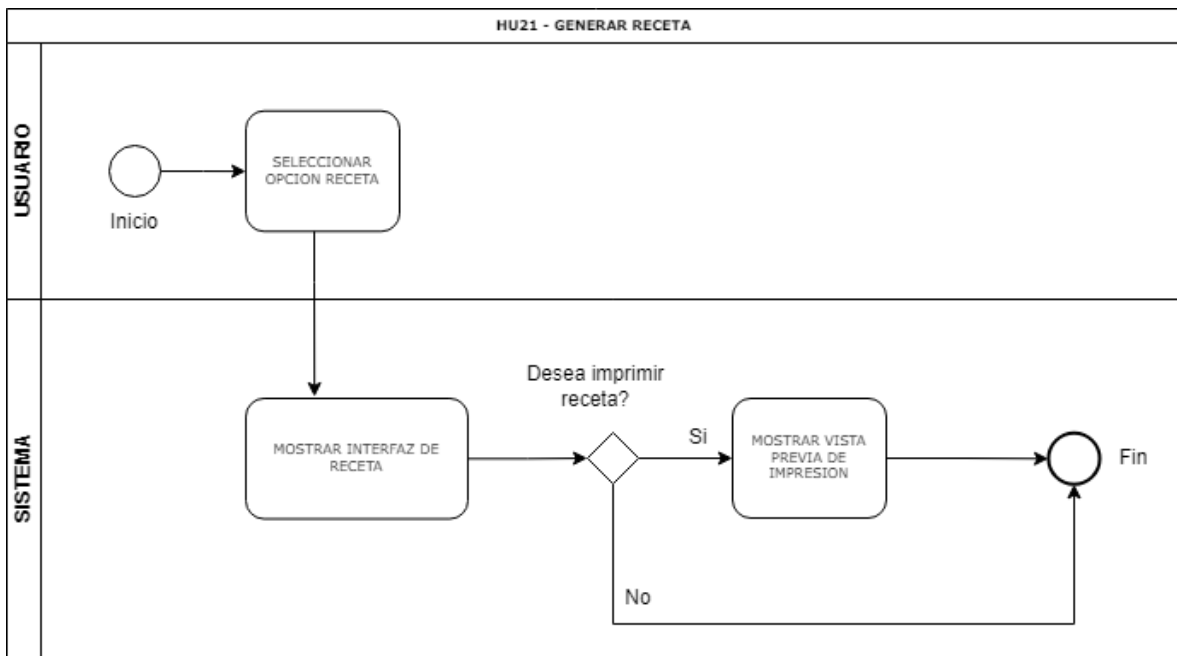


Figura 85: Diagrama de actividad HU21 – Generar receta

Fuente: Elaboración propia

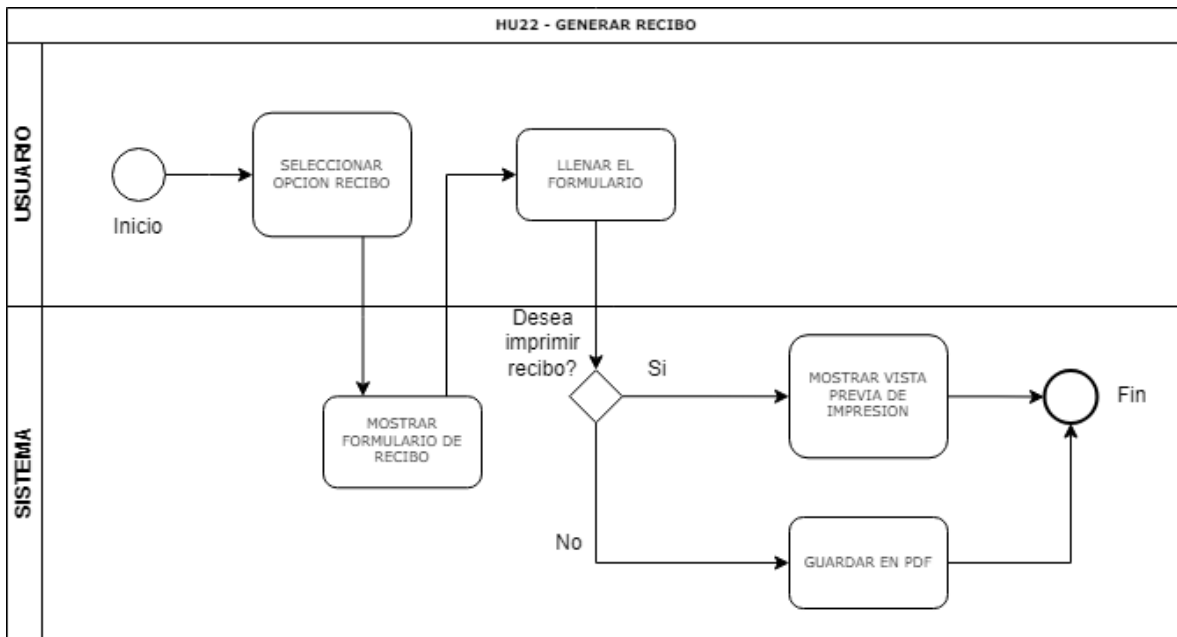


Figura 86: Diagrama de actividad HU22 – Generar recibo

Fuente: Elaboración propia

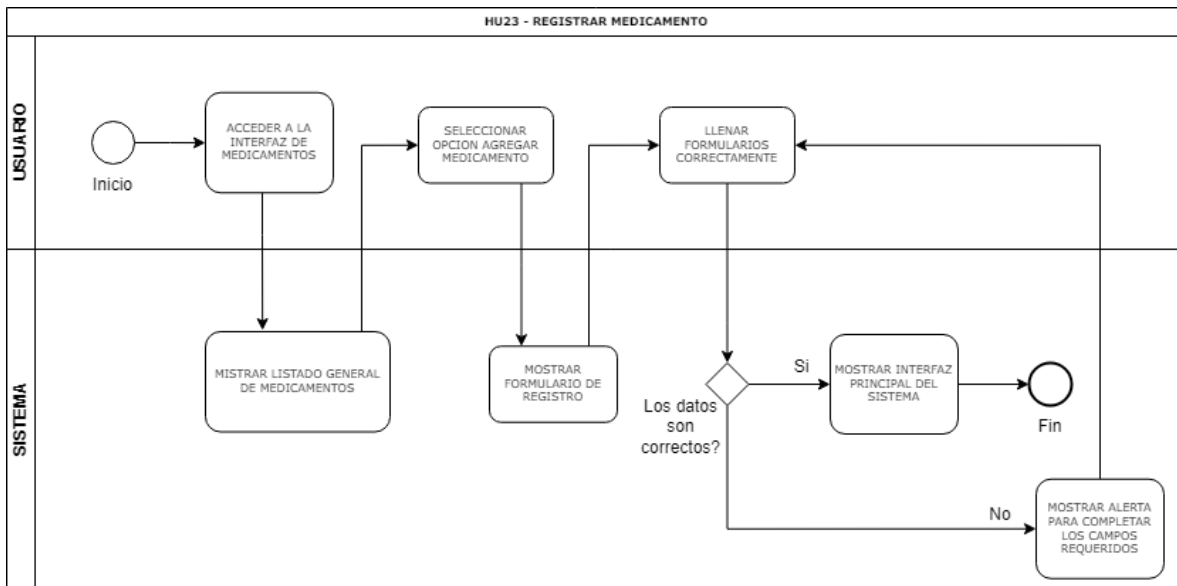


Figura 87: Diagrama de actividad HU23 - Registrar Medicamento

Fuente: Elaboración propia

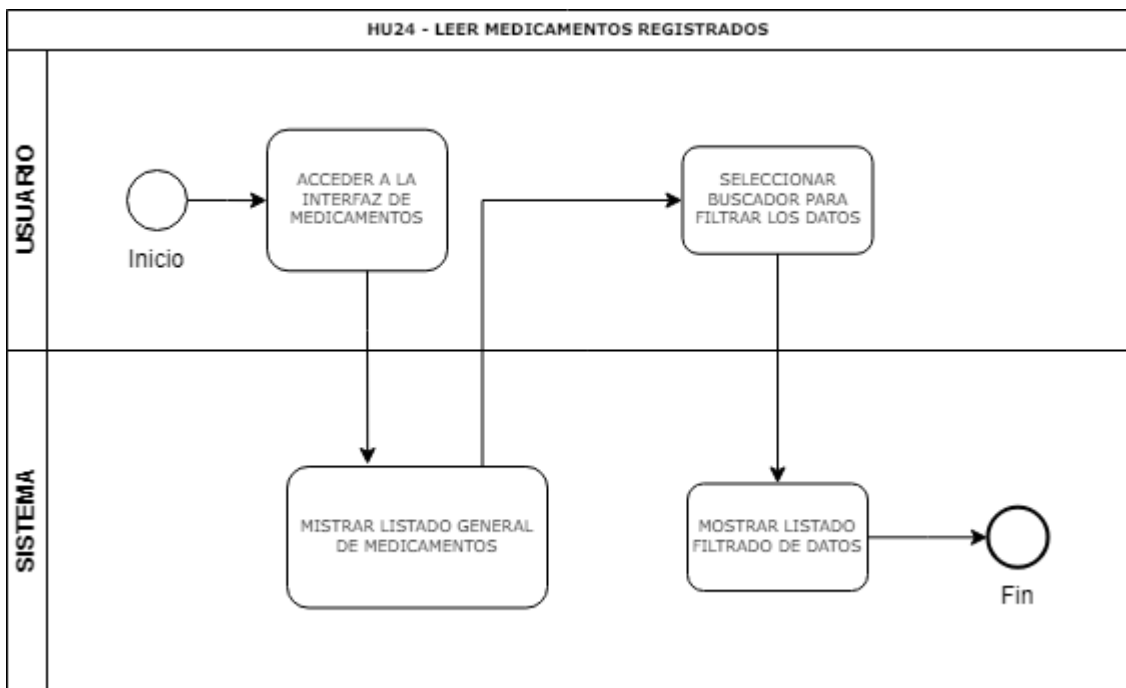


Figura 88: Diagrama de actividad HU24 - Leer medicamentos registrados

Fuente: Elaboración propia

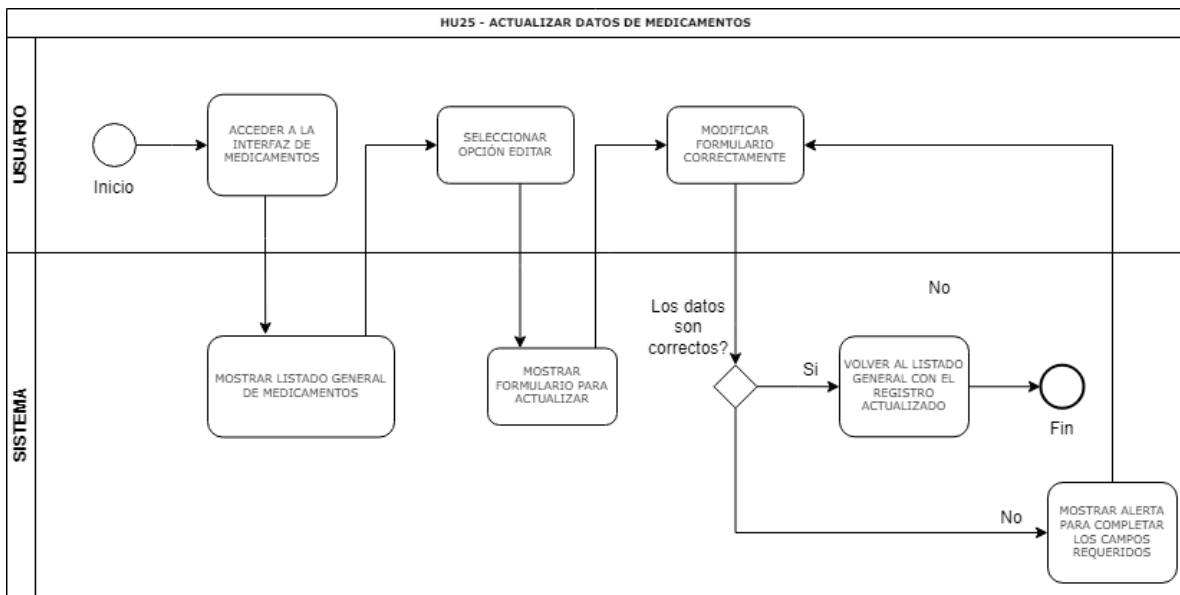


Figura 89: Diagrama de actividad HU25 - Actualizar datos de medicamentos

Fuente: Elaboración propia

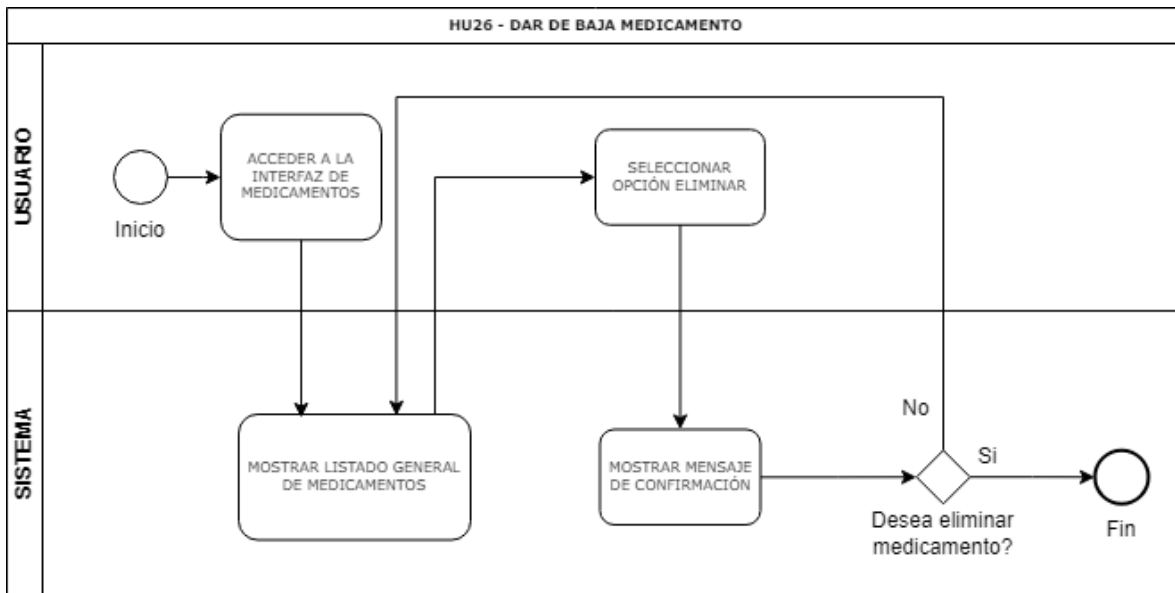


Figura 90: Diagrama de actividad HU26 - Eliminar medicamento

Fuente: Elaboración propia

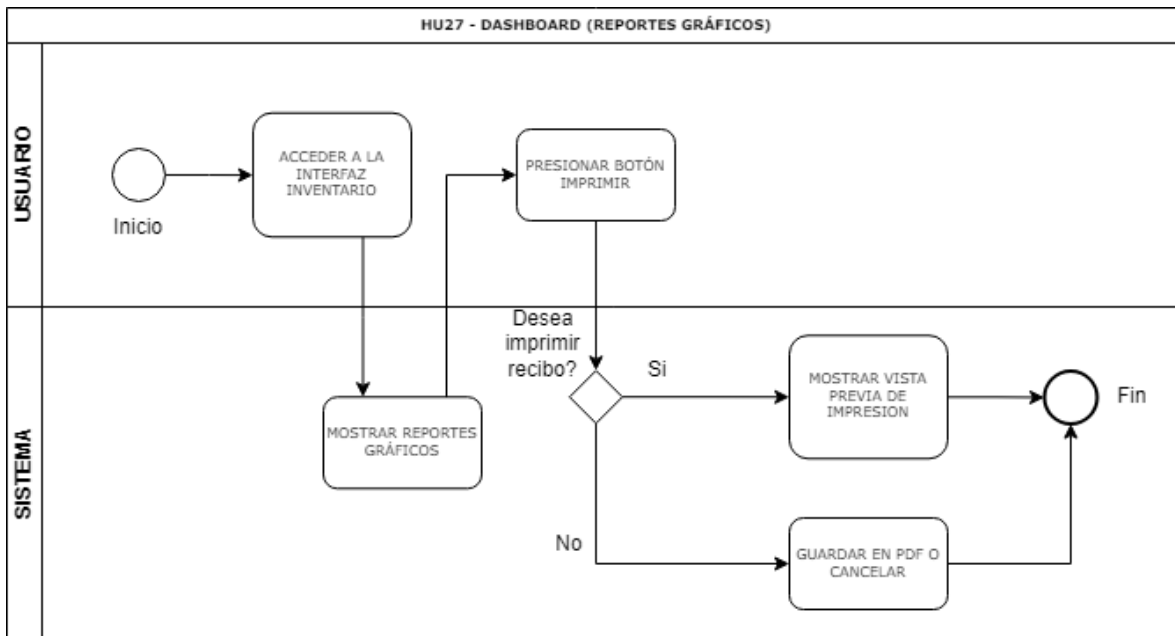


Figura 91: Diagrama de actividad HU27 – Dashboard

Fuente: Elaboración propia

A continuación, se presenta los diagramas de secuencia de las historias descritas en el Backlog para el presente Sprint. No se detallará a profundidad los diagramas por ser una metodología ágil, lo mencionado anteriormente se encuentra dentro de la descripción de cada historia de usuario.

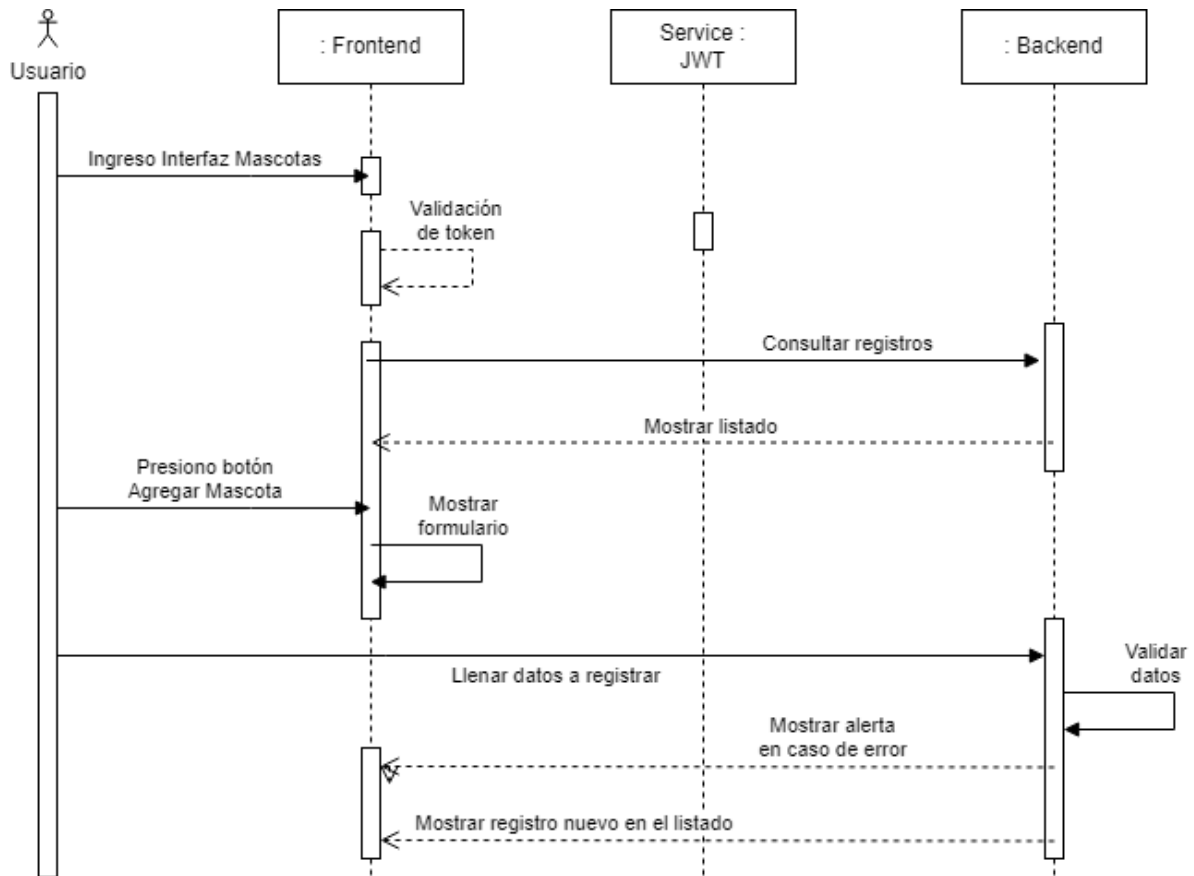


Figura 92: Diagrama de secuencia HU07 – Registrar Mascotas

Fuente: Elaboración propia

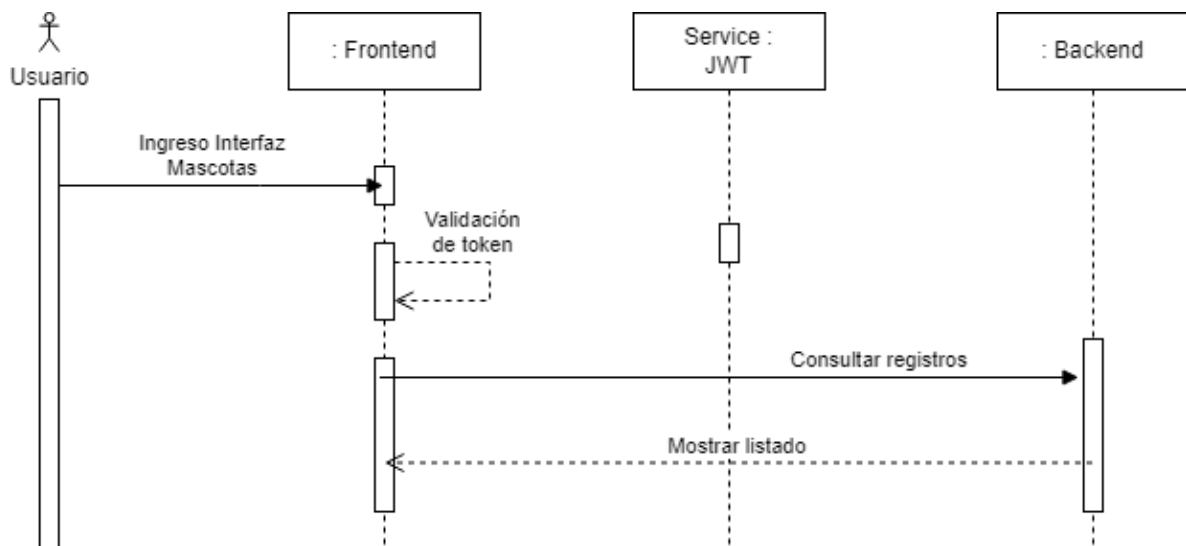


Figura 93: Diagrama de secuencia HU08 – Leer mascotas registradas

Fuente: Elaboración propia

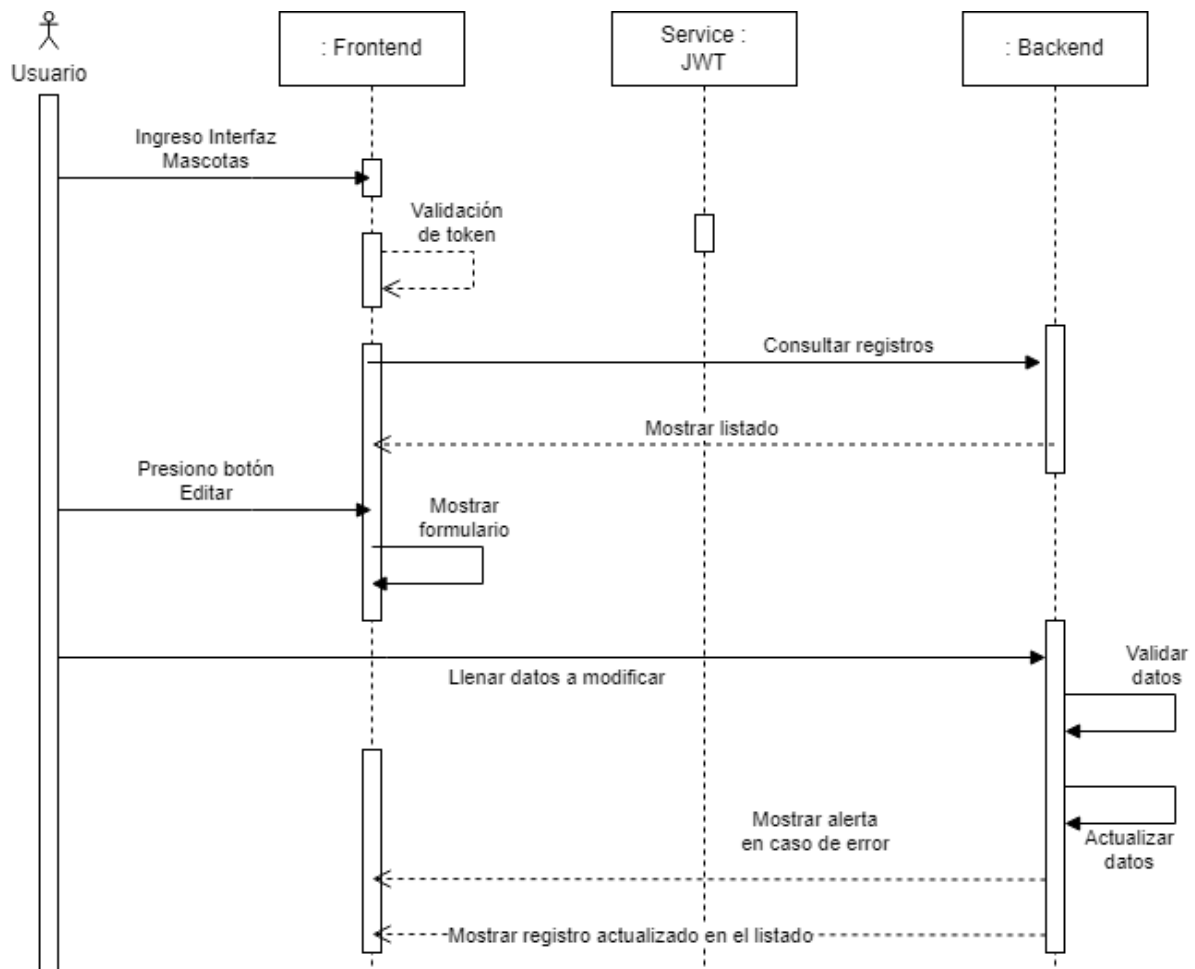


Figura 94: Diagrama de secuencia HU09 – Actualizar datos de mascotas

Fuente: Elaboración propia

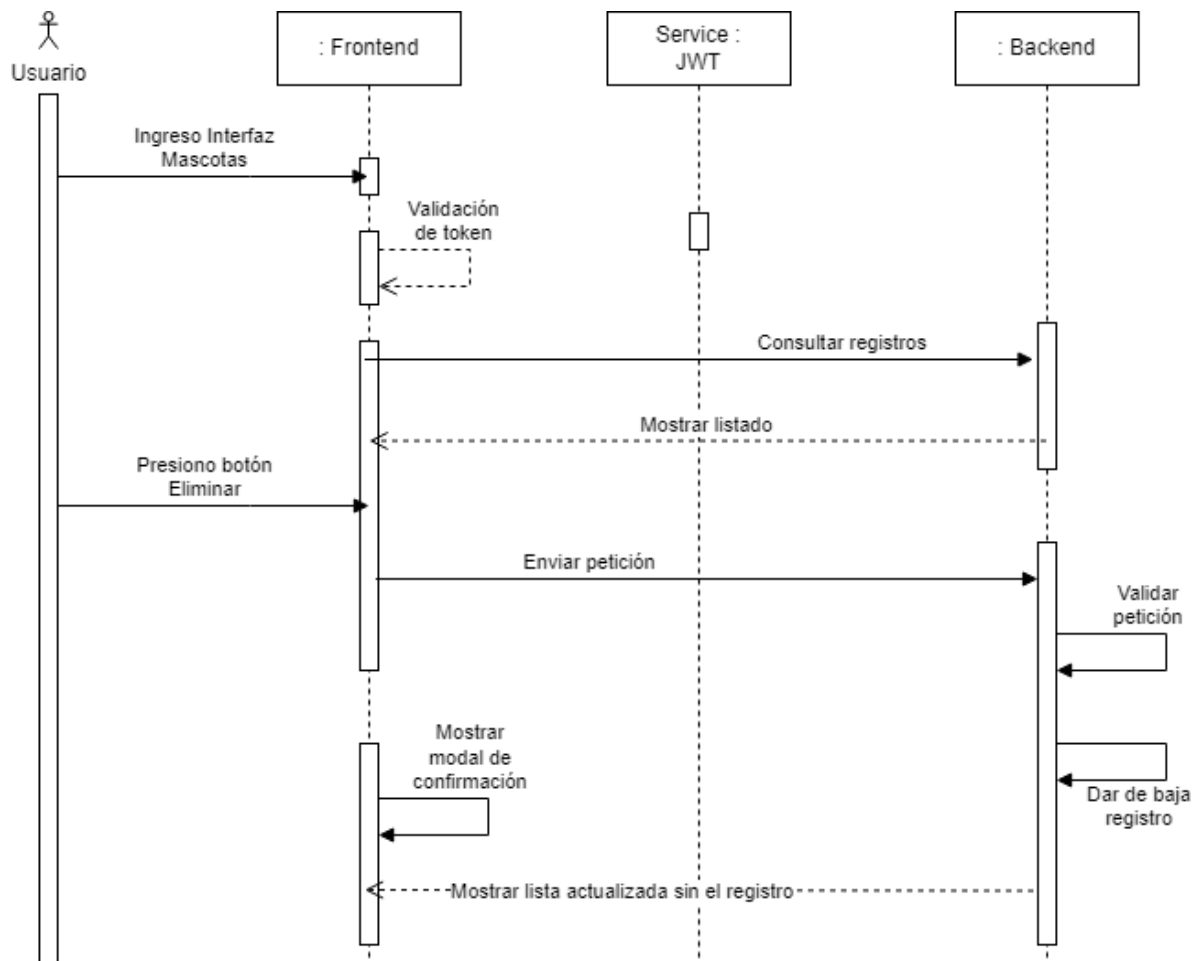


Figura 95: Diagrama de secuencia HU10 – Dar de baja mascota

Fuente: Elaboración propia

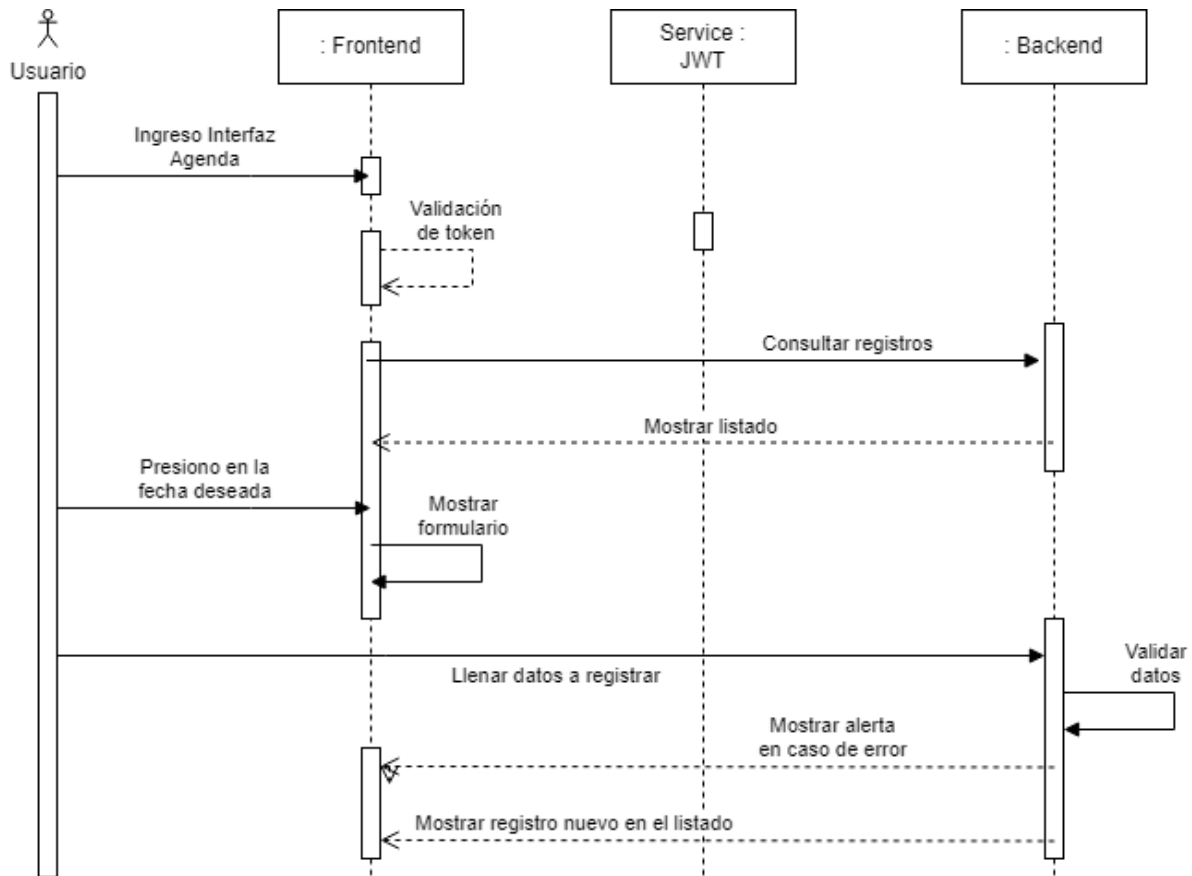


Figura 96: Diagrama de secuencia HU11 – Registrar cita

Fuente: Elaboración propia

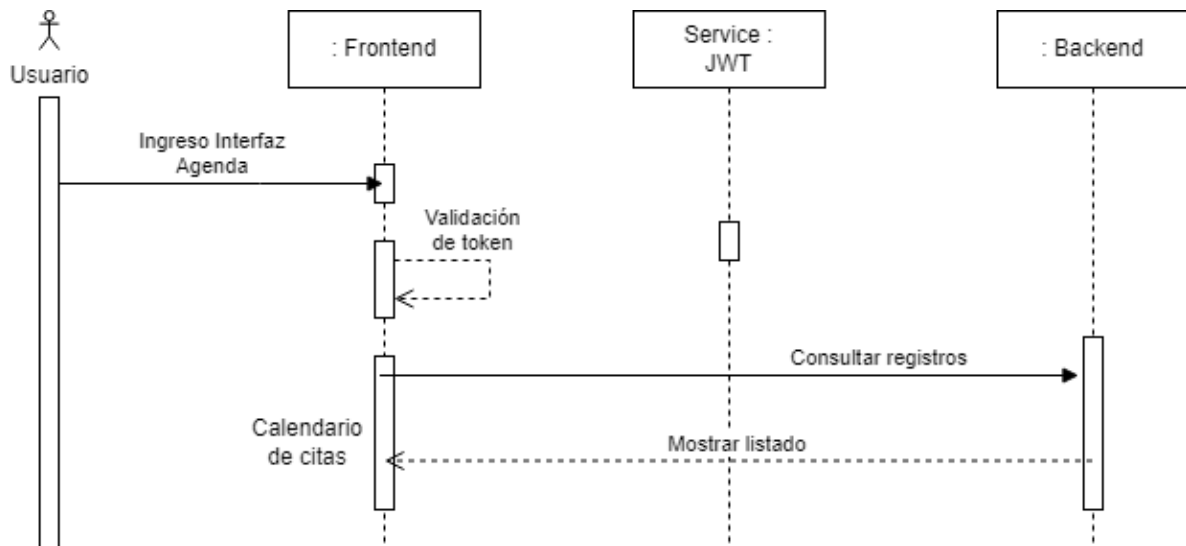


Figura 97: Diagrama de secuencia HU12 – Leer citas registradas

Fuente: Elaboración propia

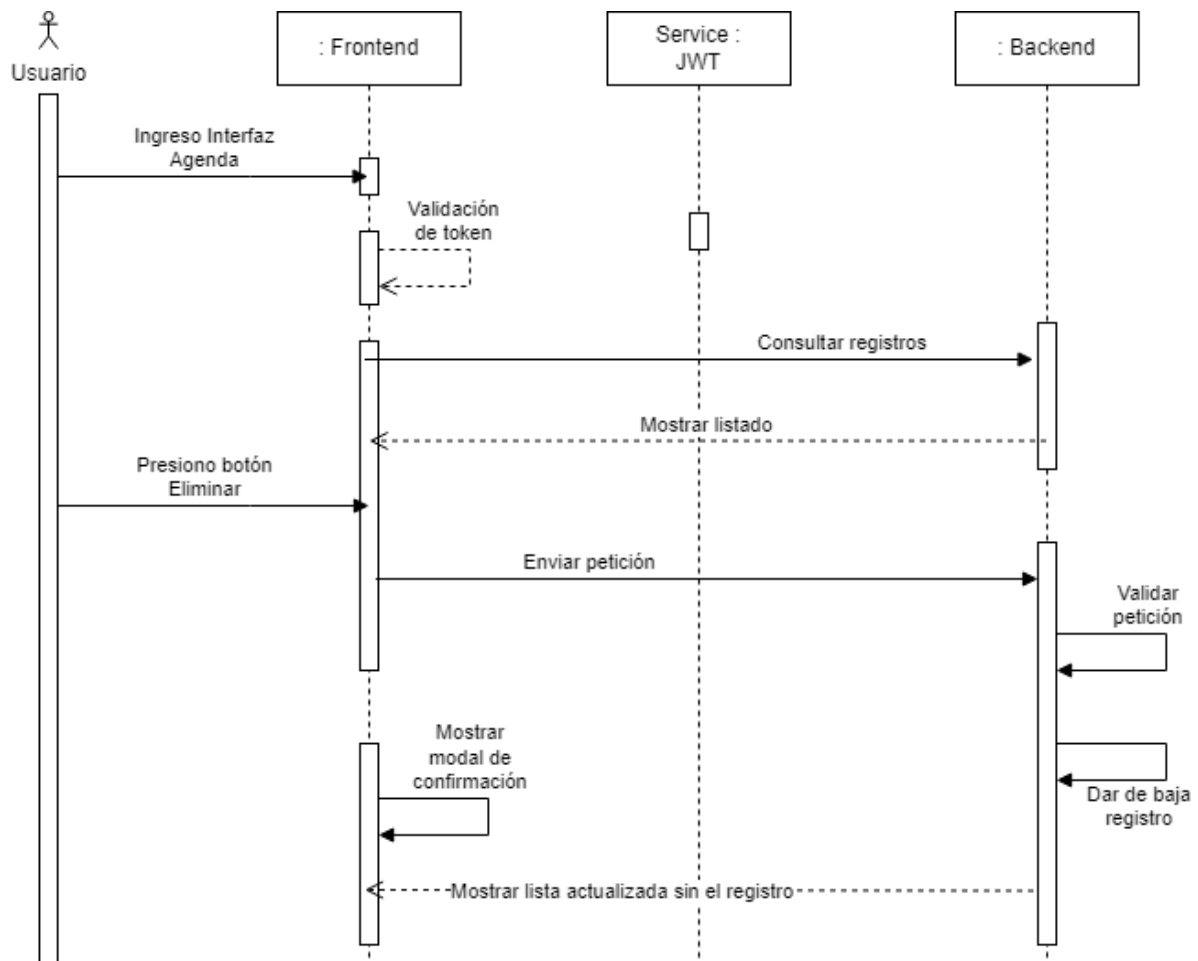


Figura 98: Diagrama de secuencia HU13 – Cancelar cita

Fuente: Elaboración propia

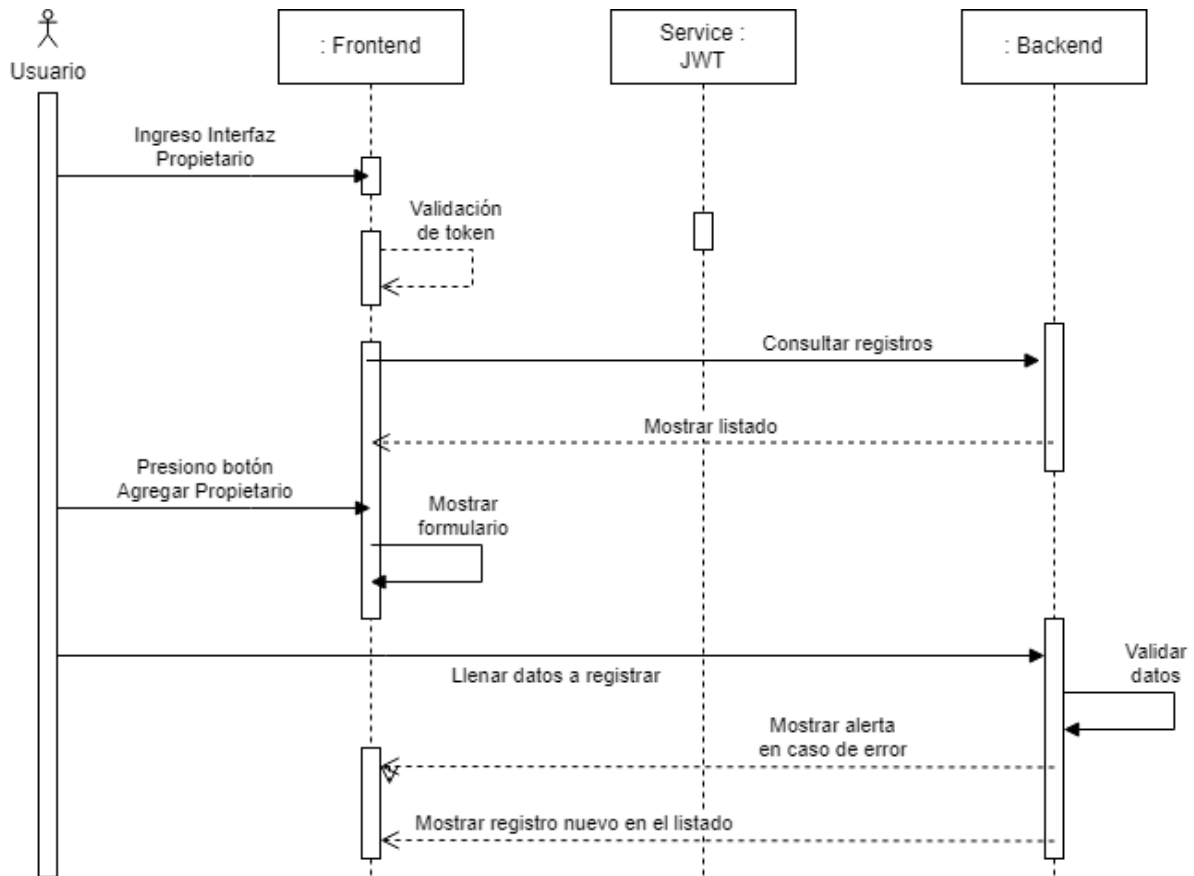


Figura 99: Diagrama de secuencia HU14 – Registrar propietario

Fuente: Elaboración propia

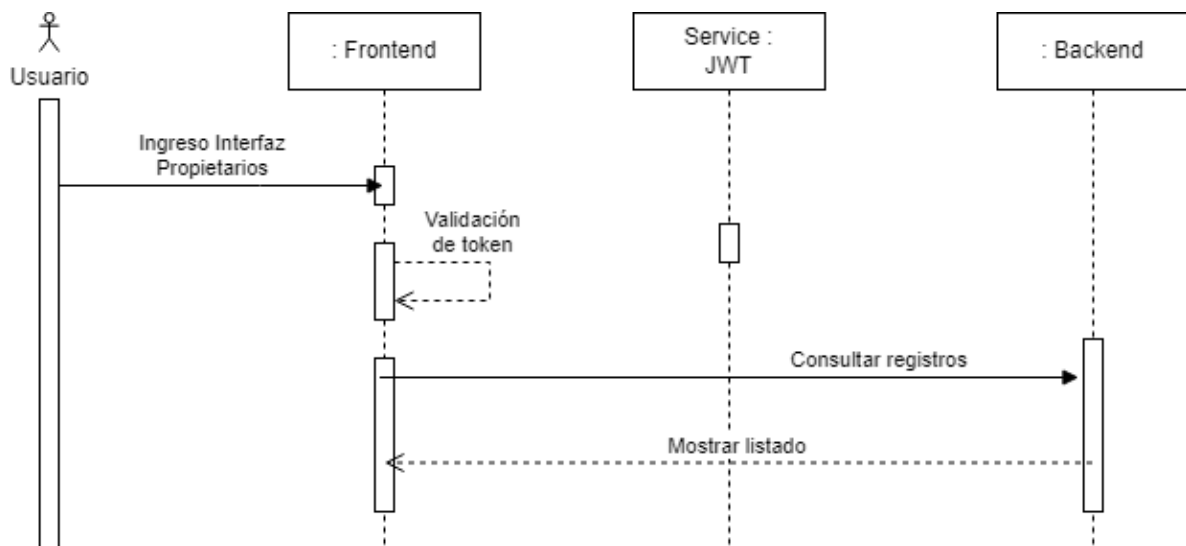


Figura 100: Diagrama de secuencia HU15 – Leer propietarios registrados

Fuente: Elaboración propia

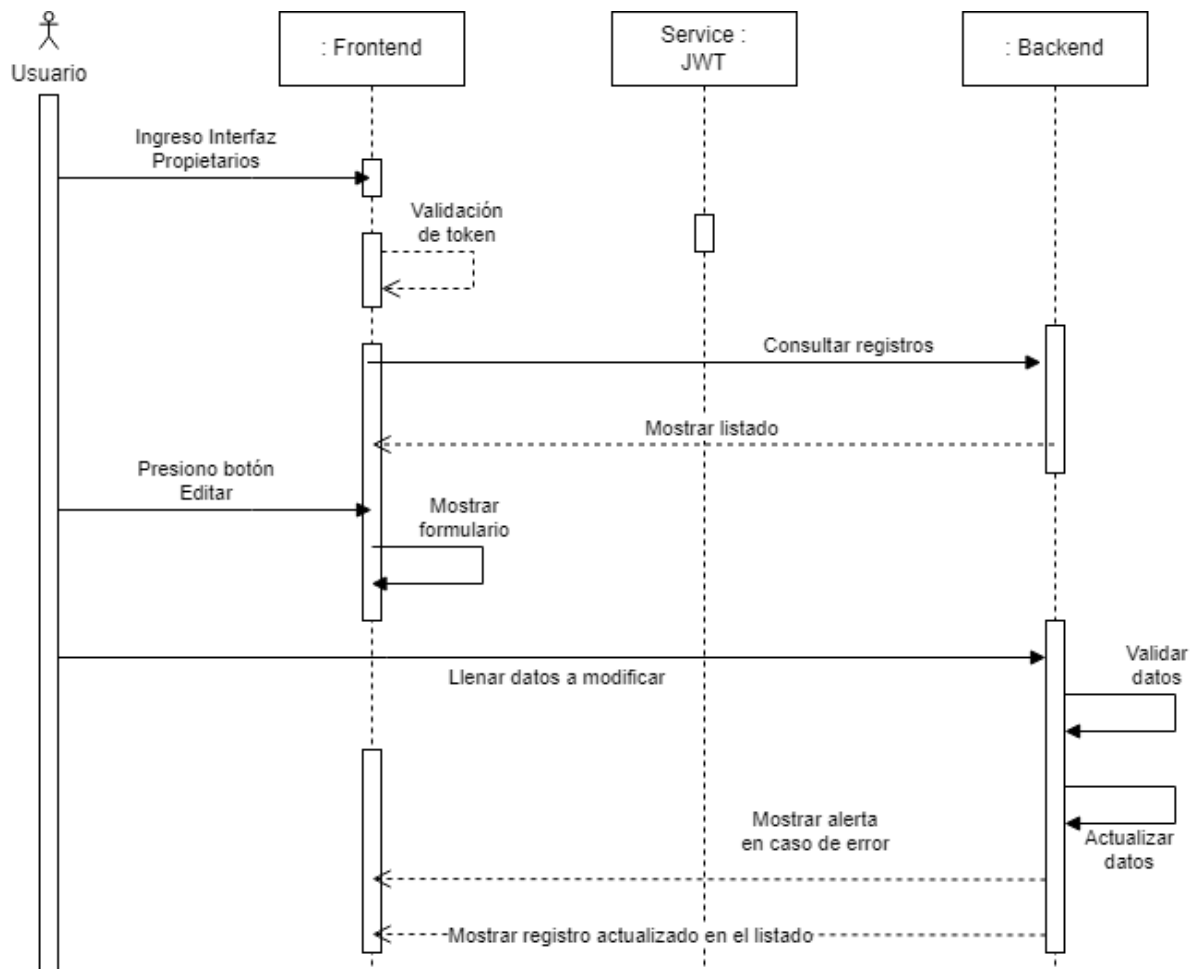


Figura 101: Diagrama de secuencia HUI6 – Actualizar datos de propietarios

Fuente: Elaboración propia

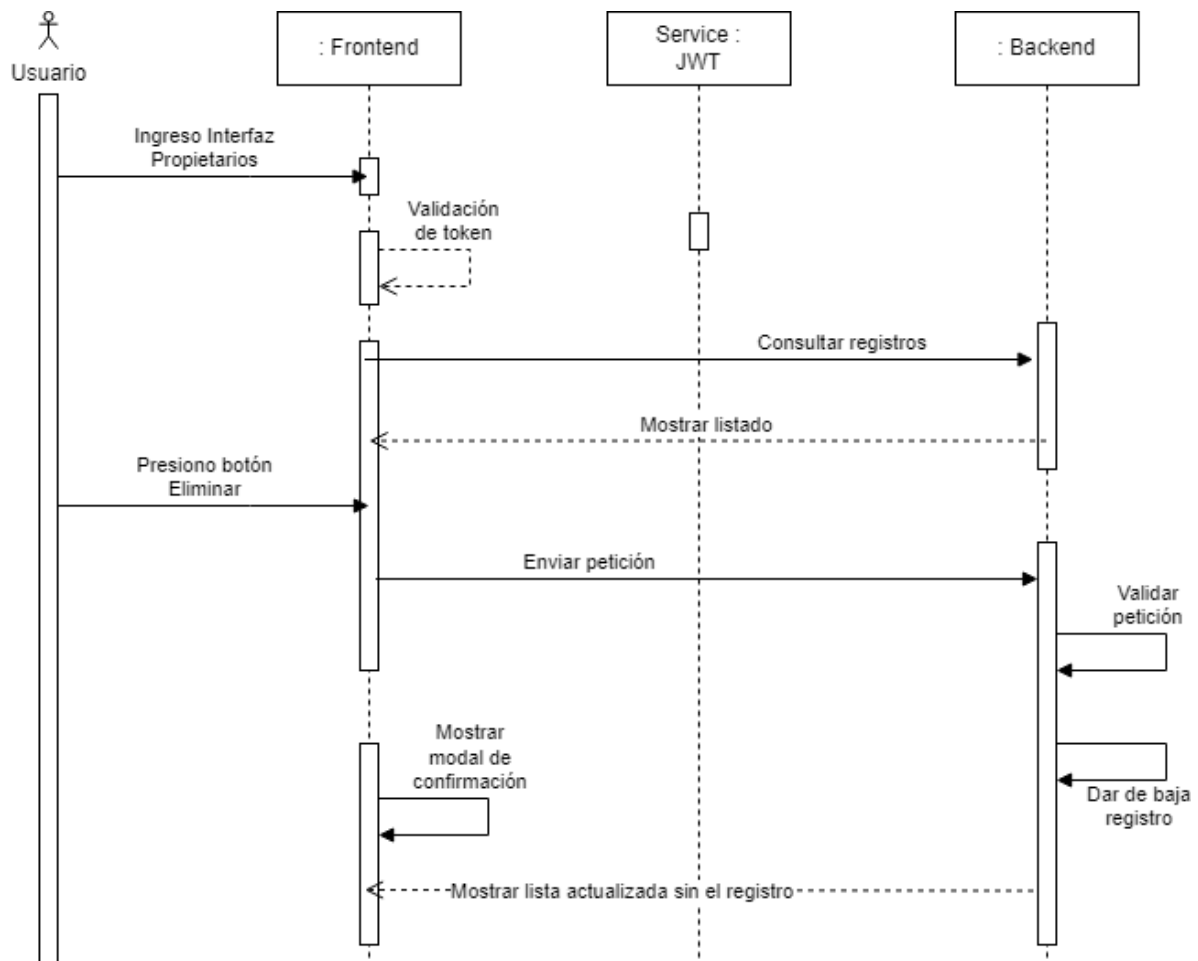


Figura 102: Diagrama de secuencia HU17 – Dar de baja propietario

Fuente: Elaboración propia

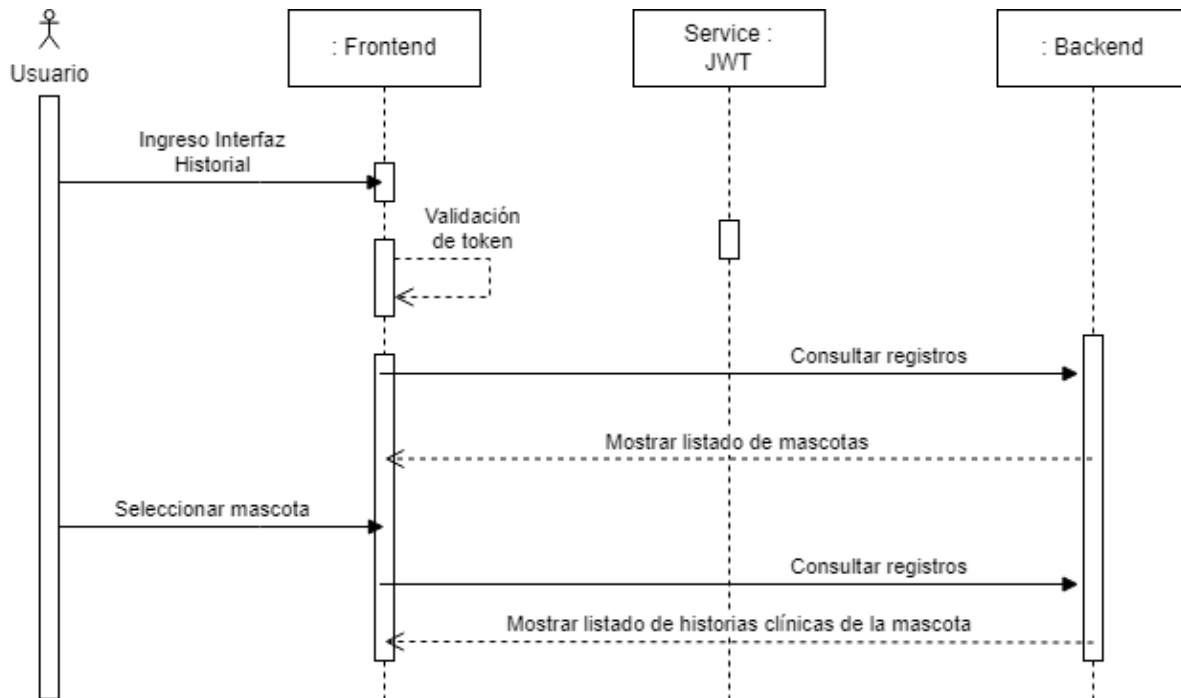


Figura 103: Diagrama de secuencia HU18 – Leer historiales clínicos

Fuente: Elaboración propia

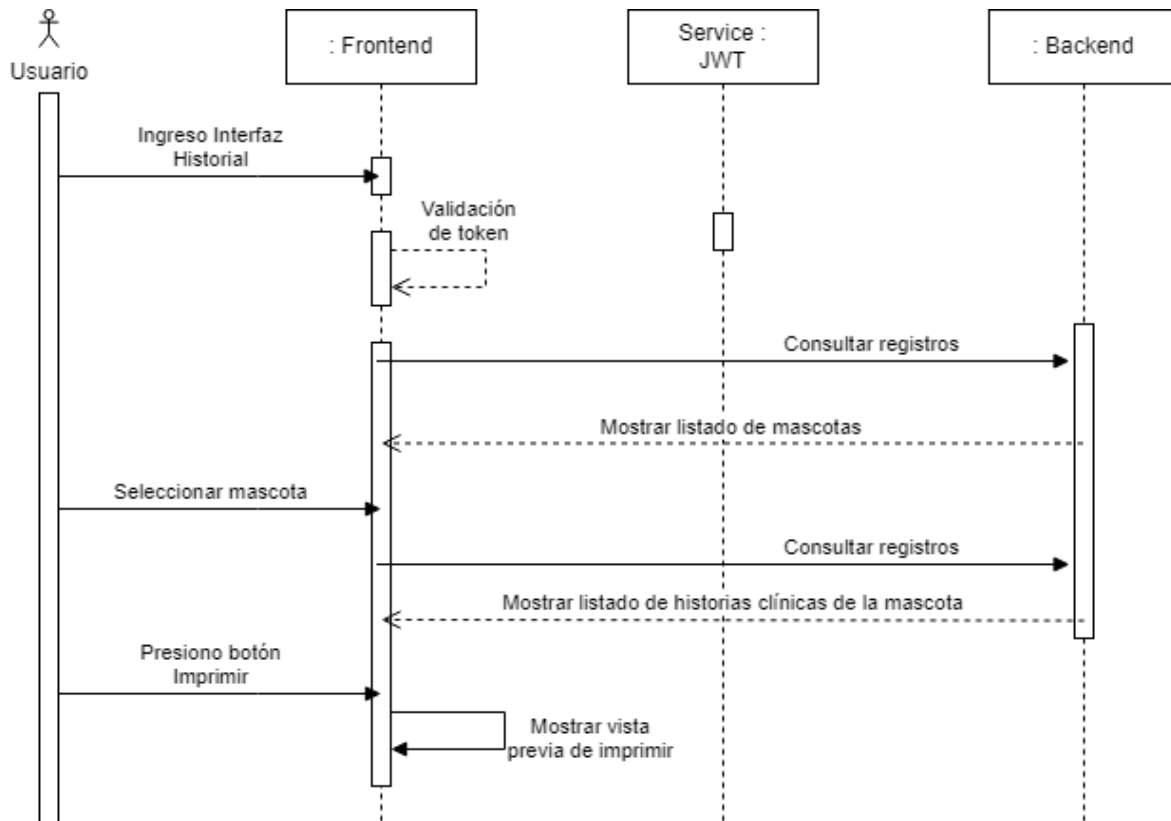


Figura 104: Diagrama de secuencia HU19 – Reporte historia clínica

Fuente: Elaboración propia

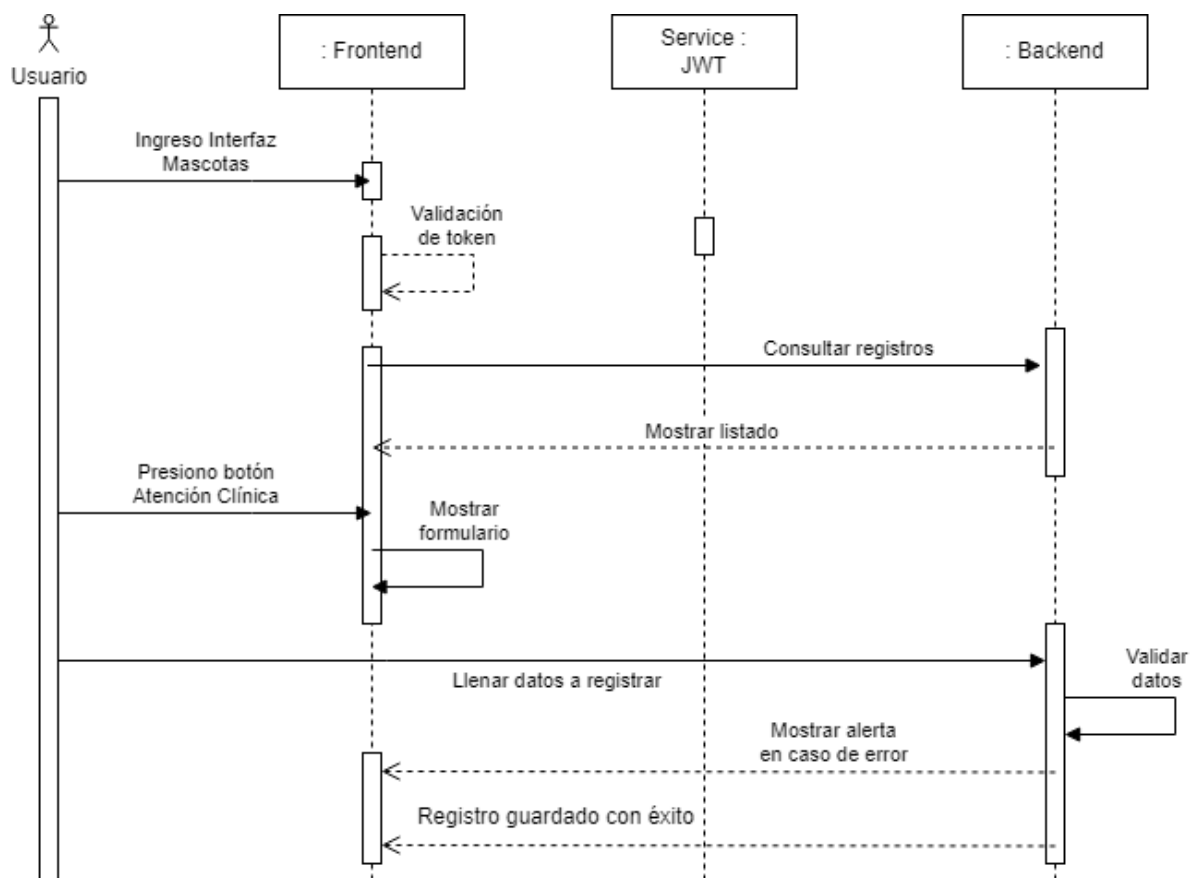


Figura 105: Diagrama de secuencia HU20 – Registrar atención clínica

Fuente: Elaboración propia

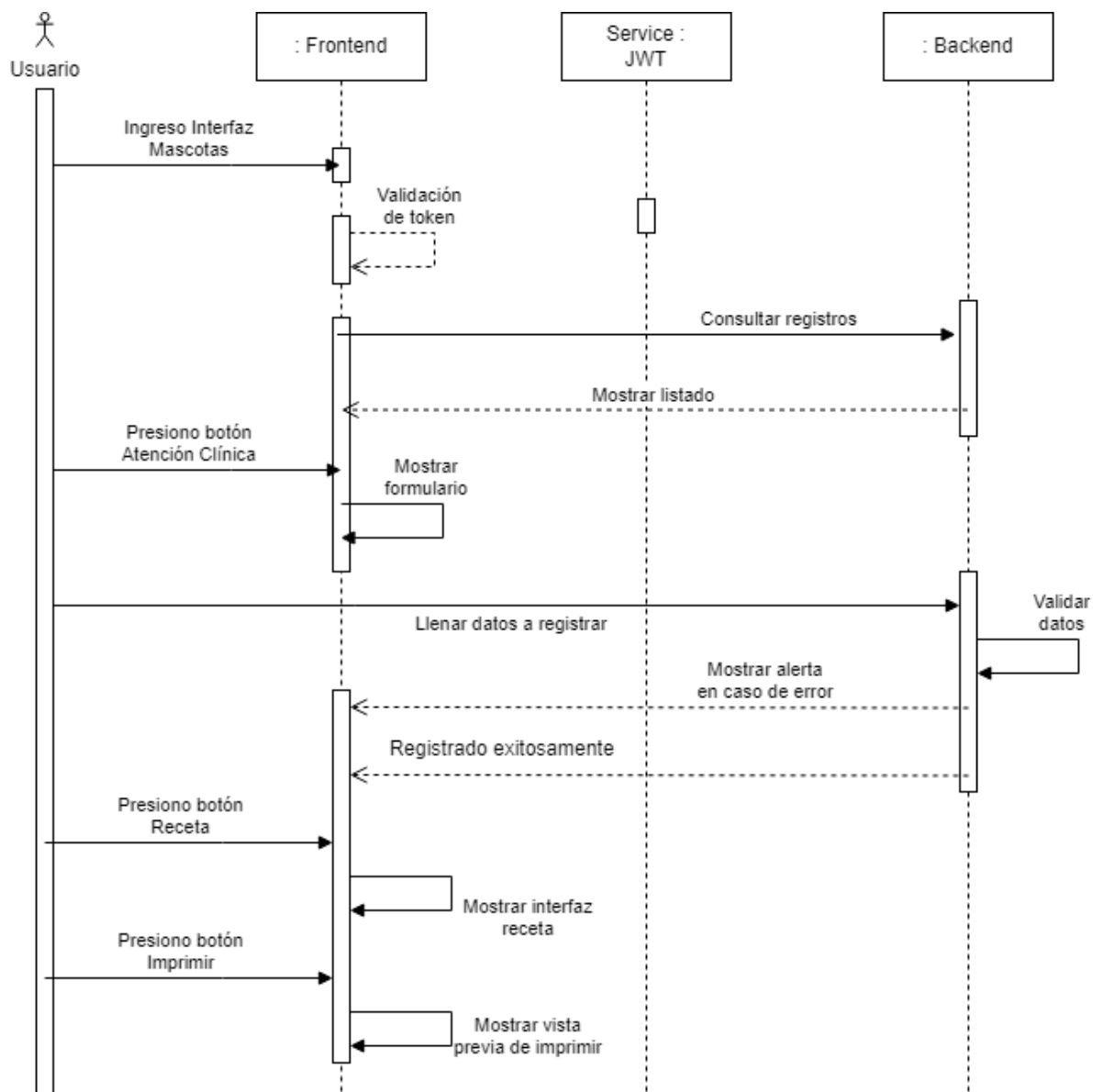


Figura 106: Diagrama de secuencia HU21 – Generar receta

Fuente: Elaboración propia

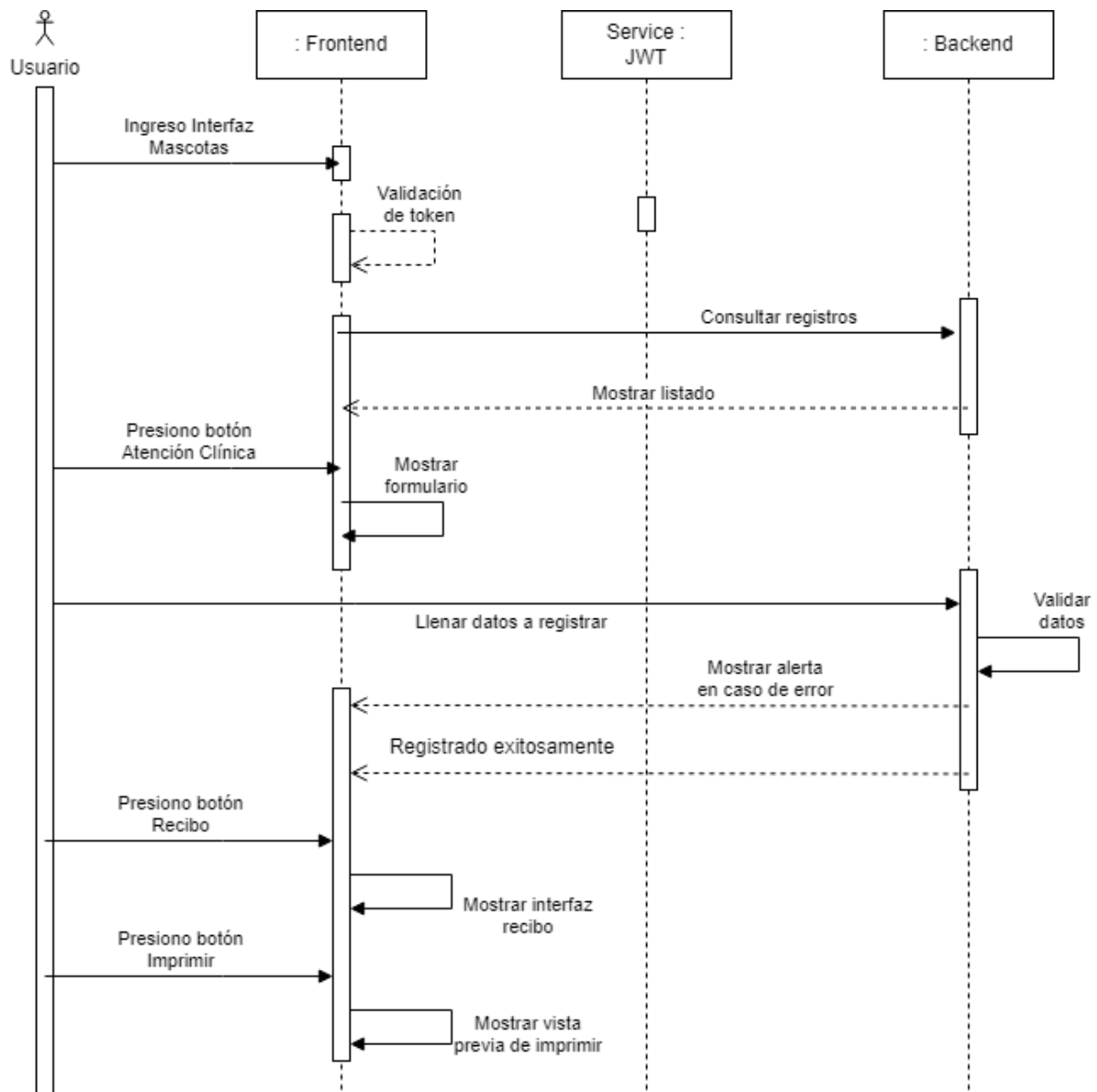


Figura 107: Diagrama de secuencia HU22 – Generar recibo

Fuente: Elaboración propia

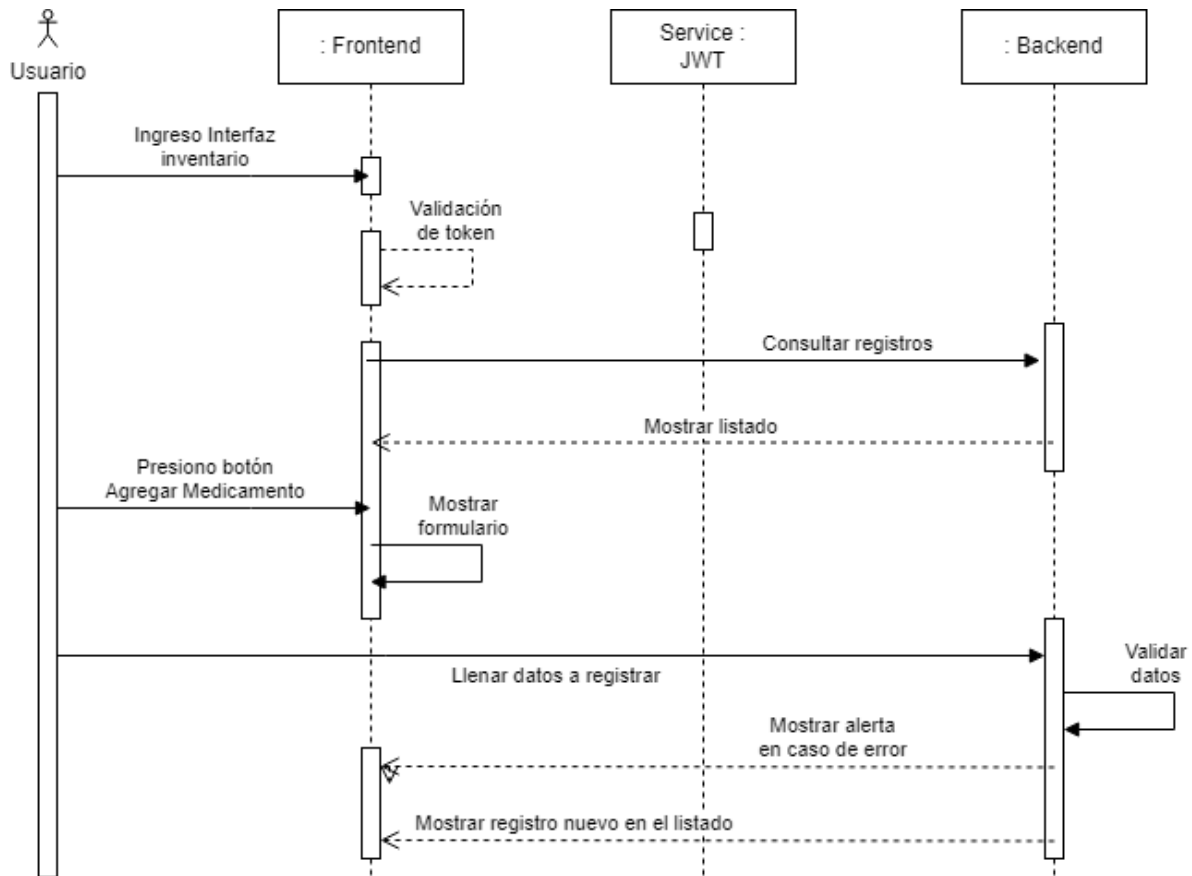


Figura 108: Diagrama de secuencia HU23 - Registrar medicamentos

Fuente: Elaboración propia

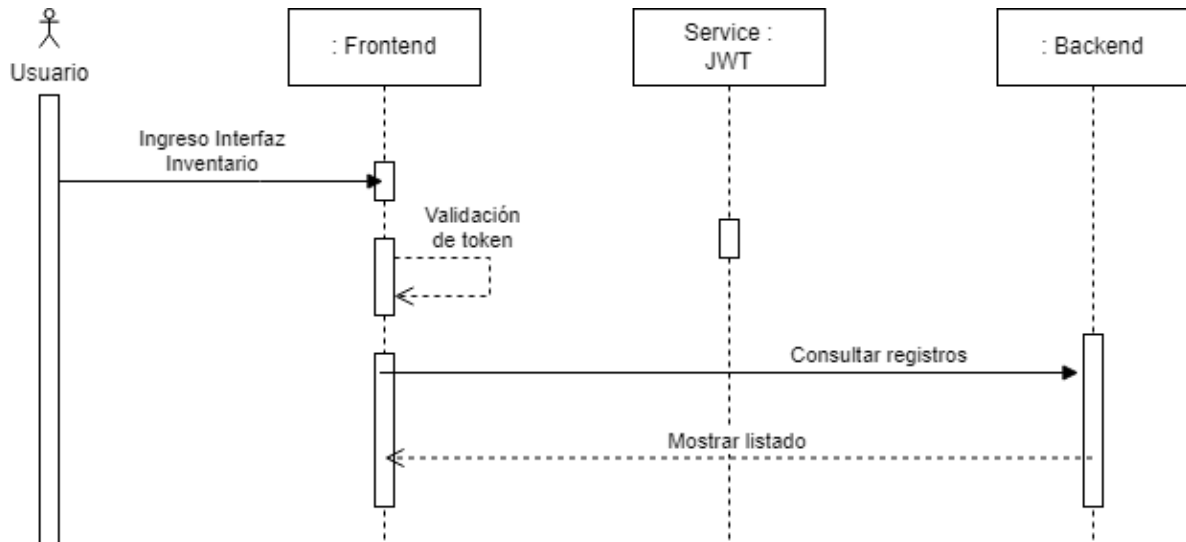


Figura 109: Diagrama de secuencia HU24 - Leer medicamentos registrados

Fuente: Elaboración propia

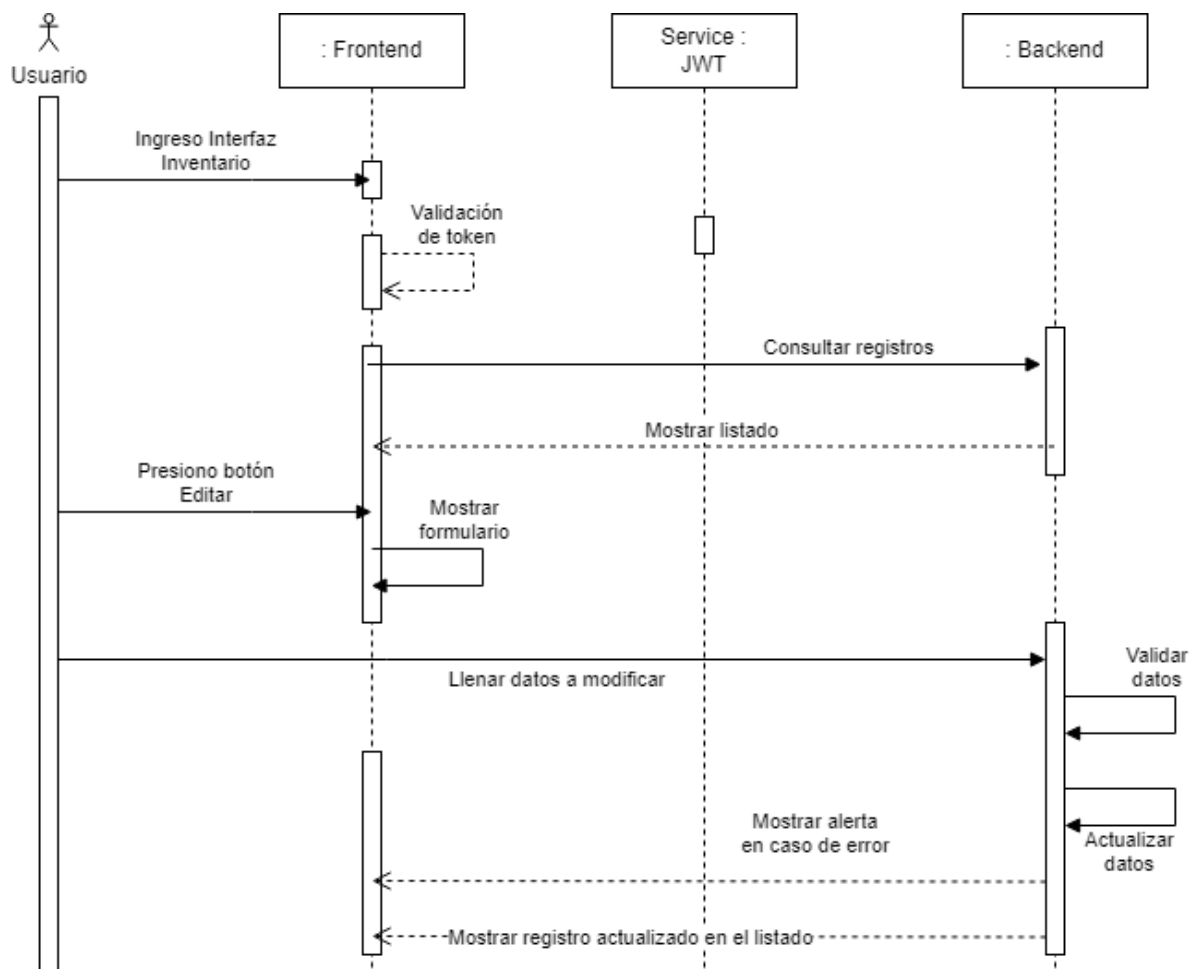


Figura 110: Diagrama de secuencia HU25 - Actualizar datos de medicamentos

Fuente: Elaboración propia

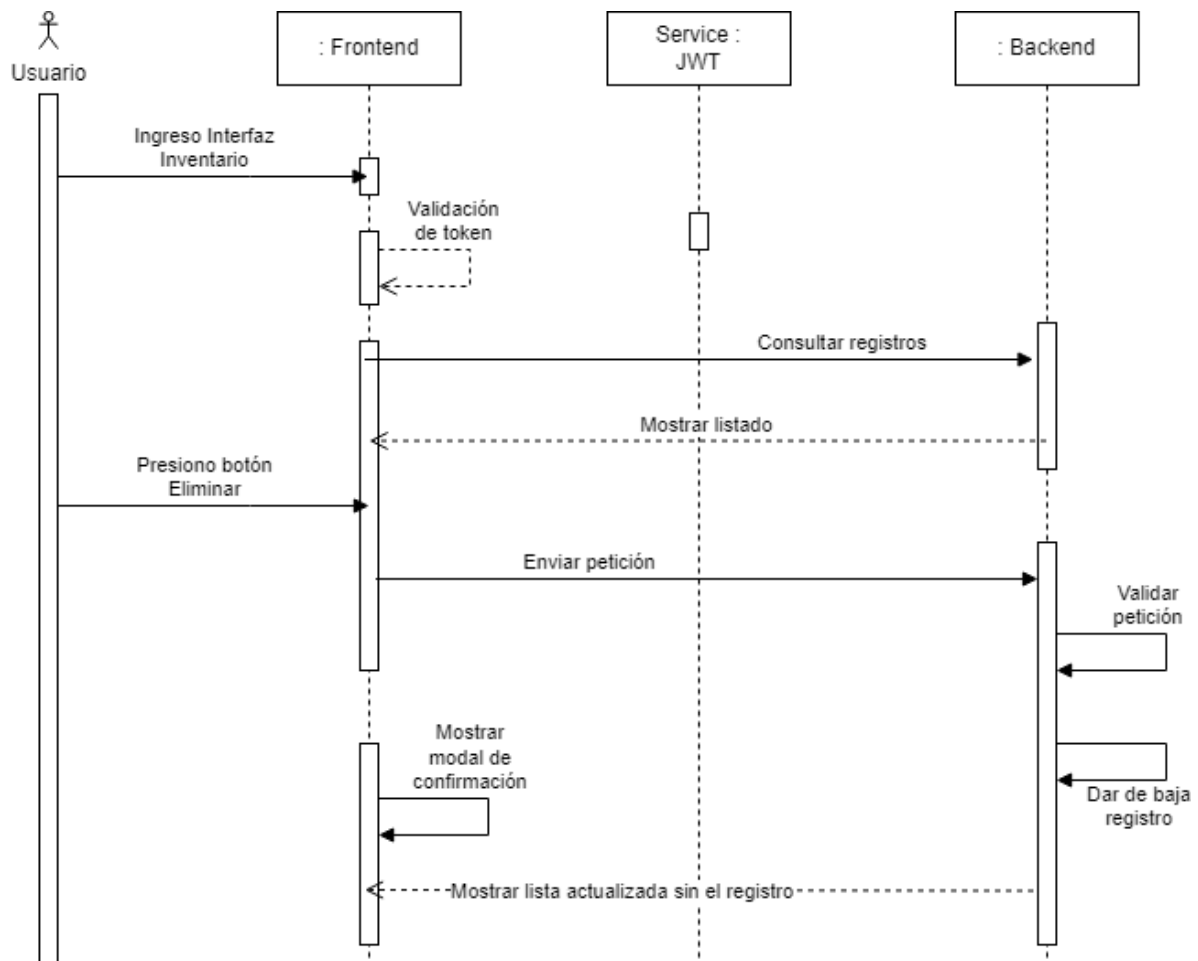


Figura 111: Diagrama de secuencia HU26 - Eliminar medicamento

Fuente: Elaboración propia

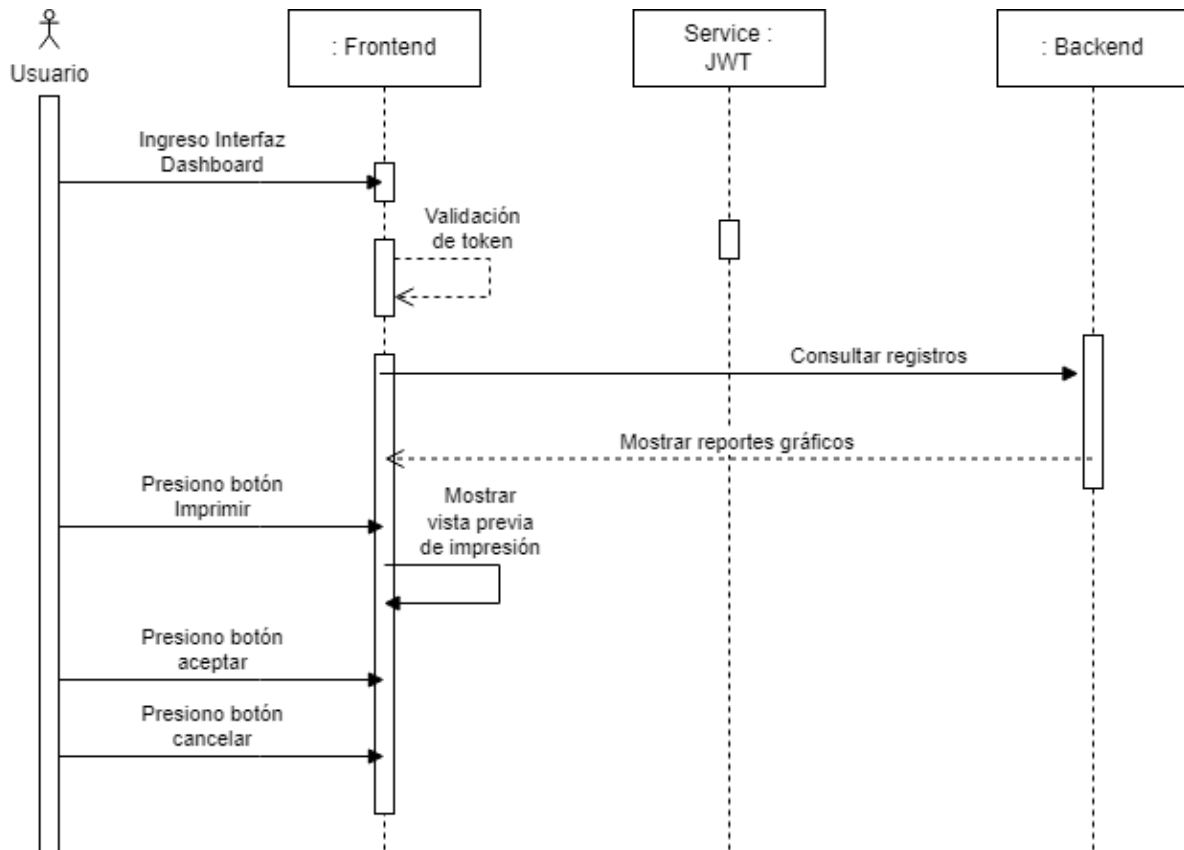


Figura 112: Diagrama de secuencia HU27 – Dashboard

Fuente: Elaboración propia

II.1.2.2.3.4 Interfaces

A continuación, se muestran las interfaces de las historias de usuario para este Sprint 3:

HU07 – Registrar Mascotas

En la siguiente imagen se muestra el formulario de registro de mascota:

VETERINARIA MALAGA

Mascotas

AGREGAR MASCOTA

Nombre: Ej: Coco

Sexo: Hembra Macho

Fecha de Nacimiento: dd/mm/aaaa

Raza: Ej: Chapi

Color: Ej: Blanco y Negro

Aceptar

© 2023 By Ross

DATOS DEL PROPIETARIO
 Nombre: Cynthia Williams Carson
 Celular:+1-524-966-1093x523

LogOut

Figura 113: Interfaz HU07 – Registrar Mascota

Fuente: Elaboración propia

HU08 – Leer Mascotas Registrados

En la siguiente imagen se muestra la interfaz principal de mascotas, un listado en tarjetas de todas las mascotas registradas de acuerdo al propietario seleccionado:

VETERINARIA MALAGA

Mascotas

AGREGAR MASCOTA
Atención

ROCKY - 0 AÑOS
Especie: Canino
Alergias: Ácaros ambient...

REX - 0 AÑOS
Especie: Canino
Alergias: Dermatitis atópi...

NALA - 0 AÑOS
Especie: Felino
Alergias: Alimentos. Vacu...

LILLY - 2 AÑOS
Especie: Felino
Alergias: acaros

DATOS DEL PROPIETARIO
 Nombre: Cynthia Williams Carson
 Celular:+1-524-966-1093x523

Dirección: 69602 Brown Squares Apt. 787 North Troyport, MP 61697

LogOut

Figura 114: Interfaz HU08 – Leer mascotas registradas

Fuente: Elaboración propia

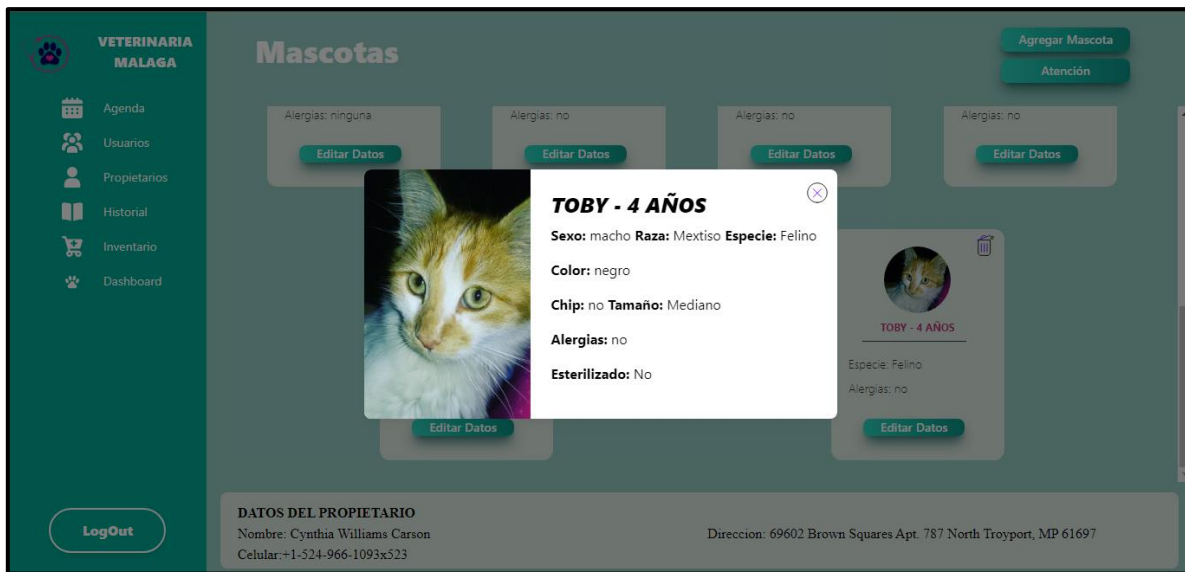


Figura 115: Interfaz HU08 – Ver información de mascotas registradas

Fuente: Elaboración propia

HU09 – Actualizar datos de Mascotas

En la siguiente imagen se muestra el formulario para editar una mascota:

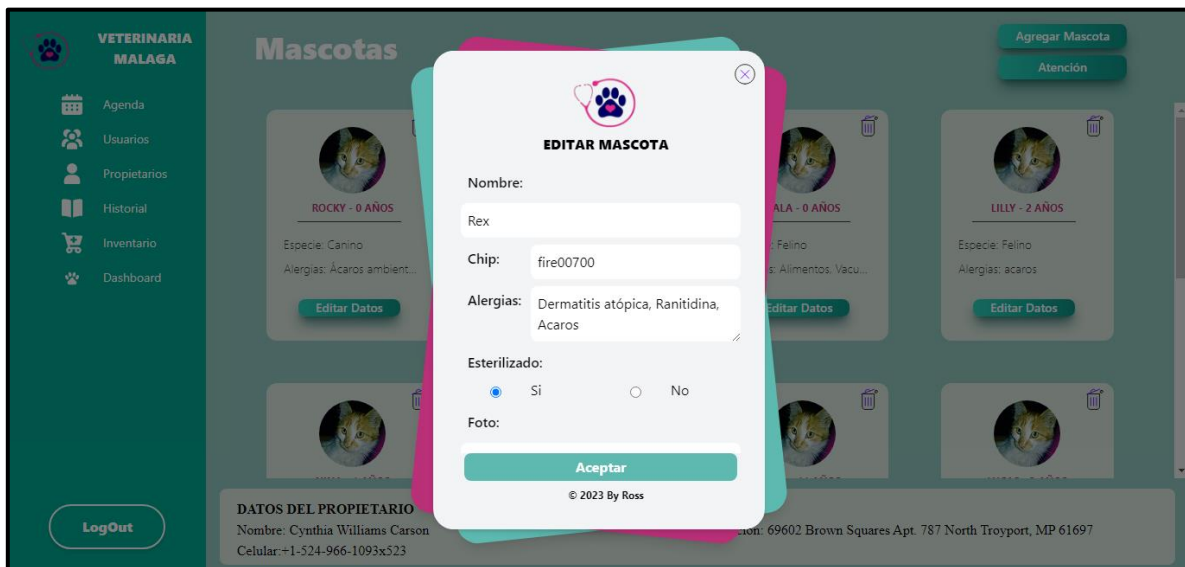


Figura 116: Interfaz HU08 – Actualizar datos de mascotas

Fuente: Elaboración propia

HU10 – Dar de Baja Mascota

En la siguiente imagen se muestra la validación para eliminar una mascota:

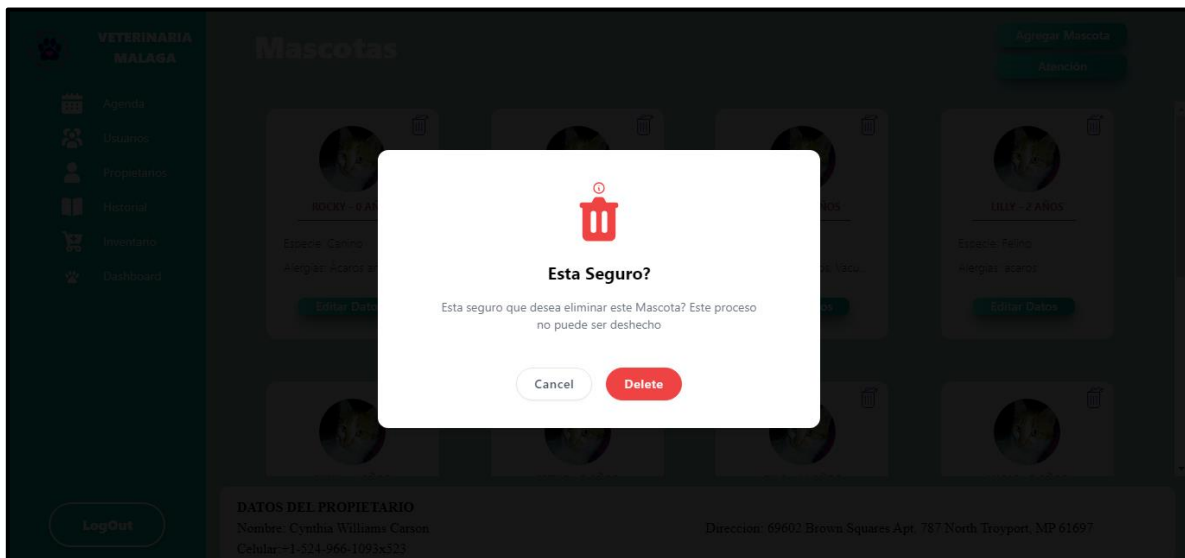


Figura 117: Interfaz HU10 – Dar de Baja Mascota

Fuente: Elaboración propia

HU11 – Registrar Cita

En la siguiente imagen se muestra el formulario de registro de citas en la agenda:

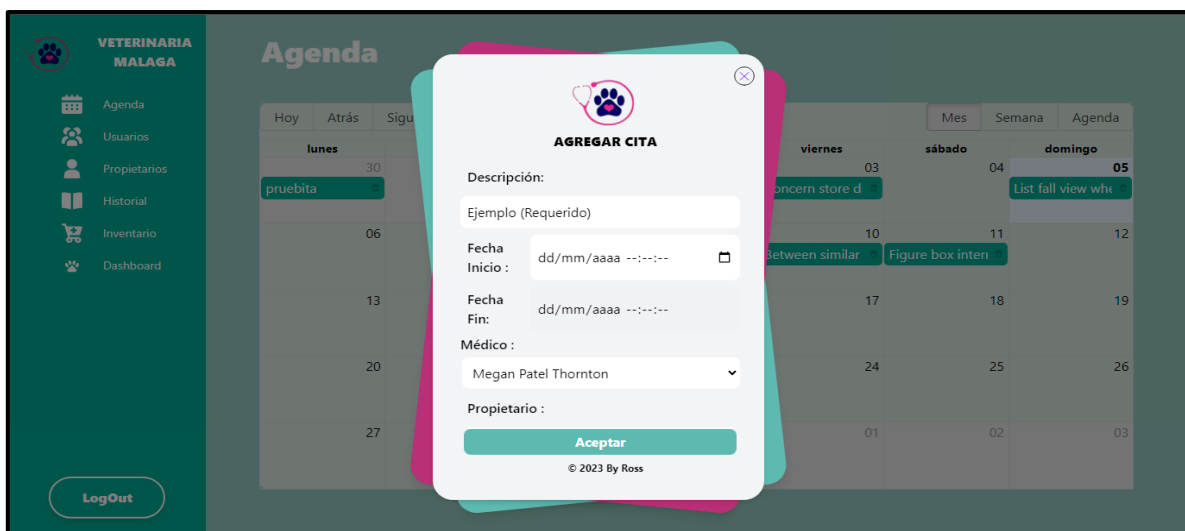


Figura 118: Interfaz HU11 – Registrar Cita

Fuente: Elaboración propia

HU12 – Leer Citas Registradas

En la siguiente imagen se muestra la interfaz principal de agenda, un calendario mensual donde se encuentran todas las citas registradas de acuerdo al mes seleccionado:



Figura 119: Interfaz HU12 – Leer citas registradas

Fuente: Elaboración propia

HU13 – Cancelar cita

En la siguiente imagen se muestra la validación para cancelar una cita:

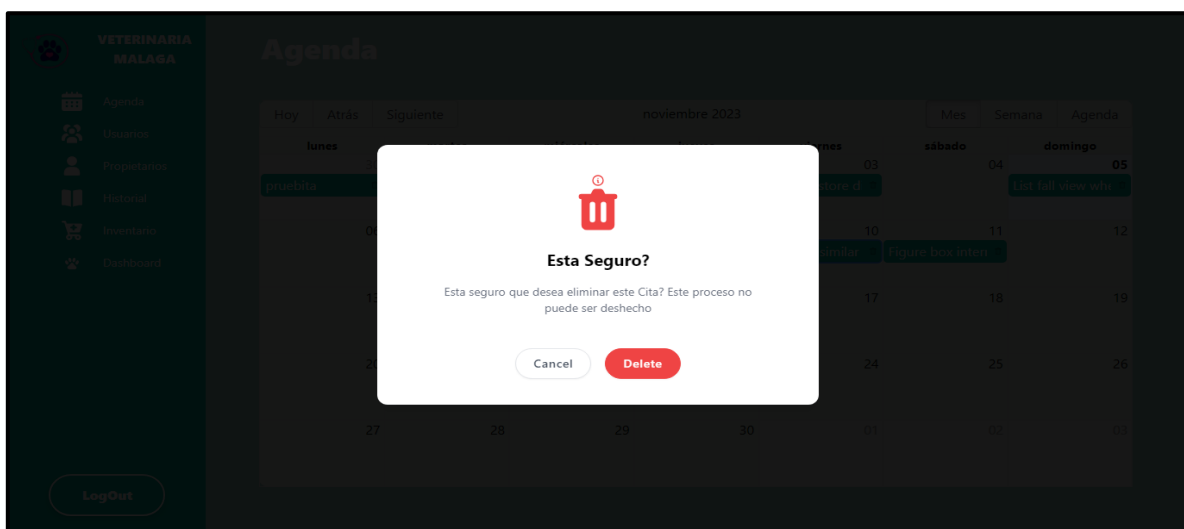


Figura 120: Interfaz HU13 – Cancelar Cita

Fuente: Elaboración propia

HU14 – Registrar Propietario

En la siguiente imagen se muestra el formulario de registro de propietarios:

Figura 121: Interfaz HU14 – Registrar Propietario

Fuente: Elaboración propia

HU15 – Leer Propietarios Registrados

En la siguiente imagen se muestra la interfaz principal de propietarios, un listado de todos los propietarios registrados:

Nombre	Apellido Paterno	Apellido Materno	Celular	Dirección	Acciones	Mascotas
Cynthia	Williams	Carson	+1-524-966-1093x523	69602 Brown Squares Apt. 787 North Troyport, MP 61697	🔍 🗑️	Ver
Terri	Anderson	Meyer	(338)512-0665x03008	4217 Burton Brooks Robinville, VI 34119	🔍 🗑️	Ver
Debra	Morgan	Oliver	(212)940-0034x855	658 Jared Radial Suite 022 Mathewstad, VT 12783	🔍 🗑️	Ver
Blake	Wells	Randall	(329)945-6824	2814 Houston Hills Rodriguezside, PW 63569	🔍 🗑️	Ver
Theresa	Richards	Jackson	317-317-6045x22961	60168 West Overpass South Christopher, IN 34699	🔍 🗑️	Ver
James	Elliott	Little	610-587-3176	213 Gray Plains North Vanessa, NM 19952	🔍 🗑️	Ver
Rodney	Rivera	Martin	475.677.3893x05550	9471 Zachary Canyon East Davidport, AL 59298	🔍 🗑️	Ver
Eduardo	Hahn	Turner	+1-875-586-8809x1891	6993 Diane Alley Apt. 489 Marvinfort, VI 34842	🔍 🗑️	Ver
Ricky	Banks	Lee	(302)423-4500x7627	0976 Williams Mountains Apt. 009 Lake Mario, UT 72478	🔍 🗑️	Ver
Krista	Chan	Hale	001-371-509-7951x94264	5375 Sue Canyon Suite 408 East Clayton, GU 82739	🔍 🗑️	Ver

Figura 122: Interfaz HU15 – Leer Propietarios Registrados

Fuente: Elaboración propia

HU16 – Actualizar datos de Propietarios

En la siguiente imagen se muestra el formulario para editar un propietario:

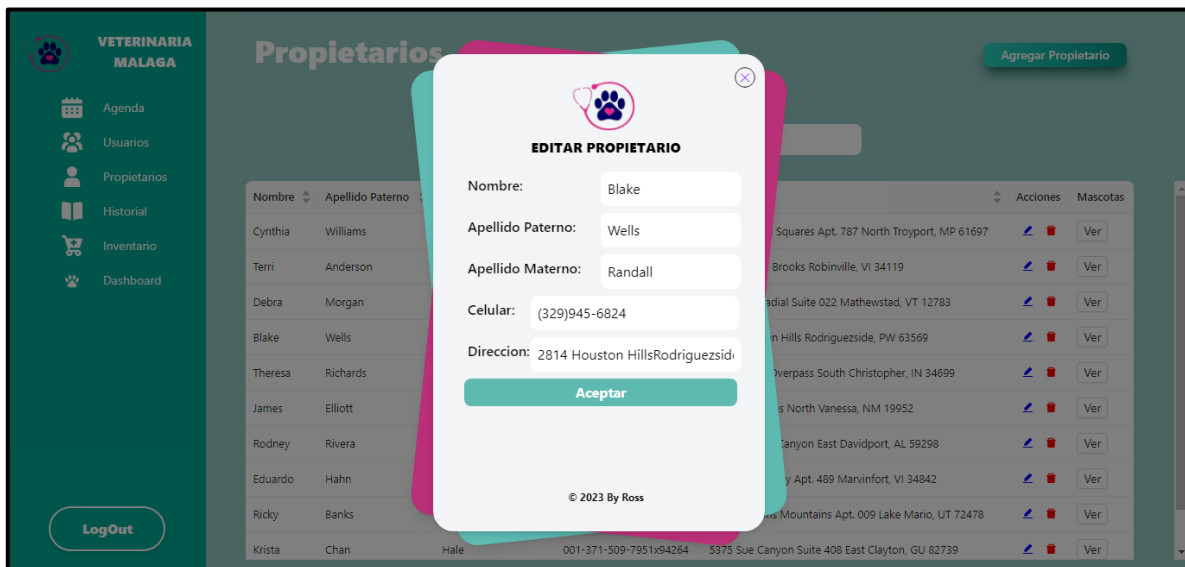


Figura 123: Interfaz HU17 – Actualizar datos de Propietarios

Fuente: Elaboración propia

HU17 – Dar de Baja Propietario

En la siguiente imagen se muestra la validación para eliminar un propietario:

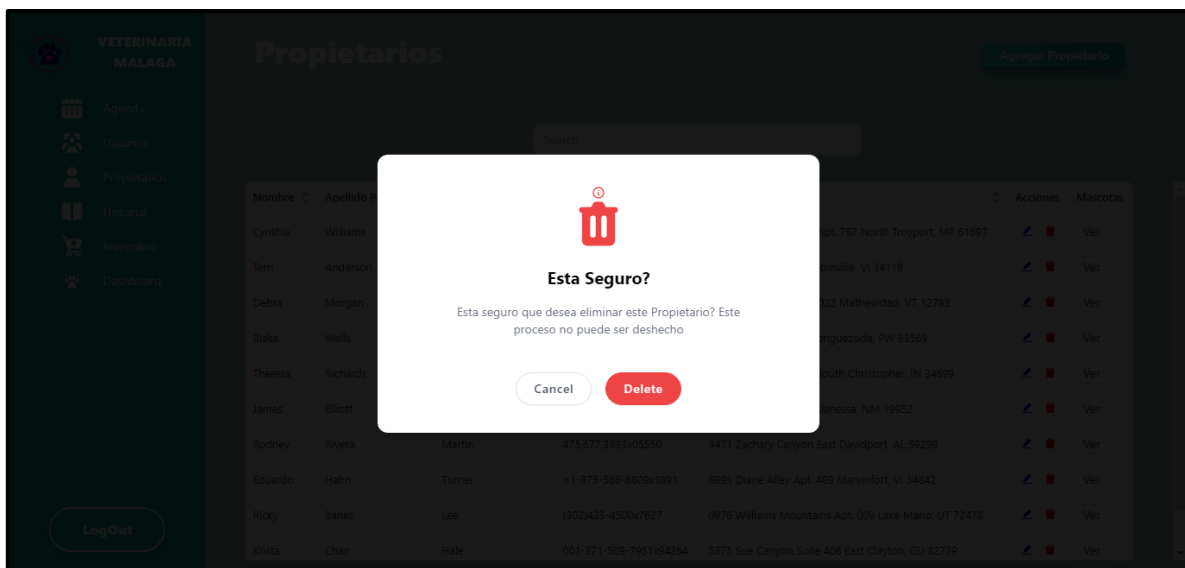
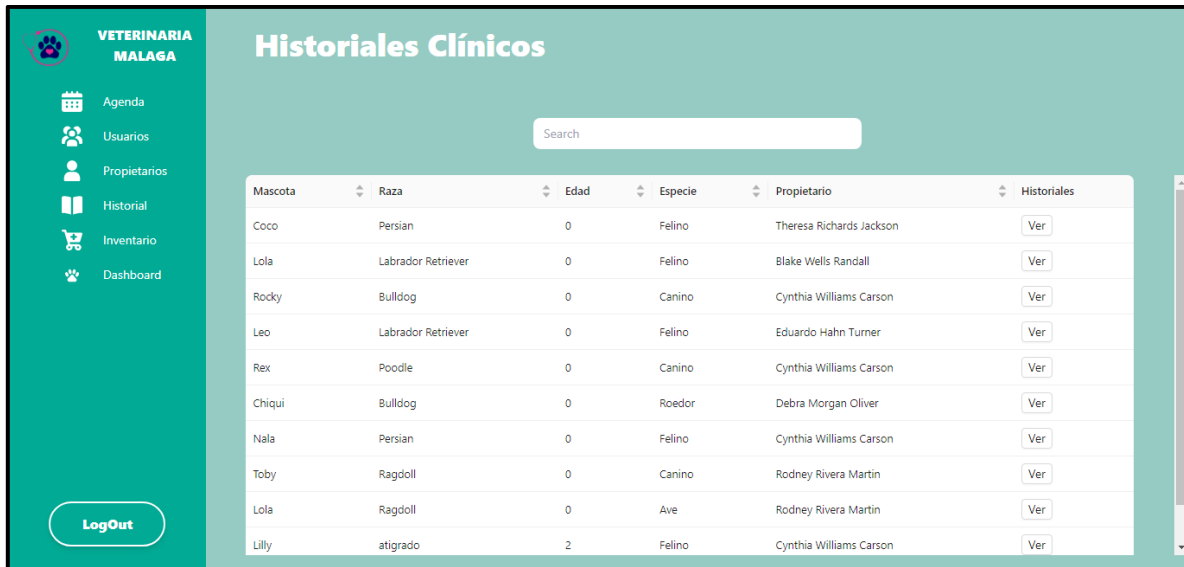


Figura 124: Interfaz HU17 – Dar de Baja Propietario

Fuente: Elaboración propia

HU18 – Leer Historiales Clínicos

En la siguiente imagen se muestra la interfaz principal de historiales clínicos, un listado de todas las mascotas con sus respectivos historiales clínicos:



The screenshot shows the 'Historiales Clínicos' interface. On the left is a sidebar with navigation options: Agenda, Usuarios, Propietarios, Historial, Inventario, and Dashboard. The main area has a search bar and a table of pets with their medical records. The table columns are Mascota, Raza, Edad, Especie, Propietario, and Historiales. Each row has a 'Ver' button to view the pet's history.

Mascota	Raza	Edad	Especie	Propietario	Historiales
Coco	Persian	0	Felino	Theresa Richards Jackson	Ver
Lola	Labrador Retriever	0	Felino	Blake Wells Randall	Ver
Rocky	Bulldog	0	Canino	Cynthia Williams Carson	Ver
Leo	Labrador Retriever	0	Felino	Eduardo Hahn Turner	Ver
Rex	Poodle	0	Canino	Cynthia Williams Carson	Ver
Chiqui	Bulldog	0	Roedor	Debra Morgan Oliver	Ver
Nala	Persian	0	Felino	Cynthia Williams Carson	Ver
Toby	Ragdoll	0	Canino	Rodney Rivera Martin	Ver
Lola	Ragdoll	0	Ave	Rodney Rivera Martin	Ver
Lilly	atigrado	2	Felino	Cynthia Williams Carson	Ver

Figura 125: Interfaz HU18 – Leer Historiales Clínicos Interfaz Principal

Fuente: Elaboración propia



The screenshot shows the 'Historial Clínico de Coco' interface. It displays detailed medical records for the pet 'Coco'. The top section shows basic information: EDAD: 0, ESPECIE: Felino, SEXO: macho, RAZA: Persian, COLOR: Ivory con manchas Maroon, ESTERILIZADO: Si, CHIP: movie, ALERGIAS: Alimentos, Vacuna contra la Enfermedad de Lyme. Below this is a search bar and a table of medical records with columns for Fecha and Detalle. Each record includes a date, time, diagnosis, services provided, prescribed medications, and vital signs like temperature, weight, heart rate, and respiratory rate.

Fecha	Detalle
2023-12-06 05:16:15	<p>Diagnóstico: Set discussion seven evidence worker building this American either.</p> <p>Servicios Prestados: Consulta, Cirugía</p> <p>Medicamentos Prescritos: Omeprazol, Tableta de Desparasitación</p> <p>Temperatura: 41.834636340051716</p> <p>Peso: 32,24189146647173</p> <p>Frecuencia Cardíaca: 173</p> <p>Frecuencia Respiratoria: 44</p>
2023-10-26 16:02:16	<p>Diagnóstico: Bien</p> <p>Servicios Prestados: Consulta, Desparasitación</p> <p>Medicamentos Prescritos: Diclofenac, Amoxicilina</p> <p>Temperatura: 27</p> <p>Peso: 6</p> <p>Frecuencia Cardíaca: 40</p> <p>Frecuencia Respiratoria: 60</p>
2023-10-26 16:02:16	<p>Diagnóstico: Bien</p> <p>Servicios Prestados: Consulta, Desparasitación</p> <p>Medicamentos Prescritos: Diclofenac, Amoxicilina</p> <p>Temperatura: 27</p> <p>Peso: 6</p> <p>Frecuencia Cardíaca: 40</p> <p>Frecuencia Respiratoria: 60</p>

Figura 126: Interfaz HU18 – Leer Historiales Clínicos Interfaz por Mascota

Fuente: Elaboración propia

HU19 – Reporte Historia Clínica

En la siguiente imagen se muestra la ficha clínica con los datos de la atención seleccionada:

FICHA CLÍNICA

Fecha: 2023-10-11

Ficha de Identificación

Nombre: Lola	Edad: 0 años	Sexo: macho	Raza: Labrador Retriever
Especie: Felino	Color: GhostWhite con manchas LightSteelBlue	Esterilizado: No	

Alergias: Alimentos, Vacuna contra la Enfermedad de Lyme

Servicios
Consulta, Desparasitación

Detalles de la Atención

Temperatura: 36.20 grados	Peso: 15.20 kilogramos
Frecuencia Cardíaca: 40 LPM	Frecuencia Respiratoria: 40 RPM

Diagnóstico
enfermo

Medicación

CANCELAR
IMPRIMIR

Figura 127: Interfaz HU19 - Reporte historia clínica

Fuente: Elaboración propia

FICHA CLÍNICA

Fecha: 2023-12-05

Ficha de Identificación

Nombre: Chiqui	Edad: 0 años	Sexo: hembra	Raza: Bulldog
Especie: Roedor	Color: LightBlue con manchas Red	Esterilizado: SI	

Alergias: Alimentos, Jarabe para la Tos

Servicios
Consulta, Tratamiento

Detalles de la Atención

Temperatura: 29 grados	Peso: 5 kilogramos
Frecuencia Cardíaca: 45 LPM	Frecuencia Respiratoria: 50 RPM

Diagnóstico
Infeccion

Medicación
Diclofenac, Amoxicilina

Imprimir 1 página

Destino Guardar como PDF ▼

Páginas Todo ▼

Diseño Retrato ▼

Más ajustes ▼

Figura 128: Reporte historia clínica imprimir

Fuente: Elaboración propia

HU20 – Registrar Atención Clínica

En la siguiente imagen se muestra el formulario de registro de atención clínica:

Figura 129: Interfaz HU20 – Registrar Atención Clínica

Fuente: Elaboración propia

HU21 – Generar Receta

En la siguiente imagen se muestra la receta generada al finalizar el registro de una atención clínica:

Figura 130: Interfaz HU21 – Generar Receta

Fuente: Elaboración propia

HU22 – Generar Recibo

En la siguiente imagen se muestra el recibo generado al finalizar el registro de una atención clínica:

ATENCIÓN CLÍNICA

TEMPERATURA °C

DIAGNÓSTICO

SERVICIOS:

- Consulta
- Vacunación
- Desparasitación
- Cirugía
- Tratamiento

MEDICAMENTOS:

Diclofenac x Amoxicilina x Prednisona x VIA DE ADMINISTRACIÓN

horas hasta terminar.
Prednisona media tableta una vez al día

VETERINARIA MÁLAGA RECIBO

Fecha: 11/5/2023

Dirigido a: Debra Morgan Oliver

Nombre	Precio	Cantidad	Precio Parcial
Diclofenac	27.53	1	27.53
Amoxicilina	16.45	2	32.90
Prednisona	10.51	1	10.51

Recibo Total: 70.94

Cancelar IMPRIMIR

Aceptar

Figura 131: Interfaz HU22 – Generar Recibo

Fuente: Elaboración propia

HU23 – Registrar Medicamentos

En la siguiente imagen se muestra el formulario de registro de medicamentos:

VETERINARIA MÁLAGA

Inventario

Agenda Usuarios Propietarios Historial Inventario Dashboard

LogOut

AGREGAR MEDICAMENTO

Nombre: Ej: Amoxicilina

Cantidad: Ej: 10

Precio: Ej: 55.55

Tipo de Medicamento: Inyectable

Aceptar

© 2023 By Ross

Nombre	Precio	Acciones
Diclofenac	27.53	[+]
Amoxicilina	16.45	[+]
Ivermectina	11.37	[+]
Ranitidina	10.28	[+]
Metoprolol	38.1	[+]
Enrofloxacina	24.72	[+]
Prednisona	10.51	[+]
Dexametasona	23.13	[+]
Cefalexina	14	[+]
Omeprazol	35.96	[+]

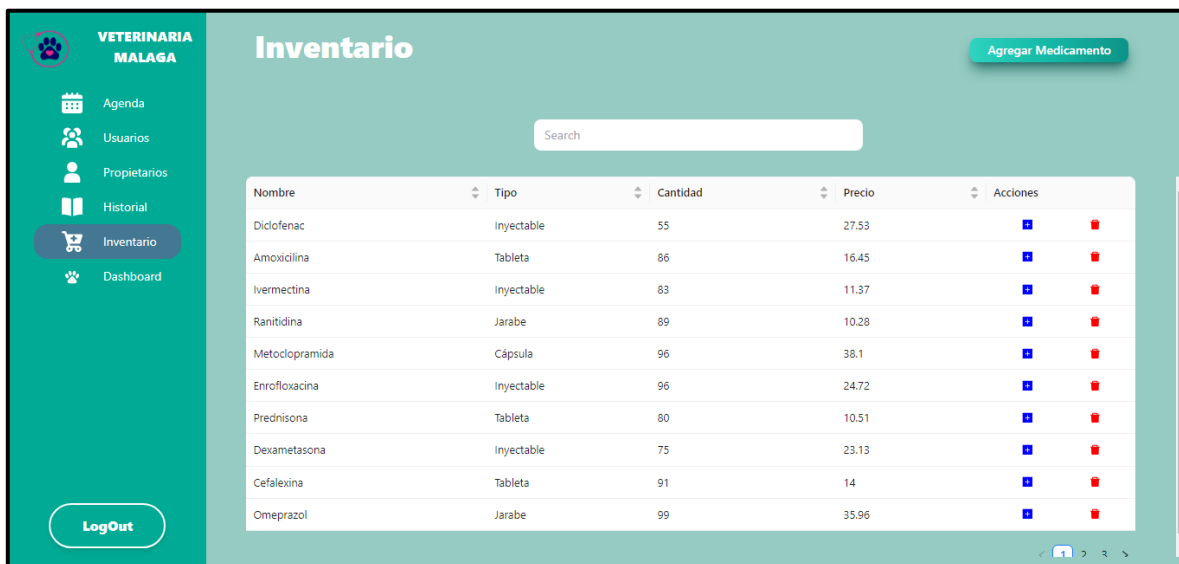
Agregar Medicamento

Figura 132: Interfaz HU23 – Registrar medicamentos

Fuente: Elaboración propia

HU24 – Leer Medicamentos Registrados

En la siguiente imagen se muestra la interfaz principal de medicamentos, un listado de todos los medicamentos registrados:



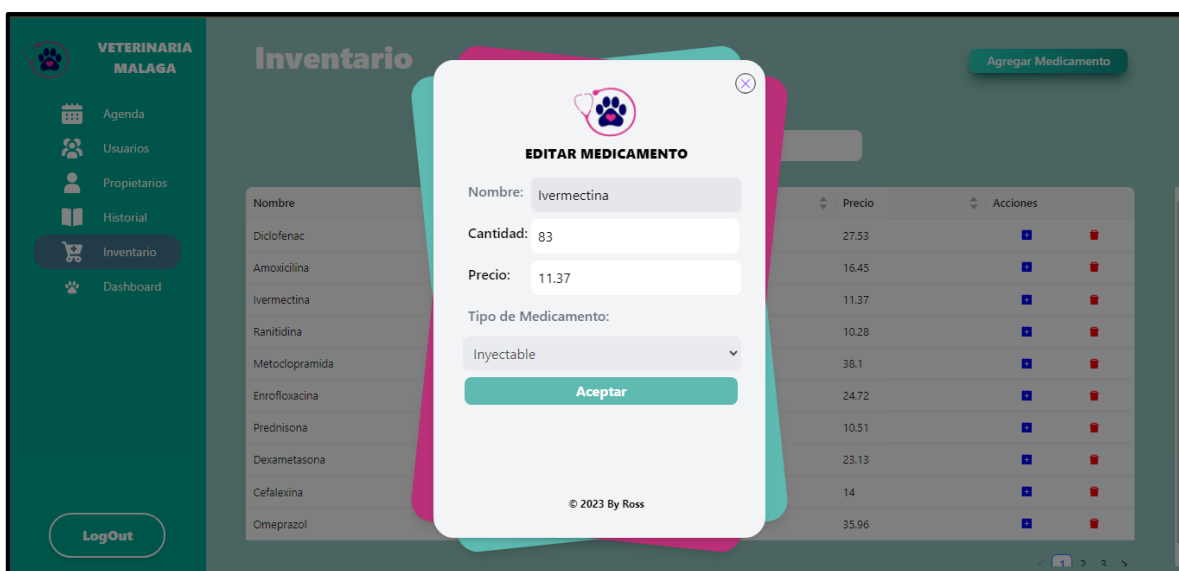
Nombre	Tipo	Cantidad	Precio	Acciones
Diclofenac	Inyectable	55	27.53	[+][x]
Amoxicilina	Tableta	86	16.45	[+][x]
Ivermectina	Inyectable	83	11.37	[+][x]
Ranitidina	Jarabe	89	10.28	[+][x]
Metoclopramida	Cápsula	96	38.1	[+][x]
Enrofloxacin	Inyectable	96	24.72	[+][x]
Prednisona	Tableta	80	10.51	[+][x]
Dexametasona	Inyectable	75	23.13	[+][x]
Cefalexina	Tableta	91	14	[+][x]
Omeprazol	Jarabe	99	35.96	[+][x]


Figura 133: Interfaz HU24 – Leer medicamentos registrados

Fuente: Elaboración propia

HU25 – Actualizar datos de Medicamentos

En la siguiente imagen se muestra el formulario para editar un medicamento:





EDITAR MEDICAMENTO

Nombre:

Cantidad:

Precio:

Tipo de Medicamento:

Aceptar

© 2023 By Ross

Figura 134: Interfaz HU25 – Actualizar datos de medicamentos

Fuente: Elaboración propia

HU26 – Eliminar Medicamento

En la siguiente imagen se muestra la validación para eliminar un medicamento:

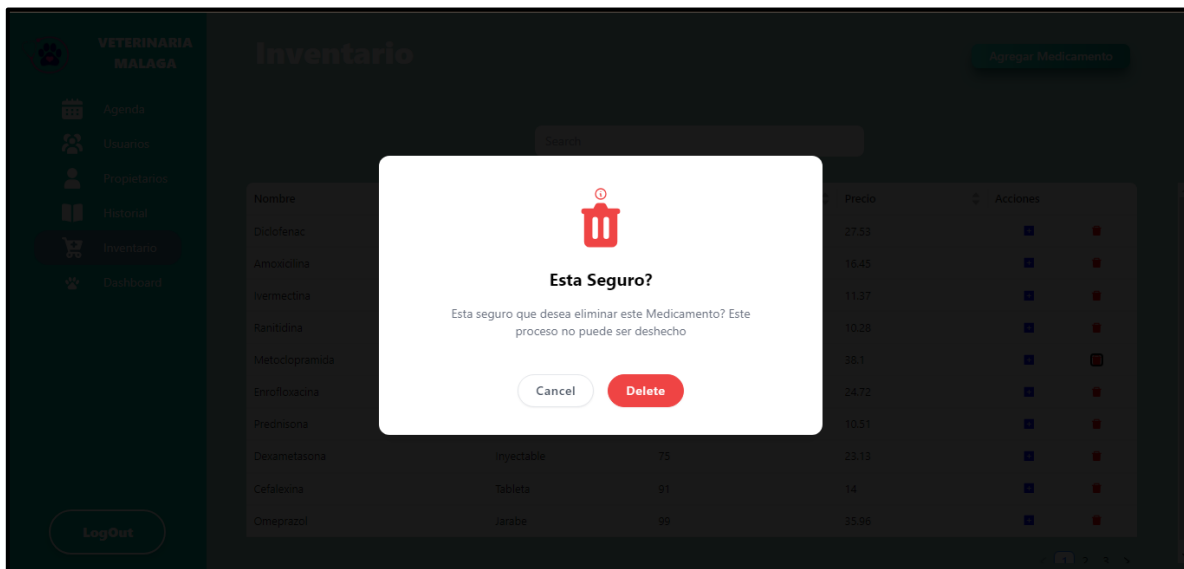


Figura 135: Interfaz HU26 – Eliminar medicamento

Fuente: Elaboración propia

HU27 – Dashboard

En la siguiente imagen se muestran las gráficas de algunos de los reportes generados por el sistema:

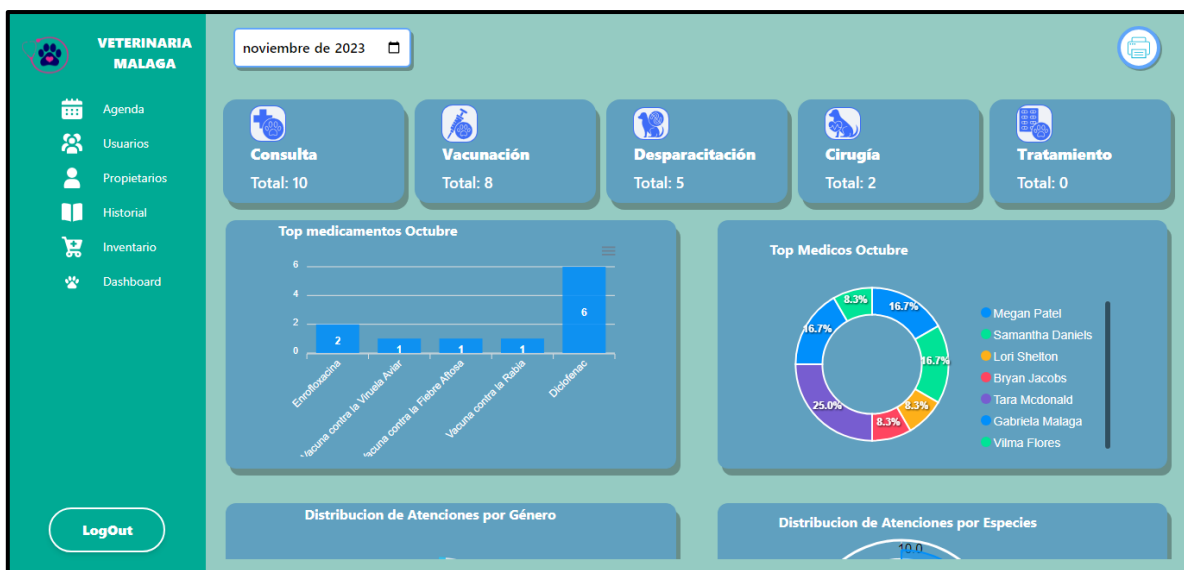


Figura 136: Interfaz HU27 – Dashboard parte 1

Fuente: Elaboración propia

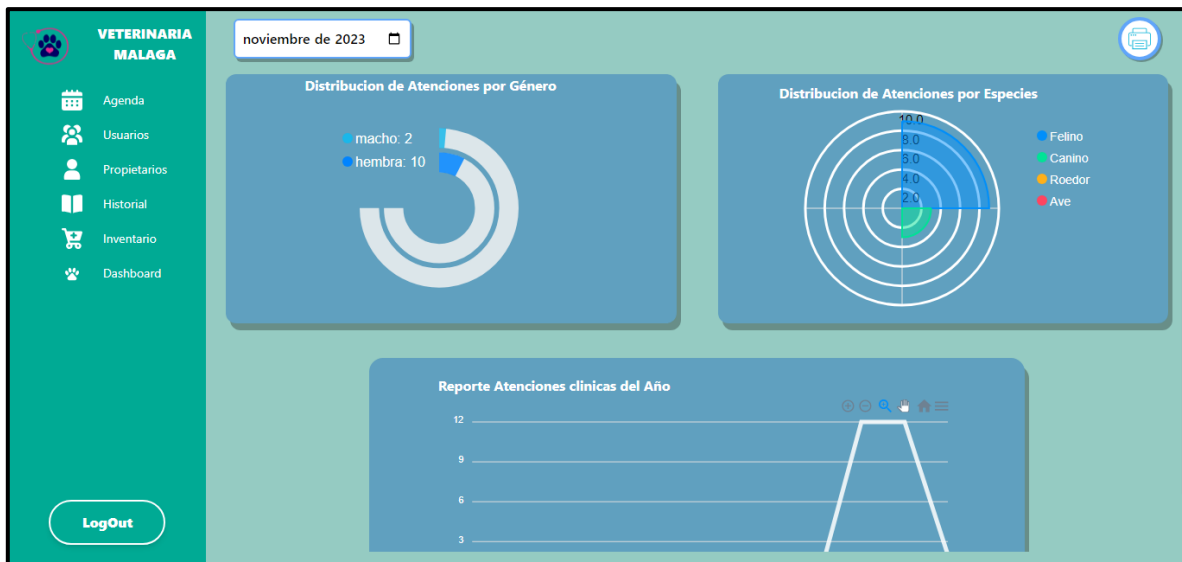


Figura 137: Interfaz HU27 – Dashboard parte 2

Fuente: Elaboración propia

II.1.2.2.3.5 Gráfico Burn Down

En la siguiente gráfica se puede observar que a lo largo del desarrollo del sprint las estimaciones fueron de acuerdo al tiempo estimado.

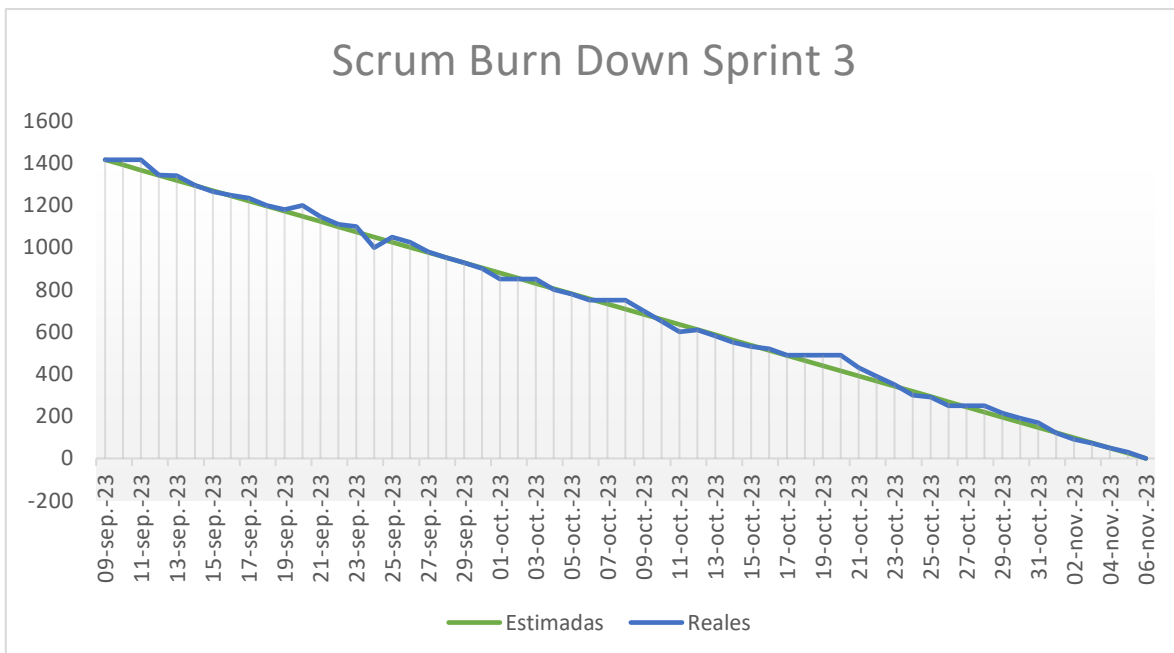


Figura 138: Gráfico Burn Down Sprint 3

Fuente: Elaboración propia

II.1.2.2.3.6 Revisión del Sprint (Sprint Review)

En la siguiente tabla se muestra el resultado que se obtuvo durante este tercer sprint, de un total de dieciséis historias implementadas, se aprobaron todas.

Tabla 33: Sprint Review

Código	Historia de usuario	Aprobada	Rechazada	Detalles
HU07	Registrar mascota	X		(OK)
HU08	Leer mascotas registradas	X		(OK)
HU09	Actualizar datos de mascotas	X		(OK)
HU10	Dar de baja mascota	X		(OK)
HU11	Registrar cita	X		(OK)
HU12	Leer citas registradas	X		(OK)
HU13	Cancelar/Eliminar cita	X		(OK)
HU14	Registrar propietario	X		(OK)
HU15	Leer propietarios registrados	X		(OK)
HU16	Actualizar datos de propietarios	X		(OK)
HU17	Dar de baja propietario	X		(OK)
HU18	Leer historiales clínicos	X		(OK)
HU19	Reporte historia clínica	X		(OK)
HU20	Registrar atención clínica	X		(OK)
HU21	Reporte de receta impresa	X		(OK)
HU22	Reporte de recibo impreso	X		(OK)
HU23	Registrar medicamento	X		(OK)
HU24	Leer medicamentos registrados	X		(OK)
HU25	Actualizar datos de medicamentos	X		(OK)
HU26	Eliminar medicamento	X		(OK)
HU27	Dashboard	X		(OK)

Fuente: Elaboración propia

II.1.2.3 Fase 3 – Revisión General de los Sprints

II.1.2.3.1 Gráfico general Burn-up

El gráfico a continuación muestra las estimaciones que se tuvieron en la fase inicial del proyecto. Se puede observar que todas las historias de usuario fueron completadas en su totalidad.

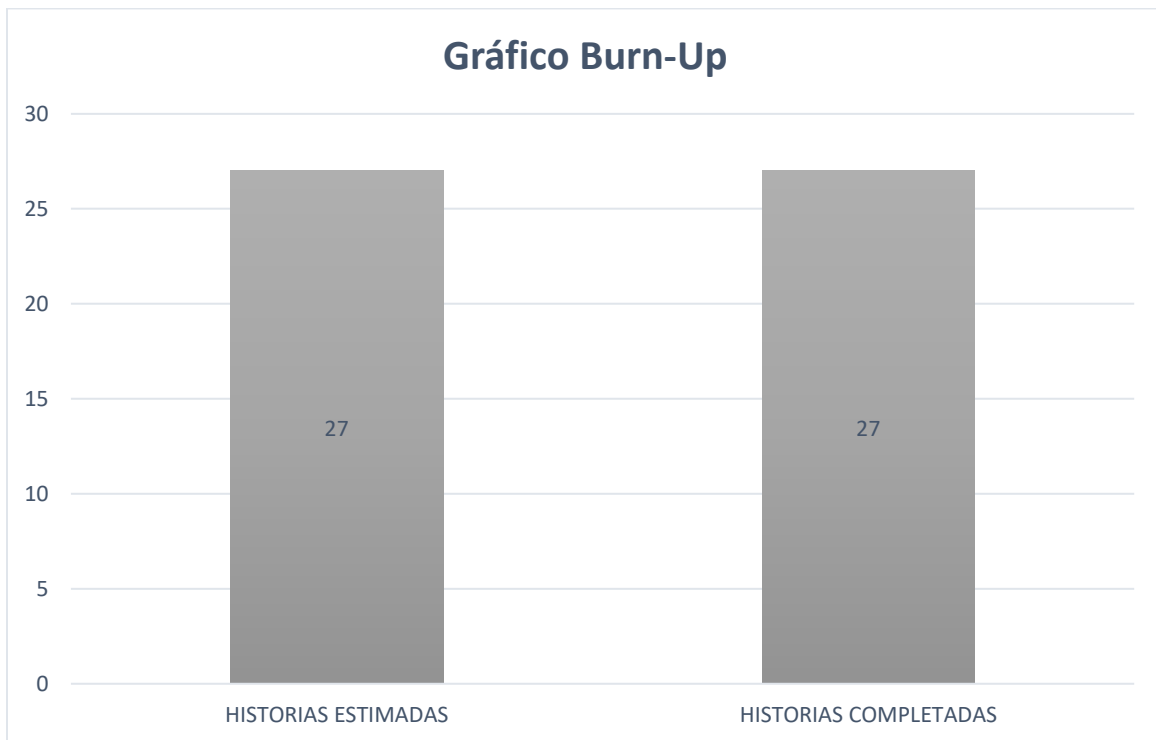


Figura 139: Gráfico Burn-Up

Fuente: Elaboración propia

II.2 Componente 2: Capacitación

II.2.1 Introducción

Al concluir con el diseño y programación del sistema y habiendo el mismo superado las distintas pruebas de funcionalidad, posteriormente corrigiendo los errores existentes se procedió a la capacitación del personal encargado del manejo futuro del sistema.

II.2.1.1 Objetivos

II.2.1.1.1 Objetivo General

Capacitar al personal de la clínica para el uso correcto del sistema automatizado.

II.2.1.1.2 Objetivos Específicos

- Realizar la capacitación al personal responsable del manejo del sistema.
- Instalar el sistema en el equipo del personal responsable.
- Realizar una guía de usuario y servicio.

II.2.2 Contexto

La capacitación se desarrollará en tres partes: la primera parte tiene como objetivo que el personal responsable del manejo del sistema, conozca en forma global los alcances y beneficios que le proporciona dicho sistema, así como los cambios positivos y responsabilidades que esto implica.

La segunda parte de la capacitación será netamente práctica, se mostrará el login y se explicará las partes que posee el sistema y el funcionamiento de cada interfaz.

Para finalizar la última parte de la capacitación constará de una breve explicación sobre el mantenimiento futuro que necesitará el sistema para continuar funcionando de manera óptima.

Nivel Ejecutivo: Se resaltaré la relevancia de la capacitación, se establecerán objetivos claros y se fomentará la participación activa del personal designado en esta categoría. Se proporcionará una visión general de la importancia estratégica de la capacitación para el éxito global del proyecto.

Personal Técnico: Las actividades de capacitación se adaptarán específicamente a las siguientes categorías de personal, asegurando una formación adecuada y especializada:

- Personal de soporte técnico al usuario final (si aplicable): Se abordarán las habilidades necesarias para brindar un soporte técnico eficiente y resolver problemas relacionados con el sistema.
- Personal de administración de servicios (si aplicable): Se enfocará en las tareas de gestión y administración de servicios relacionadas con el sistema, garantizando un manejo efectivo de las operaciones.

Usuarios Finales: Para los usuarios finales, se implementarán programas de capacitación que abordarán de manera práctica y comprensible el uso del sistema, centrándose en las funciones esenciales necesarias para llevar a cabo sus responsabilidades de manera eficiente. La capacitación se diseñará para asegurar que los usuarios finales puedan aprovechar al máximo las capacidades del sistema en su contexto operativo.

II.2.3 Propuesta Pedagógica

La propuesta pedagógica define la estrategia y enfoque que guiará la capacitación del personal de la Clínica Veterinaria Málaga en el uso del nuevo sistema.

El aprendizaje colaborativo se entiende como el proceso en el que los alumnos aprenden mientras proponen y comparten ideas para resolver una tarea, favoreciéndose con el diálogo y la reflexión sobre las propuestas propias y las de sus compañeros.

Este punto sí podría enriquecerse con el uso de las tecnologías. Se trata por tanto de construir sistemas muy adaptables a los diferentes grupos de usuarios, en donde el soporte y la intervención no impongan un comportamiento prescriptivo. Desde el campo de la psicología, algunos autores, especialmente ligados a lo que se ha llamado la psicología socio-cultural, postulaban que aprender es una experiencia de carácter fundamentalmente social, en donde el lenguaje juega un papel básico como herramienta de mediación no sólo entre profesor y alumno sino también entre compañeros.

Esta estrategia se ha diseñado cuidadosamente para asegurar que la capacitación sea efectiva y relevante para las necesidades específicas del personal y de la clínica.

II.2.3.1 Elaboración del Material

Material de Capacitación: Se desarrollará un conjunto de materiales de capacitación que incluirá manuales de usuario, presentaciones interactivas y ejercicios prácticos. Estos recursos se crearán de manera clara y concisa para facilitar la comprensión y el aprendizaje.

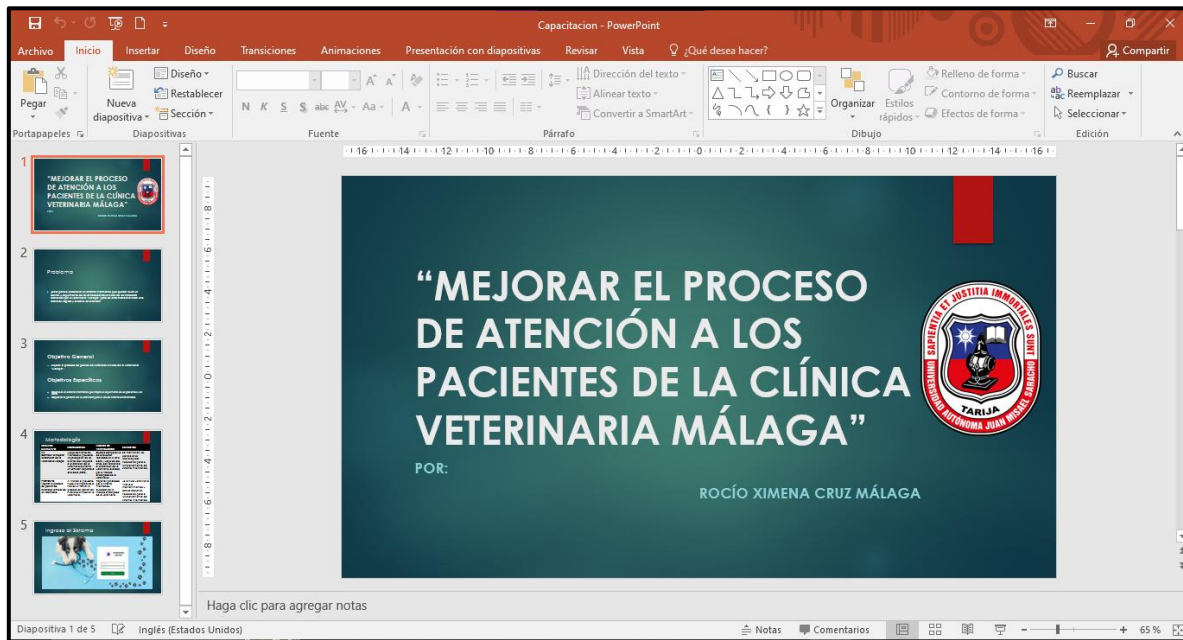


Figura 140: Diapositivas para capacitación

Fuente: Elaboración propia

Guía de Usuario: Se elaborará una guía de usuario detallada que servirá como referencia para el personal en el uso cotidiano del sistema. Esta guía proporcionará instrucciones paso a paso y consejos útiles.

II.2.3.2 Estrategia de Capacitación

Enfoque de Aprendizaje Activo: La capacitación se basará en un enfoque de aprendizaje activo, fomentando la participación activa de los empleados a través de ejercicios prácticos, discusiones grupales y escenarios de uso real del sistema.

Sesiones de Capacitación en Pequeños Grupos: Se llevarán a cabo sesiones de capacitación en grupos pequeños para garantizar la atención personalizada y la interacción con los instructores. Esto permitirá a los participantes plantear preguntas y resolver dudas de manera efectiva.

Evaluación Continua: Se realizarán evaluaciones periódicas durante la capacitación para medir el progreso de los participantes. Estas evaluaciones ayudarán a identificar áreas en las que se necesita refuerzo y asegurarán que los objetivos de aprendizaje se cumplan.

La propuesta pedagógica se ha diseñado con el objetivo de garantizar que la capacitación sea eficaz, interactiva y personalizada para las necesidades del personal de la Clínica Veterinaria Málaga. La combinación de materiales de alta calidad y una estrategia de capacitación práctica permitirá una transición exitosa hacia la implementación del sistema.

II.2.3.3 Desarrollo de la Capacitación

Esta sección detalla el proceso de implementación de la capacitación, incluyendo aspectos clave como la planificación de las sesiones, la logística y las evaluaciones. El desarrollo de la capacitación se llevará a cabo de la siguiente manera:

- **Planificación de Sesiones**

Se ha establecido un programa de capacitación detallado que abarca los diferentes módulos del sistema y sus respectivas funcionalidades.

Cada sesión de capacitación se ha programado con fechas y horarios específicos, teniendo en cuenta las disponibilidades del personal.

Se ha asignado un instructor capacitado para cada sesión y se han definido los temas y objetivos a cubrir en cada una, en caso de ser más de una.

- **Logística**

Las sesiones de capacitación se llevarán a cabo en instalaciones de la clínica, con los recursos tecnológicos necesarios, como computadoras y acceso a internet.

Se ha asegurado el acceso al sistema en los equipos donde se realizarán las prácticas para una experiencia de capacitación práctica y realista.

Se proporcionarán materiales de capacitación impresos, como manuales y hojas de referencia.



Figura 141: Capacitación

Fuente: Elaboración propia

- **Participantes**

El personal de la clínica ha sido informado con anticipación sobre las fechas y horarios de las sesiones de capacitación para que puedan planificar su asistencia.

Se ha dividido a los empleados en grupos pequeños para permitir una interacción más efectiva y una atención personalizada durante la capacitación.

- **Evaluación**

Se llevarán a cabo evaluaciones periódicas durante y al final de cada sesión de capacitación para medir el progreso y la comprensión de los participantes.

Las evaluaciones se utilizarán para identificar áreas de mejora y para ajustar el enfoque de capacitación si es necesario.



Figura 142: Capacitación Evaluación

Fuente: Elaboración propia

II.2.4 Plan de Clases

Tabla 34: Capacitación (Plan de Clases)

Nro.	Contenido	Objetivo	Duración	Material Didáctico	Medio De Enseñanza
1	Introducción	Que el personal responsable de la “Clínica Veterinaria Málaga” conozca en forma global los alcances y beneficios del sistema “Mejorar El Proceso De Atención A Los Pacientes De La Clínica Veterinaria Málaga”	15 min	Diapositivas	Computadora, Laptop
2	Despliegue	<p>Explicación sobre el despliegue del manejo del sistema:</p> <ul style="list-style-type: none"> • Cómo ingresar al sistema con su usuario y clave. • Cómo Agendar y eliminar una cita. • Cómo agregar un usuario nuevo, modificar y dar de baja un usuario registrado. • Cómo agregar un propietario nuevo, modificar y dar de baja un propietario registrado. • Cómo agregar una mascota nueva, modificar y dar de baja una mascota registrada. • Cómo controlar los historiales clínicos. • Cómo registrar una atención clínica, generar receta y recibo. 	60 min	Diapositivas, Guía de uso.	Computadora, Laptop

		<ul style="list-style-type: none"> • Cómo agregar un medicamento nuevo, modificar y dar de baja un medicamento registrado. • Cómo generar e interpretar los reportes gráficos. 			
3	Mantenimiento y Plan de Contingencia	Explicación sobre el correcto mantenimiento del sistema. Explicación de los procedimientos del plan de contingencia.	45 min	Documentación	Computadora, Laptop

Fuente: Elaboración propia

II.2.5 Cronograma

Id	% trabajo completado	Modo de tarea	SPRINT	Nombre de tarea	Duración	Comienzo	Fin	6 nov '23	13 nov '23	20 nov '23
28	100%	★	4	DISEÑO DE	2 días	mar 7/11/23	mié 8/11/23	■		
29	100%	★	4	PREPARACION DEL	2 días	jue 9/11/23	vie 10/11/23	■		
30	100%	★	4	CORRECCION DEL	1 día	lun 13/11/23	lun 13/11/23		■	
31	100%	★	4	DIA 1 CAPACITACION	1 día	mar 14/11/23	mar 14/11/23		■	
32	100%	★	4	RECOLECCION DE DUDAS	1 día	mié 15/11/23	mié 15/11/23		■	
33	100%	★	4	DIA 2 CAPACITACION	1 día	jue 16/11/23	jue 16/11/23		■	
34	100%	★	4	DIA 3 CAPACITACION	1 día	vie 17/11/23	vie 17/11/23		■	
35	100%	★	4	RECOLECCION DE DUDAS	1 día	lun 20/11/23	lun 20/11/23			■

Figura 143: Cronograma Actividades C2

Fuente: Elaboración propia

II.2.6 Resultados Esperados

- Al finalizar el proyecto, se han realizado al menos dos capacitaciones en el uso del sistema informático a los encargados de la Clínica Veterinaria Málaga.

II.2.7 Medios de Verificación

- Los medios de verificación se encuentran en el anexo F. *Ver anexo F*

II.2.8 Conclusiones

El Componente 2, centrado en la capacitación, culminó con éxito al lograr una implementación efectiva del programa de formación para todas las categorías de personal, desde el nivel ejecutivo hasta los usuarios finales. Se destacaron la comprensión profunda del sistema, la identificación de áreas de mejora y la superación de desafíos iniciales. La capacitación proporcionó una base sólida para la próxima fase de implementación, demostrando un alto nivel de compromiso del personal con el éxito del proyecto y preparándolos para enfrentar con confianza sus responsabilidades con el nuevo sistema en funcionamiento.

CAPÍTULO III: CONCLUSIONES Y RECOMENDACIONES

III.1 Conclusiones

Según la información recolectada durante la investigación sobre los datos manejados en la “Clínica Veterinaria Málaga” de la ciudad de Tarija:

- La implementación del sistema ha sido exitosa. Se ha logrado mejorar significativamente la eficiencia en la gestión de los pacientes, los registros médicos y la comunicación interna en la clínica. El sistema ha sido bien recibido por el personal y ha facilitado la prestación de servicios de alta calidad.
- El sistema ha mejorado la atención al paciente al agilizar los procesos de diagnóstico, seguimiento y tratamiento. La automatización de tareas administrativas ha permitido al personal médico centrarse más en la atención médica directa, lo que ha tenido un impacto positivo en la calidad de los servicios veterinarios.
- La implementación del sistema ha mejorado la eficiencia operativa de la clínica. La capacidad de programar citas, gestionar registros y acceder a información clínica de manera rápida y precisa ha reducido el tiempo dedicado a tareas administrativas y ha aumentado la productividad del personal.
- Se ha observado un aumento en la satisfacción del personal con el nuevo sistema. La capacitación efectiva y el soporte técnico adecuado han contribuido a que los empleados se sientan cómodos y seguros al utilizar la tecnología, lo que ha mejorado su moral y su rendimiento laboral.
- A lo largo del proyecto, se han enfrentado desafíos técnicos y de capacitación, que se han superado con éxito. La colaboración y el compromiso del equipo de trabajo, así como la adaptación a las necesidades cambiantes, han sido clave en la resolución de problemas.

III.2 Recomendaciones

A continuación, se presentan recomendaciones basadas en las lecciones aprendidas durante la ejecución del proyecto y en la implementación del sistema. Estas recomendaciones tienen como objetivo optimizar aún más el uso del sistema y el proceso de atención a los pacientes en la Clínica Veterinaria Málaga.

- Se recomienda establecer un programa de mantenimiento regular para garantizar que el sistema siga funcionando sin problemas. Esto incluye actualizaciones de software, resolución de errores y mejoras de seguridad.
- Para asegurar que el personal siga utilizando el sistema de manera eficaz, se sugiere realizar capacitaciones periódicas. Estas capacitaciones pueden incluir actualizaciones sobre nuevas características y mejores prácticas.
- Es importante implementar un sistema de seguimiento y medición del desempeño del sistema para evaluar su impacto continuo en la clínica. Esto permitirá ajustar y mejorar el sistema según sea necesario.
- Fomentar la retroalimentación constante del personal y los usuarios para identificar áreas de mejora y realizar ajustes en el sistema o en la capacitación.
- Considerar la posibilidad de expandir el uso del sistema a otras áreas de la clínica o a clínicas veterinarias asociadas para aprovechar al máximo su potencial.

Estas recomendaciones se basan en las experiencias del proyecto y están diseñadas para ayudar a la Clínica Veterinaria Málaga a continuar beneficiándose del sistema y a mantener su eficiencia operativa a lo largo del tiempo.