



# **INSTITUTO TECNOLÓGICO SUPERIOR DE MISANTLA**

---

## **“DESARROLLO DE APLICACIÓN MÓVIL PARA EL CONTROL REMOTO DEL PROTOTIPO DISPENSADOR PET-HOME ALBA UTILIZANDO EL IOT”**

### **TESIS**

QUE PARA OBTENER EL TÍTULO DE

**INGENIERO EN TECNOLOGÍAS DE LA  
INFORMACIÓN Y COMUNICACIONES**

P R E S E N T A:

**CINTHIA GUTIÉRREZ GORGOÑO**

ASESOR

**MSC. ARNULFO GAMALIEL HERNÁNDEZ  
GONZÁLEZ**

Co-ASESORES

**LIC. DANIELA LIZBETH HERNÁNDEZ SÁNCHEZ**

**DR. ALEJANDRO DEL REY TORRES RODRIGUEZ**

MISANTLA, VERACRUZ

MARZO, 2020



INSTITUTO TECNOLÓGICO SUPERIOR DE MISANTLA  
DIVISIÓN DE ESTUDIOS PROFESIONALES  
AUTORIZACIÓN DE IMPRESIÓN DE TRABAJO DE TITULACIÓN

FECHA: 03 de Marzo de 2020.

ASUNTO: AUTORIZACIÓN DE IMPRESIÓN  
DE TESIS PROFESIONAL.

**A QUIEN CORRESPONDA:**

Por medio de la presente hago constar que el (la) C:

**CINTHIA GUTIÉRREZ GORGONÓ**

pasante de la carrera de INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIONES con No. de Control 152T0081 ha cumplido satisfactoriamente con lo estipulado por el **Manual de Procedimientos para la Obtención del Título Profesional de Licenciatura** bajo la opción **Titulación Integral (Tesis Profesional)**

Por tal motivo se Autoriza la impresión del **Tema** titulado:

**“DESARROLLO DE APLICACIÓN MÓVIL PARA EL CONTROL REMOTO DEL PROTOTIPO DISPENSADOR PET-HOME ALBA UTILIZANDO EL IOT”**

Dándose un plazo no mayor de un mes de la expedición de la presente a la solicitud del Acto de Recepción para la obtención del Título Profesional.

ATENTAMENTE

ING. GERBACIO TLAXALO ESPINOZA  
DIVISIÓN DE ESTUDIOS PROFESIONALES



Archivo.

## **Agradecimientos**

### **A mi familia.**

Gracias a Dios por permitirme tener y disfrutar a mi familia, gracias a mi familia por apoyarme en cada decisión y proyecto, no ha sido sencillo el camino hasta ahora, pero gracias a sus aportes, a su amor, a su inmensa bondad y apoyo, lo complicado de lograr esta meta se ha notado menos. Les agradezco, y hago presente mi gran afecto hacia ustedes, mi hermosa familia.

### **A mis padres.**

Agradezco a mis padres, pues ellos han sido los principales promotores de mis sueños, gracias por cada día confiar y creer en mí y en mis expectativas, gracias a mi mamá por estar dispuesta a acompañarme cada noche de estudio, por alentarme y no dejarme sola cuando sentía que no podía más, a mi papá por desear y anhelar siempre lo mejor para mi vida, gracias por cada consejo y por cada una de sus palabras que guiaron durante mi vida.

### **A mi hermana.**

Has sido una de las principales personas involucradas en ayudarme a que este proyecto fuera posible, gracias por el apoyo y estar siempre para mí en las buenas y en las malas. Te quiero mucho hermana.

### **A mis profesores.**

Agradezco a mis formadores, personas de gran sabiduría quienes se han esforzado por ayudarme a llegar al punto en el que me encuentro, el proceso no ha sido sencillo, pero gracias a la dedicación de transmitir sus conocimientos conmigo ha logrado importantes objetivos como culminar el desarrollo de este proyecto con éxito.

### **A mis amigos.**

El desarrollo de esta tesis no lo puedo catalogar como algo fácil, pero lo que sí puedo hacer, es afirmar que durante todo este tiempo pude disfrutar de cada momento, que cada investigación, proceso, y proyectos que se realizaron dentro de esta, lo disfruté mucho, y no fue porque simplemente me dispuse a que así fuera, fue porque mis amigos siempre estuvieron ahí, fue porque la vida misma me demostró que de las cosas y actos que yo realicé, serán los mismos que harán conmigo.

## **Dedicatoria**

### **A mi papá.**

Por ser un pilar importante en mi vida, apoyarme y sacar a nuestra familia adelante, gracias por todos tus esfuerzos, los desvelos y por todo lo que haces, te aseguro que valdrá la pena, estoy tan agradecida de tener un papá como tú eres el mejor hombre que conozco, nunca nadie podrá tener el amor que te tengo.

### **A mi mamá.**

Por estar conmigo siempre, en las buenas y malas por todo el esfuerzo que haces para que mi hermana y yo seamos personas de bien, por todo el amor que nos das, cuidarnos siempre, por los consejos, los juegos, las risas, por darme la vida, todo el esfuerzo valdrá la pena, el amor que te tengo no lo tendrá nadie más.

### **A mi hermana.**

Por ser mi mejor amiga cuidarme, apoyarme, orientarme y enseñarme a ser una persona valiente, estoy tan feliz de tenerte, desde que estás conmigo nunca estaré sola, no sabes el orgullo que siento por ser tu hermana te quiero demasiado.

### **A mis amigos, mis personas favoritas.**

Por los más grandes momentos que pasamos durante estos 4 años, son muy especiales para mí, por aceptarme tal y como soy, por hacerme reír en los momentos malos y ayudarme siempre que podía, soy muy afortunada de tenerlos en mi vida, me di cuenta que no es necesario estar rodeada de muchas personas si tienes a las correctas en tu vida, seremos pocos, pero siempre unidos, los quiero Montse, Dalia, David, César. Ricardo, you are the most wonderful person, I love you thanks for everything.

“No te das cuenta de lo rápido que todo se termina hasta que sucede,  
y no quieres desperdiciar todo lo bueno de la vida hasta que  
termina”

## Resumen

El trabajo que se presenta a continuación se realizó en la empresa ALBA Desarrollo Tecnológico e Innovación (ALBA-DTI), la cual en busca de explorar en el mundo del Internet de las cosas (IoT) decide el desarrollo de un prototipo dispensador de alimento dosificable, es decir, suministrar la cantidad de alimento de acuerdo a la información de las mascotas que va dirigido especialmente para perros y gatos. El funcionamiento de este prototipo consiste en un nodo central siendo el servidor, un nodo lector y una aplicación móvil para su control remoto. Para el servidor se ha instalado Mosquitto como Broker MQTT al ser Open-source beneficia en la disminución de los costos y es adecuado para el uso de producción; permite la comunicación por el protocolo Message Queue Telemetry Transport (MQTT) con la aplicación móvil y el nodo lector, siendo que este Broker se encuentra alojado en una Raspberry Pi que trabaja en un entorno Ubuntu. El nodo lector consiste en un módulo NodeMcu ESP8266 al que se conecta un circuito electrónico integrado por sensores que obtienen el nivel de alimento y actuadores que ejecutan la tarea de dispensar el alimento. Por último, la aplicación móvil se realiza para controlar de forma remota en tiempo real o automática el prototipo, asimismo monitorear el nivel de alimento en el contenedor del dispensador; la información ingresada de la mascota y los horarios para el dispenso del alimento por la aplicación se almacena en la base de datos que se encuentra en el servidor, cabe mencionar que al hacer uso de Flutter permite que el diseño de la aplicación sea atractiva e intuitiva para el usuario.

# ÍNDICE DE CONTENIDO

Introducción. ....	10
Capítulo I. Generalidades del proyecto. ....	11
1.1. Planteamiento del problema. ....	11
1.2. Antecedentes. ....	12
1.2.1. Dispensadores comunes. ....	12
1.2.2. Dispensadores automáticos o inteligentes. ....	13
1.2.2.1. Dispensador Petnet. ....	13
1.2.2.2. Dispensador Bistro. ....	13
1.2.2.3. Dispensador Fresh Element. ....	14
1.2.2.4. Dispensador CleverPet. ....	14
1.2.2.5. Dispensador Catspad. ....	15
1.2.2.6. Dispensador Petsafe. ....	16
1.2.2.7. Dispensador Petmate. ....	16
1.3. Justificación. ....	17
1.4. Objetivos. ....	19
1.4.1. Objetivo general. ....	19
1.4.2. Objetivos específicos. ....	19
1.5. Hipótesis. ....	19
1.6. Alcances y limitaciones. ....	1
1.6.1. Alcances. ....	1
1.6.2. Limitaciones. ....	1
1.7. Propuesta de solución. ....	2
Capítulo II. Análisis de los fundamentos. ....	3
2.1. Marco histórico. ....	3

2.1.1. El internet de las cosas. ....	3
2.1.2. El dispensador. ....	4
2.2. Marco teórico. ....	5
2.2.1. Base de datos. ....	5
2.2.2. Sistema gestor de base de datos. ....	5
2.2.2.1. Oracle. ....	5
2.2.2.2. PostgreSQL. ....	5
2.2.2.3. MySQL. ....	6
2.2.3. Servidor web. ....	6
2.2.3.1. Wamp. ....	6
2.2.3.2. Laragon. ....	6
2.2.3.3. Lamp. ....	7
2.2.4. Software de diseño. ....	7
2.2.4.1. Adobe Illustrator. ....	7
2.2.4.2. Adobe Photoshop. ....	7
2.2.4.3. Adobe XD. ....	8
2.2.5. Desarrollo multiplataforma. ....	8
2.2.5.1. Aplicaciones nativas. ....	9
2.2.5.2. Aplicaciones web. ....	9
2.2.5.3. Aplicaciones híbridas. ....	9
2.2.6. Plataforma de desarrollo móvil. ....	11
2.2.6.1. Xamarin. ....	11
2.2.6.2. Ionic. ....	11
2.2.6.3. Flutter. ....	11
2.2.7. Entornos de desarrollo. ....	12



2.2.7.1. Android studio.....	12
2.2.7.2. Visual Studio Code.....	12
2.2.8. Arquitectura de software. ....	13
2.2.8.1. Vanilla. ....	13
2.2.8.2. Scoped Model. ....	14
2.2.8.3. BLoC. ....	14
2.2.9. Lenguaje PHP. ....	15
2.2.10. Protocolos para el internet de las cosas. ....	16
2.2.10.1. REST API .....	16
2.2.10.2. MQTT.....	17
2.3. Estado del arte. ....	18
2.3.1. Dispensadores automáticos .....	18
2.3.1.1. Dispensador automático de alimento para mascotas.....	18
2.3.1.2. Programmable Pet Feeder.....	18
2.3.1.3. Happy Pet Dispensador electrónico para mascotas de agua y comida.....	18
2.3.1.4. Automatic Pet Feeder .....	19
2.3.1.5. Dispensador automático de alimento seco para mascotas caninas .....	19
2.3.1.6. Diseño de dispensador automático de alimento para mascotas usando una celda .....	19
2.3.2. Dispensadores basados en el IoT. ....	20
2.3.2.1. Smart dog Feeder design using Wireless communication, MQTT and Android Client .....	20
2.3.2.2. Desarrollo de un prototipo para dispensar alimento y agua para mascotas con tecnología IOT monitoreado y controlado a través de un dispositivo móvil .....	20
2.3.2.3. The study and Application of the IoT in Pet System .....	20
2.3.2.4. Control domótico basado en el protocolo MQTT .....	21

2.3.2.5. Monitorización de sensores con Arduino utilizando el protocolo MQTT ..	21
2.3.3. Dispensadores a control remoto.....	21
2.3.3.1. Diseño e implementación de un sistema dispensador de alimento para mascota, controlado vía ethernet .....	21
2.3.3.2. Arduino-based Food and Water Dispenser for Pets with GSM Technology Control .....	22
2.3.3.3. Dispensador automático de comida para mascotas, programable y controlado remotamente .....	22
2.3.3.4. Low-Cost IoT Design and Implementation of a Remote Food and Water Control System for Pet Owners .....	23
2.3.3.5. Automatización y asistencia remota para cuidado de mascotas .....	23
Capitulo III. Desarrollo.....	24
3.1. Metodología.....	25
3.2. Análisis.....	28
3.2.1. Obtener requerimientos. ....	28
3.2.1.1. Diseño de la investigación y procedimiento. ....	28
3.2.1.2. La población y la muestra.....	28
3.2.1.3. Criterios de inclusión y exclusión. ....	30
3.2.1.4. Instrumentos.....	30
3.2.2. Clasificar requerimientos. ....	31
3.2.2.1. Requerimientos no funcionales o de información.....	31
3.2.2.2. Requerimientos funcionales.....	32
3.2.2.3. Elaboración de las tablas de porciones de alimento.....	41
3.3. Diseño. ....	43
3.3.1. Estructurar el software. ....	43
3.3.1.1. Modelo funcional. ....	43

3.3.1.2. Casos de uso.....	44
3.3.2. Asignar recursos. ....	44
3.3.2.1. Herramientas para el desarrollo. ....	44
3.3.2.2. Maquetación de la aplicación.....	45
3.3.2.3. Selección de colores. ....	46
3.3.2.4. Diseño UI de la aplicación.....	47
3.4. Desarrollo. ....	50
3.4.1. Desarrollo de la aplicación en Flutter. ....	50
3.4.2. Instalación de Flutter.....	50
3.4.3. Plugin de Flutter en Visual Studio Code. ....	51
3.5. Pruebas de funcionamiento. ....	52
3.5.1. Emulación y simulación. ....	52
3.5.2. Dispositivos reales. ....	52
Capítulo IV. Análisis de los resultados. ....	54
4.1. Datos Pet-Home ALBA .....	54
4.2. Aplicación móvil. ....	55
4.2.1. Pantallas de inicio: Splashscreen y login. ....	55
4.2.2. Pantalla de error de datos. ....	56
4.2.3. Pantalla principal y menú. ....	57
4.2.4. Modo automático .....	59
4.2.5. Perfil mascota: Tabla de información y lista de mascota.....	60
4.2.6. Perfil mascota: Registro y detalle de la mascota .....	61
4.2.7. Perfil mascota: Editar y eliminar mascota.....	62
Conclusiones. ....	63
Referencias.....	65

Anexos .....	70
--------------	----

## ÍNDICE DE FIGURAS

Figura 1. Dispensador común. ....	12
Figura 2. Dispensador Petnet. ....	13
Figura 3. Dispensador Bistro. ....	14
Figura 4. Dispensador Fresh Element. ....	14
Figura 5. Dispensador CleverPet.....	15
Figura 6. Dispensador Catspad. ....	15
Figura 7. Dispensador Petsafe.....	16
Figura 8. Dispensador Petmate. ....	16
Figura 9. Porcentaje de la población según condición de Internet, 2018. ....	17
Figura 10. Porcentaje de la población según uso de telefonía celular.....	18
Figura 11. Propuesta de solución: Pet-Home ALBA. ....	2
Figura 12. Cronograma de actividades. ....	24
Figura 13. Metodología propuesta para el proyecto.....	25
Figura 14. Ecuación para obtener del tamaño de la muestra. ....	28
Figura 15. Determinar el tamaño de muestra.....	30
Figura 16. Porciones de alimento para gatos. ....	41
Figura 17. Porciones de alimento para perros.....	42
Figura 18. Modelo funcional.....	43
Figura 19. Casos de uso.....	44
Figura 20. Maquetación Pet Home parte 1. ....	45
Figura 21. Maquetación Pet Home parte 2. ....	46
Figura 22. Maquetación Pet Home parte 3. ....	46
Figura 23. Diseño UI Splashscreen, Login, error de datos. ....	48
Figura 24. Diseño UI Menú y modo automático. ....	48
Figura 25. Diseño UI Perfil de mascotas.....	49
Figura 26. Diseño UI Registro, editar y eliminar mascotas.....	49
Figura 27. Consola para ejecutar comandos de Flutter. ....	51
Figura 28. Agregar Plugin de Flutter.....	51
Figura 29. Iniciar un proyecto en Flutter. ....	52
Figura 30.Resultado de la población que cuenta con Internet y un Smartphone. ....	54

Figura 31. Resultado de la población que cuentan con una mascota y que tipo es.....	55
Figura 32. Pantallas de inicio: Splashscreen y Login.....	56
Figura 33. Pantalla error de datos. ....	57
Figura 34. Pantallas después de inicio: Pantalla principal y menú. ....	58
Figura 35. Pantalla de modo automático. ....	59
Figura 36. perfil mascota: Tabla de información y lista de mascota.....	60
Figura 37. Perfil mascota: Registro y detalle de la mascota.....	61
Figura 38. Perfil mascota: Editar y eliminar mascota. ....	62

## ÍNDICE DE TABLAS

Tabla 1. Características de las diferentes opciones para disponer de funcionalidades especiales en dispositivos móviles. ....	10
Tabla 2. Arquitectura de software BLoC.....	15
Tabla 3. Nivel de confianza. ....	29
Tabla 4. SplashScreen. ....	31
Tabla 5. Iniciar sesión.....	32
Tabla 6. Error de datos. ....	33
Tabla 7. Inicio. ....	34
Tabla 8. Menú Principal. ....	35
Tabla 9- Modo automático.....	37
Tabla 10. Perfil de mascota (Tablas).....	38
Tabla 11. Perfil de mascota.....	39
Tabla 12. Eliminar mascota. ....	40
Tabla 13. Herramientas de desarrollo.....	44
Tabla 14. Colores a utilizar.....	47
Tabla 15. Dispositivos móviles utilizados en pruebas de aplicación móvil. ....	53

## Introducción.

En la actualidad las personas cuentan con un ritmo de vida acelerado a lo que ha llevado a *“la sociedad de la información y del conocimiento el acelerado desarrollo de productos intensivos en tecnología y con un gran valor agregado que se inserta en la sociedad, a nivel mundial”* [1], lo cual ha inundado nuestro entorno con televisores, teléfonos, módems, redes y etc., por medios de sistemas tecnológicos integrados; uno de los responsables de estos acontecimientos son las *Tecnologías de la información y comunicación* (TIC) siendo uno de sus paradigmas que en la actualidad está ganando cada vez más reconocimientos: el *Internet de las cosas* y más aún en el concepto de la *domótica* permitiendo controlar diferentes tareas del hogar de forma remota e incluso automatizarlas mejorando la calidad de vida.

México es uno de los países con un gran porcentaje de su población que cuenta con alguna mascota, siendo una de estas razones para aprovechar el innovar productos en este campo. por esto, se lleva a cabo el desarrollo de un prototipo dispensador para perros y gatos que brinde el suministro de alimento para la mascota ofreciendo al usuario el control de porción del alimento a ingerir con base a la información de la mascota. Este prototipo, está basado en Raspberry pi y Arduino de tal manera que la integración de estas herramientas permite al usuario monitorear y controlar de forma remota y automática mediante un dispositivo móvil. Este proyecto de titulación se encuentra organizado de la siguiente forma, donde:

Capítulo I. Generalidades muestra el porqué, los objetivos, alcances y limitaciones que presenta el desarrollo del proyecto *Pet-Home ALBA*. Capítulo II. Análisis de los fundamentos subdividido en un marco histórico que son los conceptos y áreas de aplicación, marco teórico siendo las herramientas para solucionar el desarrollo del proyecto y estado del arte que son antecedentes del proyecto. Capítulo III. Consiste en el desarrollo que provee en detalle la metodología del prototipo dividida en secciones la parte hardware siendo el nodo lector y el servidor, y la parte software que consiste en una aplicación móvil para cumplir los requerimientos y permitir el funcionamiento del sistema. Capítulo IV. Se muestra el análisis de los datos y resultados siendo el funcionamiento del prototipo y la aplicación móvil y las Conclusiones donde se presentan una compilación de los conocimientos adquiridos en el desarrollo de este proyecto.



# **Capítulo I. Generalidades del proyecto.**

## **1.1. Planteamiento del problema.**

El ser humano ha recibido grandes beneficios de los animales uno de los más importantes, su compañía. Actualmente, se ha incrementado el número de animales domésticos en los hogares, debido a diferentes factores sobre todo porque representan una fuente importante de afecto. En México de cada 100 personas 57 tienen una mascota en su hogar, según datos obtenido del INEGI; los perros representan aproximadamente el 85% y los gatos un 15% del total [2]. El tener un nuevo integrante en la familia implica una responsabilidad sobre satisfacer sus necesidades básicas como la adecuada alimentación, un espacio físico amplio, cuidado sanitario por mencionar algunos. Por lo tanto, la mayoría de las personas enfrentan cambios en su estilo de vida que se deben a diferentes compromisos personales que conllevan a la irresponsabilidad por parte del dueño para satisfacer las necesidades de su mascota afectando a una de las más importantes, la alimentación.

Para evitar este problema los dueños de las mascotas adquieren un dispensador de alimento para abastecer de alimento a sus mascotas en situaciones que no pueden hacerlo. Actualmente se encuentran dos tipos de dispensadores existentes en el mercado: Dispensadores comunes los cuales no proporcionan la cantidad recomendada de alimento para las mascotas además de que no hay forma de comunicar la necesidad de abastecerse de más alimento, lo que provoca una mala alimentación. También, se encuentran los dispensadores automáticos los cuales se disponen de dos tipos los programables que permiten ajustar las porciones de comida y determinar la hora; y aquellos que funcionan con WiFi a través de una aplicación móvil en los que ajustan las porciones de alimento a dispensar y visualizan la cantidad de alimento restante en el dispensador.

Teniendo en cuenta lo anterior, el presente proyecto propone la construcción de un dispensador de alimento y bebida para mascotas dosificable con la finalidad de suministrar la cantidad de alimento de acuerdo a la información de la mascota, mientras que el desarrollo de una aplicación móvil ayudará a controlar de manera remota el dispensador, asimismo la opción de programar las horas para dispensar el alimento, registrar a la mascota y ver la cantidad de alimento almacenado en el dispensador.

Por lo anterior, se plantea la siguiente pregunta ¿La construcción de un prototipo dispensador de alimento de bajo costo utilizando el Internet de las cosas, controlado remotamente mediante una interfaz, logrará alimentar las mascotas remotamente?

## **1.2. Antecedentes.**

En la actualidad se han desarrollado varios productos de esta temática, los cuales se clasifican en dos tipos de dispensadores: comunes y automáticos que estos a su vez tienen distintos modos de trabajo por baterías o corriente eléctrica. Los trabajos mencionados posteriormente harán énfasis en las características de cada módulo que compone la estructura física del dispensador y su funcionalidad. Aunque en el mercado comercial existe una gran variedad de dispensadores de alimento para cada tipo de mascotas se deben tener algunos puntos generales en cuenta, ya sea por la raza, tamaño, peso, dosificación adecuada, tipo de alimento o condiciones especiales que requiera la mascota y el gusto de los usuarios.

### **1.2.1. Dispensadores comunes.**

La *Figura 1* muestra los dispensadores comunes que consisten en un depósito en donde se almacena el alimento el cual se va dispensando en cantidades que tal vez no sean recomendables para la mascota sobre su base a un cuenco sea de plástico o cerámica, sin embargo, depende de las mascotas la decisión de alimentarse, además que no poseen una funcionalidad que informe cuando el alimento se esté acabando o sea necesario rellenar el contenedor más que con el contacto visual.



*Figura 1. Dispensador común.*

### **1.2.2. Dispensadores automáticos o inteligentes.**

Con el auge de la tecnología actualmente se están cubriendo más áreas e innovando. A tal punto que un dispensador común va más allá de solo ser un simple comedero, haciendo la vida más fácil a aquellos que por ciertos motivos no pueden estar en todo momento a lado de su mascota. A continuación, se describirán algunos dispensadores automáticos e inteligentes cada uno con ciertas particularidades.

#### **1.2.2.1. Dispensador Petnet.**

Petnet es un dispensador que se controla con una aplicación móvil como se muestra en la *Figura 2* el cual se puede ajustar el horario de alimentación, recibir alertas de cuando se está produciendo la ingesta de comida y saber el inventario de comida restante en el dispositivo. Posee una batería que se activa en caso de que se corte corriente eléctrica, de manera que se asegura el suministro de la alimentación en dichos casos, el costo del dispensador es de \$8,500.00 aproximadamente.



*Figura 2. Dispensador Petnet.*

#### **1.2.2.2. Dispensador Bistro.**

Bistro es un dispensador que incorpora un sistema de reconocimiento facial como se muestra en la *Figura 3* y que es controlado por una plataforma que pesa al animal cuando acude a comer, que aportan los datos necesarios para llevar un registro de los hábitos de alimentación de cada animal, y así regular la porción en función de su peso. El dispensador envía notificaciones al móvil del propietario si se producen incidencias o cambios en la alimentación habitual, su costo oscila entre los \$4625.29.



*Figura 3. Dispensador Bistro.*

### **1.2.2.3. Dispensador Fresh Element.**

Fresh Element es un dispensador para perros y gatos como se muestra en la *Figura 4*, que permite programar desde la aplicación de Pекit configurar horarios y cantidades de comida, cambiar la configuración para cuando se requiera bloquear y activar el alimento en cualquier momento. Tiene una capacidad de 2,7kg de alimento seco, activa un led y una notificación en la app cuando baja la cantidad de alimento e incorpora un botón de alimentación, permitiendo dar le un premio extra a la mascota cuando es necesario, el costo del dispensador \$4591.48 aproximadamente.



*Figura 4. Dispensador Fresh Element.*

### **1.2.2.4. Dispensador CleverPet.**

CleverPet es un dispensador para perros como se muestra en la *Figura 5* que obliga a resolver retos visuales y sonoros para entregarle el alimento. El dueño coloca cierta cantidad de premios, si el perro obedece a la instrucción, el sistema girará y entregará el premio; en caso contrario, seguirá recordándole hasta que el perro entienda el proceso para ganar la recompensa. El dispensador se controla por una app móvil que permite a los dueños llevar

un registro de cuántos premios ha conseguido la mascota en un día, los ejercicios que logró y el tiempo que tardó en resolverlos, e incluso podrá elevar la dificultad de los retos, el costo del dispensador estimado es de \$5,179.85.



*Figura 5. Dispensador CleverPet.*

#### **1.2.2.5. Dispensador Catspad.**

Catspad es un dispensador de comida y agua como se muestra en la *Figura 6* consiste en un compartimiento de alimento con una capacidad de 1.7kg y bebida con una capacidad de 8L. Controlado por una app móvil, que permite suministrar alimento cada cierto tiempo manual o programarlo para que el dispensador funcione de forma autónoma. El dispositivo posee sensores que detectan el collar de la mascota para saber de qué gato se trata en caso de que se tenga más de un gato en casa, el costo de venta estimado es de \$4,622.88.



*Figura 6. Dispensador Catspad.*

#### 1.2.2.6. Dispensador Petsafe.

Petsafe es un dispensador para gatos y perros de raza pequeño y mediano, funciona con una aplicación móvil en iPhone, iPod, IOS 9 o posterior y Android 5.0 o posterior como se muestra en la *Figura 7* el cual permite dispensar alimento hasta doce veces al día con una capacidad de 5kg de alimento seco, el costo aproximado es de \$4,407.00.



*Figura 7. Dispensador Petsafe.*

#### 1.2.2.7. Dispensador Petmate.

Petmate dispensador que repone automáticamente el depósito de alimentos cuando los niveles son bajos como se muestra en la *Figura 8*. Cuenta con dos compartimientos uno para alimento con una capacidad de 1.4 kg y otro para agua con una capacidad de 2.84 litros, el costo aproximado del dispensador es de \$399.00.



*Figura 8. Dispensador Petmate.*

### 1.3. Justificación.

Las Tecnologías de la Información y Comunicación (TIC) han dado grandes avances en los últimos años, que son un elemento clave en la innovación y la creación de nuevos productos y negocios, el cual tiene un amplio campo de aplicación que se encuentran enfocados tanto a personas como a animales. La mayoría de las tendencias tecnológicas de las TIC van dirigidas a temáticas relacionadas con: El internet, el desarrollo de software, desarrollo de aplicaciones móviles y el Internet de las Cosas (IoT). El internet actualmente es una herramienta que se utiliza a diario y ayuda a que las actividades sean más fáciles contribuyendo a la comodidad de los usuarios y a las empresas.

El uso del internet, estudiado por el Instituto Nacional de Estadística y Geografía (INEGI) con base a “Estadísticas a propósito del día mundial del Internet (17 de mayo) Datos nacionales” que se muestra en la *Figura 9*, de las personas encuestadas de seis años o más en el país en 2018, 74.3 millones declararon sí usar Internet (65.8% de la población). Lo cual indica un avance de 1.9% respecto de los datos de 2017 (63.9%) [3].

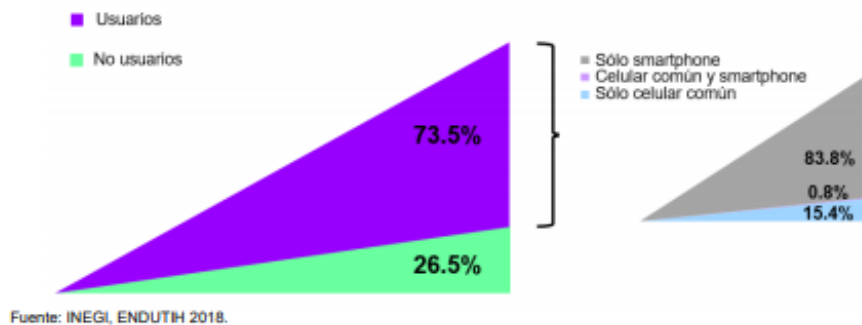


*Figura 9. Porcentaje de la población según condición de Internet, 2018.*

*Fuente: INEGI, ENDUTH 2018.*

Sobre la telefonía celular, se mantiene como un servicio de gran importancia para la población por su rápida conectividad, fácil traslado y acceso, así como su multifuncionalidad constantemente actualizada. En la *Figura 10* se aprecia que el 73.5% de la población indicó tener a su disposición un teléfono celular de lo cual, un 83.8% cuenta con al menos un celular común y un celular llamado “Smartphone”; un 0.8% de la población cuenta con al menos un

celular común y un celular inteligente, y un 15.4% de la población únicamente cuenta con al menos un celular común.



*Figura 10. Porcentaje de la población según uso de telefonía celular.*

*Fuente: INEGI, ENDUTH 2018.*

Teniendo en cuenta la información expuesta anteriormente, es viable la construcción de un prototipo aplicando la domótica en el contexto del IoT, que es un control inalámbrico o automático sin necesidad de conocer la distancia y en la cual, emplean tecnología sofisticada de internet al conectar objetos de la vida diaria con sensores incrustados en ellos y, además de ser controlados y proporcionar información al usuario [4] por algún medio como, el Smartphone. Gracias a esto el mercado de productos dirigido a las mascotas va en crecimiento y representan un nicho por lo cual invertir. En la actualidad existen diversos tipos de dispensadores los cuales tienen muchas funcionalidades (como cámaras, sensores, audio, grabación, música, etc.), pero la mayoría de ellos con un precio muy elevado en el mercado.

Por lo tanto, el dispensador “Pet-home ALBA” que se presenta como propuesta en este proyecto estará constituido por un servidor, un circuito electrónico y una aplicación móvil haciendo que este prototipo sea de bajo costo y accesible a todas aquellas personas que tienen un perro o un gato facilitándole el suministro alimenticio según la información de la mascota del usuario, además de mostrar la cantidad de alimento en el dispensador.



## **1.4. Objetivos.**

### **1.4.1. Objetivo general.**

Diseñar y construir un prototipo de dispensador remoto de alimento para mascotas basado en la Internet de las cosas, con la finalidad brindar nuevas herramientas que permitan mejorar la alimentación de las mascotas que no pueden ser asistidas físicamente.

### **1.4.2. Objetivos específicos.**

- Investigar porción de alimento para consumo de cada mascota en libros, revistas veterinaria y en empaques de alimento para una dosificación adecuada.
- Diseñar y crear la base de datos, para una dosificación de alimento adecuada para las mascotas, información del dispensador y clientes.
- Realizar un análisis de los diferentes componentes electrónicos tomando en cuenta las características y funcionalidades, para el desarrollo del dispensador.
- Determinar los componentes electrónicos y diseñar el circuito, para ser implementado al dispensador en la dosificación de alimento.
- Almacenar la información proveniente del dispensador para su posterior visualización.
- Desarrollar la aplicación móvil, para controlar de forma remota el dispensador.
- Realizar las pruebas por parte del servidor, la aplicación móvil y el dispensador en un ambiente real con la finalidad de comprobar una correcta funcionalidad.

## **1.5. Hipótesis.**

Es posible construir un dispensador remoto de alimento para mascotas basado en el Internet de las cosas para mejorar la alimentación de las mascotas a partir de herramientas de bajo costo.

## **1.6. Alcances y limitaciones.**

### **1.6.1. Alcances.**

El prototipo será probado en mascotas perros y gatos en la región de Misantla, Veracruz, se realizan las pruebas de funcionalidad a través de una aplicación móvil para controlar el dispensador dosificable de alimento.

El usuario de la aplicación:

- Iniciar sesión con nombre de usuario y contraseña.
- Programar las horas en las que desea dispensar el alimento, en caso de que se encuentre en un lugar sin internet o datos móviles.
- Registrar a su mascota e ingresar los datos que se les solicita.
- Dispensar el alimento cuando el usuario quiera.
- Ver el nivel de alimento restante en el dispensador.

Dispensador de alimento y bebida:

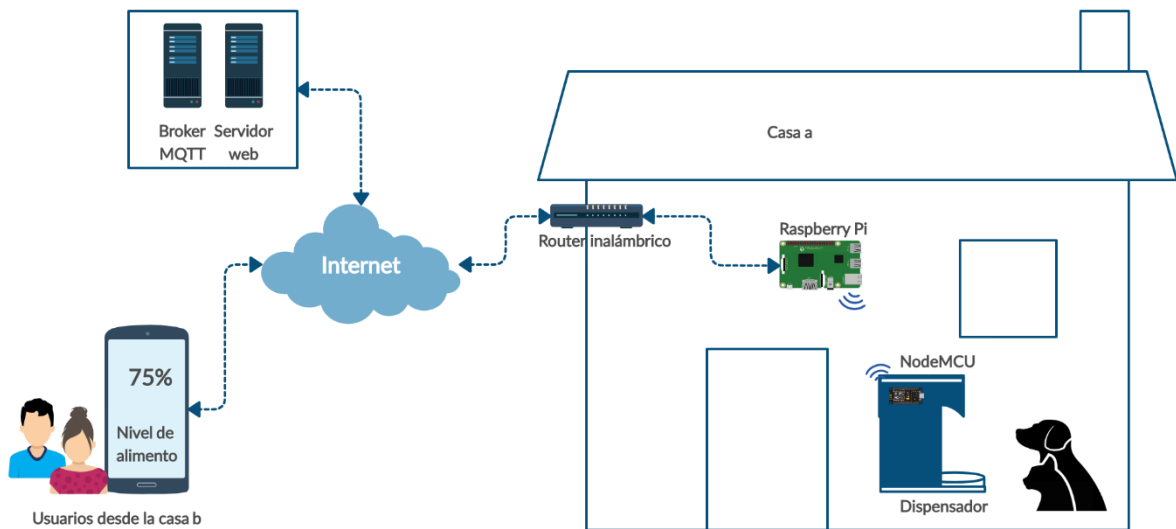
- Protocolo de comunicación IoT.
- Contenedor de alimento con una capacidad para almacenar hasta 4-5 kg.
- Fuente de energía: Siendo el principal por corriente eléctrica.
- Contará con dos modos de conexión: Conectado a WiFi; el cual enviará cierta información al usuario y modo pre-programado; donde se establecen las horas o la cantidad de alimento a libre disposición.
- Guardar datos de manera local.

### **1.6.2. Limitaciones.**

- Dispensador orientado únicamente a mascotas como perros y gatos.
- Apto sólo para alimento de tipo seco.
- Dispensador que abastece a una sola mascota.
- Dispensador que solo funciona conectado a luz eléctrica.
- Depende del material el dispensador se adapta a interior o exterior.

## 1.7. Propuesta de solución.

En la *Figura 11*, se muestra que el proyecto consiste en el diseño de un sistema versátil, utilizando plataformas y herramientas de desarrollo como un servidor que se encuentre comunicándose mediante un protocolo de internet a un controlador que proporcione conectividad a los diferentes sensores y actuadores empleados que ejecutarán diferentes tareas. La programación de las tareas por los diferentes componentes conectados al controlador se realizará en un entorno de desarrollo integrado siendo que los sensores encargados de registrar los datos del nivel de alimento y enviarlo al servidor y los actuadores que suministrarán el alimento a la mascota. Por otra parte, el usuario mediante una aplicación móvil en que accede por un nombre de usuario podrá registrar el número de dispensador, configurar los datos de la mascota y con base a esa información indicar la cantidad adecuada de alimento por porción, y configurar los horarios para suministrar el alimento a su mascota; de tal forma que pueda monitorear y controlar el prototipo que se encuentra en su casa o departamento para alimentar a su mascota.



*Figura 11. Propuesta de solución: Pet-Home ALBA.*

*Fuente: Elaboración propia*

## Capítulo II. Análisis de los fundamentos.

### 2.1. Marco histórico.

#### 2.1.1. El internet de las cosas.

Es una base de datos objeto-relacional se pueden usar los lenguajes SQL y PL/SQL para manipular datos de objetos y datos relacionales. Diseñada para almacenar y recuperar datos de objetos igual que sucede con los datos relacionales, utilizando el SQL como método estándar de comunicación con la base de datos [5].

El *Internet de las cosas*, aunque es tendencia y de importancia técnica, social y económica es un concepto que a pesar de muchos años no cuenta con una definición concisa, sin embargo, fue en 2009, Kevin Ashton, profesor del MIT por aquel entonces que acuñó este término en una presentación realizada en el Procter & Gamble (P&G) en 1999.

*“Si tuviéramos ordenadores que supieran todo lo que hay que saber sobre las cosas - utilizando los datos que recogen sin nuestra ayuda- podríamos rastrear y contar todo, y reducir enormemente el desperdicio, la pérdida y el coste. Sabríamos cuándo hay que reemplazar, reparar o retirar las cosas, y si están frescas o han pasado su mejor momento.” [6].*

Lo anterior ha originado que se realicen muchas investigaciones y proyectos en este ámbito por lo cual, se han ido obtenido a lo largo del tiempo diferentes definiciones sobre lo que significa el IoT, a lo cual podríamos decir que son: *“Dispositivos incrustados en objetos utilizados en la vida cotidiana desde una playera, una cafetera hasta un refrigerador, etc que tienen la capacidad de almacenar y comunicarse entre sí por la red Internet para así poder procesar y distribuir la información para una toma de decisiones además de poder controlarlos remotamente”*. Siendo el término del IoT cada vez más popular para descubrir escenarios en los que la conexión del internet y la capacidad de cómputo se extiendan a una variedad de objetos y dispositivos que pretendan facilitar la vida cotidiana y en el cual se puede observar que existen numerosos ámbitos de aplicación de esta tecnología y que sirven

para mejorar diferentes aspectos de nuestras vidas como lo son en la domótica que se define como un sistema capaz de automatizar todos los aparatos de una vivienda o edificación, electrodomésticos que se comunican con el usuario los cuales pueden avisar cierta información a través de un dispositivo y tendencias de uso de los consumidores como poder detectar si un corredor tiene o no los zapatos adecuados para los trayectos que realiza.

### **2.1.2. El dispensador.**

Los dispensadores de alimento surgen de la idea inicial de las máquinas expendedoras, una máquina expendedora es aquella que proporciona bebidas, aperitivos o golosinas, entre otros productos a los consumidores, su objetivo principal es vender sin la presencia de un dependiente para cobrar y despachar los artículos, de forma periódica un usuario o empleado repone los productos y recoge el dinero en forma de monedas o billetes, aunque esta última es poco frecuente, con el avance de la tecnología ya se puede pagar en algunas dispensadoras con tarjeta monedero, tarjeta de crédito o teléfono móvil. Las posibilidades de las máquinas expendedoras son amplias, suelen adaptarse de acuerdo a las necesidades de los usuarios para vender libros, cigarros, bebidas e incluso para suministrar alimento para mascotas.

Los dispensadores de alimento para mascotas fueron diseñados para facilitar el cuidado de la mascota al momento de suministrar el alimento, ahorra tiempo y así como las máquinas expendedoras solo necesitan que sean rellenadas una sola vez de forma periódica, estas funcionan de forma mecánica cuyo funcionamiento es sin intervención de ningún mecanismo eléctrico, son sencillas, y presentan algunas limitaciones, también existen las electrónicas las cuales implementan las nuevas tecnologías y hacen uso de componentes eléctricos para su funcionamiento, también necesitan de la corriente eléctrica o baterías para poder funcionar.

## **2.2. Marco teórico.**

### **2.2.1. Base de datos.**

Las bases de datos son colecciones de información (datos) dispuestos con el objetivo de proporcionar información a los usuarios y permitir transacciones como inserción, eliminación y actualización de datos, que se relacionan para crear un sentido y dar más eficiencia a una encuesta, un estudio organizado o la estructura de datos de una empresa, son de vital importancia, y en las últimas décadas se han convertido en la parte principal de los sistemas de información [7].

### **2.2.2. Sistema gestor de base de datos.**

Un sistema de gestión de bases de datos (SGBD) es el nombre dado a un conjunto de programas informáticos que gestionan una base de datos. Su objetivo principal es evitar la manipulación directa por un usuario de una base de datos y establecer un marco estándar para que los datos sean organizados y manipulados y tengan una interfaz estándar para que otros programas puedan acceder a la base de datos. En el mercado hay muchos SGBD, en las bases de datos relacionales los sistemas más populares son: Oracle, SQL Server, MySQL y PostgreSQL [7].

#### **2.2.2.1. Oracle.**

Es una base de datos objeto-relacional se pueden usar los lenguajes SQL y PL/SQL para manipular datos de objetos y datos relacionales. Diseñada para almacenar y recuperar datos de objetos igual que sucede con los datos relacionales, utilizando el SQL como método estándar de comunicación con la base de datos [5].

#### **2.2.2.2. PostgreSQL.**

Es un sistema de gestión de base de datos objeto-relacional, distribuido bajo licencia BSD y con su código fuente disponible libremente. Es uno de los sistemas de gestión de base de datos junto con MySQL de código abierto más potente del mercado [8].

### **2.2.2.3. MySQL.**

MySQL es un sistema de base de datos libre, de código abierto, rico en funciones para los usuarios novatos, y más sencillo que otros sistemas con características similares como PostgreSQL. Es importante destacar que, en la actualidad en el mercado, MySQL está más valorado en el segmento de desarrollo web [7].

### **2.2.3. Servidor web.**

#### **2.2.3.1. Wamp.**

WAMP es un acrónimo que significa Windows, Apache, MySQL y PHP. Es una pila de software que significa que se instala Apache, MySQL y PHP en su sistema operativo (Windows en el caso de WAMP). Aunque puede instalarse por separado, generalmente están agrupados y también por una buena razón. Lo que es bueno saber es que WAMP deriva de LAMP, para sistemas operativos basados en LINUX. El acrónimo de WAMP significa:

- **W:** Significa Windows.
- **A:** Significa Apache. Apache es el software del servidor responsable de servir las páginas web cuando solicita que vea una página. Apache le otorga su solicitud a través de HTTP y le muestra el sitio.
- **M:** Significa MySQL, el trabajo de MySQL es ser el sistema de gestión de bases de datos para su servidor. Almacena toda la información relevante, como el contenido de sus sitio, los perfiles de usuario, etc.
- **P:** Significa PHP. Es el lenguaje de programación que se utilizó para escribir WordPress. Actúa como pegamento para toda esta pila de software. PHP se ejecuta junto con Apache y se comunica con MySQL [9].

#### **2.2.3.2. Laragon.**

Laragon es una suite de desarrollo para PHP que funciona sobre Windows diseñado especialmente para trabajar con Laravel. Similar a otras herramientas como Xampp o Wampp, Es un entorno de desarrollo universal portátil, rápido y potente para PHP, Node.js, Python, java, Go, Ruby. Es rápido, liviano, fácil de usar y fácil de extender. Laragon es ideal para crear y administrar aplicaciones web modernas, se centra en el rendimiento: fue

desarrollado en torno a la estabilidad, la simplicidad, la flexibilidad y la libertad. Es muy ligero y su núcleo binario en sí es inferior a los 2 MB y utiliza menos de 4 MB de RAM cuando se está ejecutando. Laragon no utiliza los servicios de Windows, contiene el propio *servicie orchestration* el cual gestiona los servicios de forma asíncrona y sin bloqueo, por cual las cosas se ejecutan de manera rápida y sin problemas [10].

#### **2.2.3.3. Lamp.**

LAMP, que significa Linux, Apache, MySQL y PHP es una robusta combinación de aplicaciones impulsadas por el poderoso sistema operativo Linux. La arquitectura abierta de cada uno de estos elementos permite una integración fluida y sin problemas entre sí y resulta en una combinación de tecnologías con un historial probado de ser eficiente, segura y siempre a la vanguardia de la siempre popular Internet [11].

#### **2.2.4. Software de diseño.**

##### **2.2.4.1. Adobe Illustrator.**

Millones de artistas y diseñadores emplean el software de gráficos vectoriales estándar del sector para crear de todo, desde bellos gráficos en el sitio web y dispositivos móviles hasta logotipos, iconos, ilustraciones para libros, paquetes de productos y afiches. Adobe Illustrator es un editor de gráficos vectoriales, en forma de taller de arte que trabaja sobre un tablero de dibujo, conocido como mesa de trabajo y está destinado a la creación artística de dibujo y pintura para la ilustración, contiene opciones creativas, un acceso más sencillo a las herramientas y una gran versatilidad para producir rápidamente gráficos flexibles cuyos usos se dan en (maquetación-publicación) impresión, vídeo, publicación en la Web y dispositivos móviles [12].

##### **2.2.4.2. Adobe Photoshop.**

El software de diseño gráfico e imágenes digitales del mundo está en el núcleo de cada proyecto creativo, desde la edición y composición de fotos hasta la pintura digital, la animación y el diseño gráfico. Usado principalmente para el retoque de fotografías y gráficos. Las herramientas de fotografía profesionales facilitan las ediciones diarias o las



transformaciones totales de imágenes, desde recortar, eliminar objetos, retocar y combinar fotos. Photoshop de hecho se ha convertido, casi desde sus comienzos, en el estándar para el retoque fotográfico, pero también se usa extensivamente en multitud de disciplinas del campo del diseño y fotografía, como diseño web, composición de imágenes en mapa de bits, estilismo digital, fotocomposición, edición y grafismos de vídeo y básicamente en cualquier actividad que requiera el tratamiento de imágenes digitales [13].

#### **2.2.4.3. Adobe XD.**

Adobe XD, es la solución de diseño UX / UI para que los diseñadores de experiencia diseñen, prototipen y compartan experiencias de usuario atractivas. XD democratiza el desarrollo de características con una plataforma API que permite a los desarrolladores y usuarios extender XD con herramientas, comandos y superficies de interfaz de usuario adicionales. Con la plataforma API de XD, se podrá:

- Automatizar tareas complejas e integrar con servicios externos.
- Agregue controles de IU nativos y personalizados para mejorar los flujos de trabajo creativos [14].

#### **2.2.5. Desarrollo multiplataforma.**

Recientemente surgieron nuevas alternativas de desarrollo de aplicaciones móviles, esto es posible gracias a la extensa comunidad de desarrolladores que existe hoy en día. Dicha comunidad ayuda a que emprendedores y compañías se unan al desarrollo móvil sin tener que pasar por las dificultades que trae consigo el desarrollo nativo. Estas nuevas alternativas han mejorado de una manera considerable el tiempo, gastos y escalamiento de desarrollo de una aplicación móvil, desde el modelo vista-controlador (MVP) hasta aplicaciones que son totalmente robustas.

Basado en lo anterior, existen tres opciones de desarrollo para crear con una sola base de código, *aplicaciones móviles multiplataforma*, así mismo distribuirlas en los dos sistemas operativos móviles más populares, los cuales son Android e iOS. [15].

#### **2.2.5.1. Aplicaciones nativas.**

Las aplicaciones nativas son aquellas desarrolladas única y exclusivamente para sistemas operativos móviles ya sea Android o iOS. En Android se hace uso del lenguaje de programación Java, mientras que en iOS se hace uso de Swift u Objective-C. Estos lenguajes de programación son los oficiales para los respectivos sistemas operativos [16] .

#### **2.2.5.2. Aplicaciones web.**

Estas aplicaciones, diseñadas para correr dentro de un navegador, se desarrollan con tecnología web (HTML, CSS y JavaScript) y cuentan con una serie de características favorables: no necesitan adecuarse a ningún entorno operativo, son independientes de la plataforma y su puesta en marcha es rápida y sencilla. Por contrapartida, sus tiempos de respuesta decaen afectados por la interacción cliente-servidor, al mismo tiempo que resultan ser menos atractivas que las aplicaciones nativas ya que no se encuentran instaladas en el dispositivo. Además, las restricciones de seguridad impuestas a la ejecución de código por medio de un navegador, limitan a estas aplicaciones que no pueden acceder a todas las capacidades del dispositivo. [17].

#### **2.2.5.3. Aplicaciones híbridas.**

Son aplicaciones desarrolladas utilizando HTML5, CSS y JavaScript, desplegadas dentro de un contenedor nativo como Phonegap/Cordova, el cual brinda acceso a las capacidades del dispositivo de una forma totalmente neutral respecto al sistema operativo. Es un modelo neutro respecto a la plataforma y con portabilidad máxima. Características de las diferentes opciones para disponer de funcionalidades especiales en los dispositivos móviles (SP y tabletas). [18]

<b>Característica</b>	<b>Página móvil</b>	<b>Aplicación Nativa</b>	<b>Aplicación Híbrida</b>
<b>Plataforma</b>	Navegadores móviles	iPhone os (iOS), Windows Mobile, BlackBerry OS, Symbian, Android.	iPhone os (iOS), Windows Mobile, BlackBerry OS, Symbian, Android.
<b>Distribución</b>	URL y códigos QR	Tiendas de aplicaciones según su plataforma.	Tiendas de aplicaciones según su plataforma.
<b>Instalación</b>	Se accede directamente y puede quedar disponible mediante un launcher en el dispositivo.	Se realiza una vez y se queda disponible.	Se realiza una vez y se queda disponible para todas las plataformas.
<b>Costo de desarrollo</b>	Menores	Mayores	Menores que las nativas
<b>Rendimiento</b>	HTML5 mejora la infraestructura de la red.	Más rápido, especialmente se requieren procesos gráficos pesados.	
<b>Acceso fuera de línea</b>	Solo algunos dispositivos mediante HTML5	Completo	Completo
<b>Usabilidad</b>	Buena	Gran cantidad de efectos amigables en la interfaz, atractivos para el usuario.	Utiliza lo mejor de lo nativo y lo mejor de la red.

*Tabla 1. Características de las diferentes opciones para disponer de funcionalidades especiales en dispositivos móviles.*

*Fuente: Elaboración propia.*

## **2.2.6. Plataforma de desarrollo móvil.**

### **2.2.6.1. Xamarin.**

Xamarin es una plataforma de desarrollo que nos permite construir aplicaciones realmente multiplataforma. La gran ventaja de Xamarin es que se puede desarrollar aplicaciones que funcionen en cualquier dispositivo móvil (sea iOS o Android) con el mismo código, escrito en el lenguaje de programación C# con el framework .NET. [19]

- Las interfaces de usuario que se desarrollan con Xamarin son nativas.
- El código se comparte entre todas las plataformas.
- Las interfaces se pueden desarrollar mediante XAML.
- Open source y en constante innovación.
- Está basado en Mono.

### **2.2.6.2. Ionic.**

Ionic es un Framework para crear aplicaciones híbridas, está basado en JavaScript o lo que es lo mismo, para hacer una aplicación híbrida. Con Ionic solo se tiene que controlar HTML, CSS y JavaScript [20]. Básicamente lo que está haciendo es:

- Crear una aplicación web enfocada a dispositivos móviles, a sus resoluciones y al modo de interactuar con ellos.
- Se están utilizando librerías de JavaScript para interactuar con el Hardware, por ejemplo, la cámara o el acelerómetro.
- Por último, se está empaquetando todo como si se tratara de una aplicación nativa.

Cuando se utiliza lo que se está consumiendo realmente es una aplicación nativa que inicializa una página web con las características mencionadas. El encargado de que la aplicación web pueda tener características de las nativas es Apache Cordova, este es el proyecto que permite a Ionic las características nativas del dispositivo usando JavaScript.

### **2.2.6.3. Flutter.**

Flutter es un SDK de aplicaciones móviles, para poder crear aplicaciones de alto rendimiento y alta fidelidad para iOS y Android, a partir de una única base de código. Es el nuevo Framework de Google que utiliza Dart como lenguaje de programación para crear interfaces

nativas de alta calidad en iOS y Android en un tiempo record. Flutter funciona con código existente, es utilizado por desarrolladores y organizaciones de todo el mundo, es gratuito y de código abierto. Características principales:

- **Rápido desarrollo:** Hot Reload, en milisegundos para dar vida a la aplicación. Utiliza un amplio conjunto de widgets totalmente personalizables para construir interfaces nativas en minutos.
- **Interfaces de usuario expresiva y flexible:** Rápidamente libera funciones con un enfoque en las experiencias nativas de usuario final. La arquitectura en capas permite una personalización completa, lo que resulta en una renderización increíblemente rápida, diseños expresivos y flexibles.
- **Rendimiento nativo:** Los widgets de Flutter incorporan todas las diferencias críticas de cada plataforma, como lo es el Scrolling, navegación, iconos y fuentes para proporcionar un completo rendimiento nativo tanto en iOS como en Android [21].

### **2.2.7. Entornos de desarrollo.**

#### **2.2.7.1. Android studio.**

Este entorno de desarrollo fue presentado en el año 2013 en el *Google I/O* y paso a tomar la batuta como la herramienta, este IDE fue desarrollado por *Google*, compañía propietaria de Android así que es nombrado el IDE oficial de desarrollo para aplicaciones *Android*. Está basado en *IntelliJ* así que cuenta con diferencias notables en comparación con *Eclipse*, esto no hace ni más difícil ni más sencillo el desarrollo de una aplicación, en realidad cambian algunas cosas como la compilación, uso de librerías o paquetes externos, pero el código de una aplicación en su mayoría funciona de la misma forma sin importar el IDE que se esté utilizando [22].

#### **2.2.7.2. Visual Studio Code.**

Visual Studio Code ha sido la primera herramienta de desarrollo multiplataforma en la familia Microsoft Visual Studio que se ejecuta en Windows, Linux y macOS. Es una herramienta centrada en el código, que facilita la edición de archivos de código y sistemas de proyectos basados en carpetas, así como la escritura de aplicaciones web y móviles

multiplataforma en las plataformas más populares, como Node.js y .NET Core. con soporte integrado para una gran cantidad de idiomas, alcanzando rápidamente una definición de tipo y mucho más. El código de Visual Studio se basa en un marco de trabajo de Electrón para crear aplicaciones multiplataforma con tecnologías nativas, y combina la simplicidad de un potente editor de código con las herramientas que un desarrollador necesita para soportar el desarrollo del ciclo de vida de la aplicación, incluidos los depuradores y la integración de control de versiones basada en Git [23].

#### **2.2.8. Arquitectura de software.**

Una arquitectura de software se encarga de estructurar, diseñar y proyectar algo, para que pueda soportar cualquier eventualidad que se pueda presentar, específicamente una arquitectura de software trabaja en dos formas:

- Indica la estructura, funcionamiento e interacción entre las partes del software, muestra los archivos que están dentro de un proyecto, el código, etc. Y como va a interactuar cada archivo con otro, y como se estará comunicando en el código del proyecto, indica estructura en las partes del software.
- Permite tener el código organizado para que todo un equipo pueda entenderlo.

Tener una arquitectura permite exponer el código para que cada persona puede trabajar desde su rol la parte del proyecto que le corresponde [24].

##### **2.2.8.1. Vanilla.**

En esta arquitectura la lógica y la vista se colocan dentro de los elementos del código, su principal beneficio es que es simple y autónoma. Conectado en cualquier parte de la aplicación, recuperará y renderizará datos. Esta arquitectura es fácil de aprender, pero en realidad rompe con uno de los principios SOLID que deben seguir todas las arquitecturas. Single Responsibility la cual dice que una clase debe tener solo una responsabilidad, en este caso la vista y el controlado son dos responsabilidades que están cayendo en una misma clase [24].

### 2.2.8.2. Scoped Model.

En esta arquitectura cuando un elemento cambia de estado se reconstruye el árbol completo, en realidad utilizar esta arquitectura no es tan conveniente pues lo que realmente se requiere o sería lo ideal es reconstruir solo el elemento que está cambiando y no los demás elementos. Esta arquitectura es buena porque cumple con el principio de Single Responsibility pues separa la lógica del negocio de la UI, pero en general el mantenimiento de este se vuelve complejo por la grande dependencia entre los elementos, ya que debe controlar muchos casos para lograr el efecto que se quiere en la aplicación [24].

### 2.2.8.3. BLoC.

BLoC (Bussines Logic Components) es un patrón de diseño que ayuda a separar la lógica del negocio de la interfaz gráfica, esencialmente se concentra en dos partes, la lógica del negocio y la interfaz gráfica. Fue creada por Paolo solares Cong Hui son Googles y diseñaron esta arquitectura específicamente para Flutter, pero en general la arquitectura de software y el patrón de diseño es el mismo. La arquitectura BloC se centra en separar la vista (todos los elementos de la aplicación, toda la interfaz gráfica como botones, cards, textos, etc.), y la parte lógica (Bussines Logic) la va a separar en otra capa. [24].

#### View (UI Screens)

La parte de las vistas contiene toda la interacción con las vistas, se organizará en screens y en elementos como botones, textos, inputs, etc.

Solo se concentra la interacción con las vistas.

#### BLoC

Lógica del negocio, aquí se definen los casos de uso de la aplicación, si un usuario tiene varias acciones por ejemplo la autenticación. Toda vista que tenga una lógica de negocio deberá tener un componente BloC.

Repository	Se contendrán las clases que se conectan a una fuente de datos, se pueden tener varias fuentes de datos, esta capa permite switchear las fuentes de datos.
Data / Model	En esta capa están los modelos que nos ayudan a manejar los datos. PODO(Plain, Old , Dart Object)

*Tabla 2. Arquitectura de software BLoC.*

*Fuente: Elaboración propia*

#### **2.2.9. Lenguaje PHP.**

PHP es un Lenguaje de programación libre bastante popular, especialmente adecuado para crear sitios web e incrustarse en HTML. Este lenguaje se utiliza para generar páginas dinámicas, es decir, aquellas cuyo contenido cambia según ciertas circunstancias, por ejemplo, dependiendo de la información contenida en una base de datos, de lo que ingrese el usuario o de una búsqueda realizada. A continuación, se mencionan algunas ventajas de las más importantes:

- Puede conectarse con muchos motores de bases de datos, y se destaca su compatibilidad con algunos de los más utilizados, por ejemplo, MySQL y PostgreSQL.
- El código que creamos en PHP será invisible al navegador, ya que es ejecutado por el servidor, esto lo convierte en una opción bastante segura y confiable.
- Es posible aumentar el potencial de PHP mediante módulos o extensiones; además, podemos acceder a una completa documentación en su web oficial.
- Es un lenguaje libre, por lo que resulta accesible para todos [25].



### **2.2.10. Protocolos para el internet de las cosas.**

Los protocolos de comunicación poseen un papel vital dentro del desarrollo de proyectos que requieren comunicación constante con el servidor o con los dispositivos, es por eso que se desarrollaron protocolos diseñados específicamente para facilitar la transmisión de información de manera rápida y ligera.

#### **2.2.10.1. REST API**

REST (Representational State Transfer) es un estilo de arquitectura para sistemas hipermedia distribuidos. REST es un estilo híbrido derivado de varias arquitecturas orientadas a la red. Las restricciones establecidas son:

- **Interfaz uniforme:** Esta restricción define que la interfaz existente entre clientes y servidores debe mantenerse igual en todos los nodos, lo cual permite que pese al desenvolvimiento individual de cada nodo la información se transfiera de igual manera permitiendo conectividad todo el tiempo.
- **Sin estado:** Para minimizar el ancho de banda ocupado por una sesión abierta convencional, el sistema REST implementa un envío único de información en el que se dan todas las indicaciones necesarias al nodo receptor para el adecuado tratamiento de la información sin necesidad que este último tenga que comunicar el estado del proceso de la información.
- **Cacheable:** Las comunicaciones pueden o no almacenar un caché de las mismas para ser utilizadas en futuras transacciones minimizando el tiempo de procesamiento.
- **Cliente servidor:** El cliente y el servidor poseen estructuras internas distintas, centradas a la tarea de cada uno de estos.
- **Sistema en capas:** Rest permite la implementación de servidores intermediarios para mejorar la escalabilidad del sistema activando el balance de carga [26].

#### **2.2.10.2. MQTT**

MQTT es un protocolo completamente creado en 1999 por el Dr. Andy Stanford-Clark de IBM y Arrlen Nipper de Arcom. Es un protocolo abierto, de fácil implementación, además sencillo y ligero. Es ideal para responder a las necesidades del IoT dado que está adaptado para utilizar un mínimo de espacio en el ancho de banda, lo cual lo hace ideal para ser usado en redes inalámbricas, MQTT es ampliamente utilizado en una gran variedad de artefactos tanto industriales como particulares. Entre las principales características de MQTT se pueden notar:

- Bajo consumo energético.
- Rápido y ligero para el envío y recepción de información.
- Requiere pocos recursos para su adecuado funcionamiento.

MQTT está basado en una estructura suscriptor/publicador donde el servidor (suscriptor) llamado broker recibe los datos de los publicadores y en caso de ser necesario, enviará datos a los publicadores únicamente bajo petición del broker. Ningún mensaje puede jamás poseer un tamaño superior a 256 Mb [26].

## **2.3. Estado del arte.**

Actualmente existen distintos trabajos que han brindado a la sociedad una alternativa para el cuidado alimenticio de las mascotas, estos proyectos utilizan distintas tecnologías, mecanismos y sensores. A continuación, se muestran en detalle los proyectos.

### **2.3.1. Dispensadores automáticos**

#### **2.3.1.1. Dispensador automático de alimento para mascotas**

En Cuenca, Ecuador, Icaza Daniel *ét al* [27] dispensador que ofrece al usuario las opciones de configurar la cantidad de alimento dependiendo del tamaño de la mascota y escogiendo dos horarios al día para suministrar el alimento. La estructura consta de un tanque de almacenamiento para 5kg de alimento seco, la configuración y la cantidad de alimento con la que cuenta el tanque se visualiza a través de un display. Su modo de trabajo principal consiste estar conectado a energía eléctrica, pero, en caso de haber un corto cuenta con un banco de baterías que respalda la energía.

#### **2.3.1.2. Programmable Pet Feeder**

Tianjin, China, TG Berhan *ét al*. [28] Dispensador programable que es controlado por un microcontrolador, y al cual se le conecta una pantalla LCD para la visualización de datos, un zumbador para alertar a la mascota durante la hora de la comida o cuando el alimento está dispensado, una mesa giratoria que se divide en cuatro secciones diferentes para la cantidad de alimento que, con la ayuda de un motor paso a paso y la elección del usuario, se dispensarán en el momento deseado.

#### **2.3.1.3. Happy Pet Dispensador electrónico para mascotas de agua y comida**

Lima, Perú, A. G, Giggiola del Pilar *ét al*. [29] dispensador automatizado que cuenta con dos compartimientos uno que suministra alimento en determinado tiempo y otro para el agua que caerá de forma automática, cada compartimiento podrá ser extraíble individualmente. El dispensador cuenta con tres botones para programar las raciones de comida hasta en tres diferentes horarios diferentes lo cual se hace con un timer, un grabador de voz que se activa cuando se suministra la ración de comida y una pantalla LCD medirá la carga actual de la

batería y contar con un cargador para recargarlo en cualquier enchufe de la casa o departamento.

#### **2.3.1.4. Automatic Pet Feeder**

Mumbai, India, K. Aasavari y G. (Mohite), Vaishali [30] dispensador automático que implementa un circuito electrónico que consiste en un Arduino Uno al cual conectan componentes como: sensor de proximidad para detectar a cierta distancia a la mascota y dispensar el alimento, un sensor de peso y un servomotor SG90 con un grado angular de 0-180° que consiste como un sistema de bloqueo para controlar cuánta comida se libera cuando se abre o desbloquea el bloque.

#### **2.3.1.5. Dispensador automático de alimento seco para mascotas caninas**

Irapuato, Guanajuato, M. R, Alberto *ét al.* [31] dispensador de alimento para perros con censado y alarma de nivel de alimento, permite al usuario ingresar la edad y tamaño de la mascota, para determinar la porción de alimento a suministrar. La estructura consiste en un contenedor de madera que almacena entre 7-8kg de alimento y un circuito electrónico que incorpora una pantalla LCD, un teclado numérico, un sistema de reloj de tiempo real para suministrar el alimento a la hora especificada; estas horas de alimentación están prefijos para cuatro veces y otro para dos veces al día, un indicador luminoso para indicar bajo nivel de alimento los cuáles son controlados por un microcontrolador PIC18F4550.

#### **2.3.1.6. Diseño de dispensador automático de alimento para mascotas usando una celda**

Colombia, L. –A, Nelson Andrés *ét al.* [32] dispensador de alimento y agua, que dan prioridad al horario de suministro y la cantidad de alimento por ración a perros y gatos de razas pequeñas. El funcionamiento del dispensador integra un circuito electrónico un microcontrolador al que conectan una pantalla alfanumérica en la que se visualiza la configuración realizada por medio de un teclado, un reloj de tiempo real la cual se selecciona la fecha y hora a suministrar el alimento, un servomotor que dispensa el alimento a un dispositivo basado en Galgas extensiométricas que permite medir la cantidad de la porción

de alimento a suministrar, además de que el usuario puede grabar su voz en el prototipo que avisará a la mascota que el alimento y el agua ha sido suministrado.

### **2.3.2. Dispensadores basados en el IoT.**

#### **2.3.2.1. Smart dog Feeder design using Wireless communication, MQTT and Android Client**

Tangerang, Indonesia, Vania *ét al.* [33] El dispensador es controlado por una aplicación móvil Android en el cual se establece el tiempo de alimentación, la porción a dispensar, controlar procesos, actualizar el stock de alimento, contar con un historial de la alimentación y el tiempo en el que el alimento fue dispensado, y registrar o anular el registro del dispositivo. El circuito electrónico consiste en un Arduino Uno al que están conectados un lector RFID-RC522 para autenticar, una celda de carga para medir la porción, un servomotor para controlar la abertura de la dispensación, un RTC para establecer el horario de alimentación y un interruptor de para detectar la tapa sobre el recipiente; el Arduino Uno espera recibir órdenes del módulo NodeMCU que a su vez recibe de la aplicación móvil tal comunicación es llevada a cabo por el protocolo MQTT.

#### **2.3.2.2. Desarrollo de un prototipo para dispensar alimento y agua para mascotas con tecnología IOT monitoreado y controlado a través de un dispositivo móvil**

Quito, Ecuador, A. A., Sandra Nathaly [34] dispensador para mascotas su estructura consta de dos tanques para almacenar alimento y agua se encuentra basado en el IoT su circuito consta de una Raspberry Pi y un Arduino conectando una cámara de video permitiendo al usuario monitorearlo y controlarlo remotamente mediante un dispositivo móvil a través de Internet, los datos que pueden ser monitoreados es el nivel de alimento y agua en el dispensador. El protocolo IoT utilizado para la comunicación del circuito y la aplicación es MQTT como broker Mosquitto.

#### **2.3.2.3. The study and Application of the IoT in Pet System**

Taipéi, China, C.-M Own *ét al.* [35] proyecto que comprende en un recipiente de alimentación con subdivisiones que almacena solo alimento seco e implementa un módulo

WSN que consta de un contador de tiempo proporcionando los horarios de alimentación programada que abre o cierra la cubierta, un mecanismo de bloqueo para mantener la unidad en su lugar y un módulo de reproducción de sonido, el dispensador es accionado por un sensor de proximidad infrarrojo que consiste en mandos de distancias IR integrado en el collar de la mascota que detecta su presencia y abre la tapa para el acceso al alimento.

#### **2.3.2.4. Control domótico basado en el protocolo MQTT**

Valladolid, de la C. C., Javier, [36] proyecto que consiste en una mini-PC GIGABYTE como dispositivo domótico que se conecta a Internet a través de WiFi y el cual se comunica con un módulo NodeMCU que se encarga de un sensor de temperatura y un actuador periférico encargado de accionar la calefacción de la casa mediante un relé que para indicar su estado es a través de unos leds para la interactuar es por una aplicación móvil Android que se encarga de controlar la temperatura y programar modo automático de la caldera. La comunicación e intercambio de información entre el dispositivo domótico y la aplicación se realiza por el protocolo MQTT utilizando el broker Mosquitto.

#### **2.3.2.5. Monitorización de sensores con Arduino utilizando el protocolo MQTT**

Catalunya, España, C. C. Mario y G. L. Joan [37] proyecto caracterizado por M2M que consisten en un sensor de temperatura/humedad y un led conectados a un Arduino Uno que se encarga de enviar la información obtenida del sensor y el led a una Ethernet Shield broker MQTT: Cloud MQTT al cual se accede desde la web o móvil desarrollada en IoT MQTT Dashboards que hará de cliente y se suscribirá a los Topics correspondientes de manera que pueda ser controlado el encendido y apagado del led así como obtener información del estado del led y el nivel de la temperatura y humedad.

### **2.3.3. Dispensadores a control remoto**

#### **2.3.3.1. Diseño e implementación de un sistema dispensador de alimento para mascota, controlado vía ethernet**

D.F., México, P. T, C. Laura [38] dispensador de alimento para mascotas, como gatos y perros de raza pequeña, controlado vía Ethernet para el envío de información por la red PAN

por medio de una página web dinámica. El funcionamiento del dispensador es por un circuito electrónico que consiste en una ethernet shield para la manipulación de datos de forma remota que esta enlazada a un Arduino Uno y al cual está conectado un servomotor para abastecer el alimento es manipulado por una página web en donde se visualiza los parámetros que se desean controlar representados mediante indicadores de colores: 1) Morado, para alimentar una mascota chico, 2) Azul, alimento para un mascota mediano y 3) Rosa, alimento para un mascota grande, a lo cual, cada botón proporciona una ración de comida considerado por el tamaño de la mascota.

#### **2.3.3.2. Arduino-based Food and Water Dispenser for Pets with GSM Technology Control**

Manila, Filipinas, B. –Jr., Angelo A. *ét al.* [39] dispensador que implementa un circuito electrónico siendo una fuente de alimentación que suministra la energía al módulo GSM Shield que envía un mensaje al dueño para indicarle que el dispensador está inicializado y lista para funcionar. El dispensador consiste en un contenedor de comida y agua para su funcionamiento comienza con el dispositivo móvil enviando un SMS al módulo GSM que activará al servomotor el cual permite suministrar el alimento a la mascota y una válvula solenoide para controlar el flujo de agua, el equipo notificará al propietario y enviará un SMS que la alimentación de la mascota se llevó a cabo.

#### **2.3.3.3. Dispensador automático de comida para mascotas, programable y controlado remotamente**

Santiago de Cali, Colombia, L. Q, John David y R. A, Daniel [40] dispensador de comida para mascotas, electrónico, automático y controlado remotamente. La estructura del dispensador es especial para alimentar perros y gatos de pequeño y mediano tamaño, cuenta con un tanque donde almacena la comida durante un tiempo determinado. Se implementó un circuito siendo el módulo GPRS encargado de manejar la interfaz local los periféricos asociados son una pantalla LCD, un RTC y un teclado procesan la información obtenida de los sensores infrarrojos que indica el nivel de alimento y una celda de carga ubicada en el

plato que monitoreaba la cantidad de alimento dispensada mientras para la visualización de los datos.

#### **2.3.3.4. Low-Cost IoT Design and Implementation of a Remote Food and Water Control System for Pet Owners**

Estambul, Turquía, Ugur Baran Asaner and Armagan Elibol [41] proyecto que se compone de tres partes: unidad de ejecución, servidor y aplicación móvil. El dispensador que proporciona control de agua y alimentos para mascotas y se controla mediante una aplicación móvil en Android que muestra la cantidad de comida y agua en las últimas 10hrs que se visualizan en gráficas. El servidor una Raspberry Pi tiene el rol de conexión entre otros componentes, ya que administra las solicitudes y respuestas entre la unidad ejecutora y la aplicación móvil, de igual forma proporciona autenticación para utilizar el sistema.

#### **2.3.3.5. Automatización y asistencia remota para cuidado de mascotas**

Valencia, España, C., Jeremie [42] proyecto donde el circuito electrónico consiste en un módulo HX711 que trabaja con una celda de carga para calcular el peso del alimento suministrado y un motor para generar movimiento de tal manera mover los engranes hasta dispensar el alimento conectados a un Arduino Nano que es controlado por una Raspberry Pi Zero w implementado como servidor en NodeJS que se comunica por API REST al cliente móvil donde el usuario se registra, registra la mascota y las porciones de alimento para ofrecer un mejor control en su alimentación, y asimismo verificar el estado del dispensador.



## Capítulo III. Desarrollo.

En este apartado del documento se lleva a cabo una descripción más precisa del sistema, mostrando cada paso de los procesos para su implementación. En la Figura 12, se muestra el cronograma de actividades, el cual indica las fechas en las que se debe realizar cada actividad escrita.



Figura 12. Cronograma de actividades.

Fuente: Elaboración propia

### 3.1. Metodología.

La metodología propuesta para el desarrollo de aplicaciones móviles se fundamenta en la experiencia de investigaciones previas en aplicaciones móviles, la ingeniería de software y las metodologías ágiles. La metodología se encuentra enmarcada en cinco fases como se muestra en la Figura 13, denominadas: análisis, diseño, desarrollo, pruebas de funcionamiento y análisis de los resultados. A continuación, se describe cada una de las actividades que intervienen en el desarrollo de la propuesta.

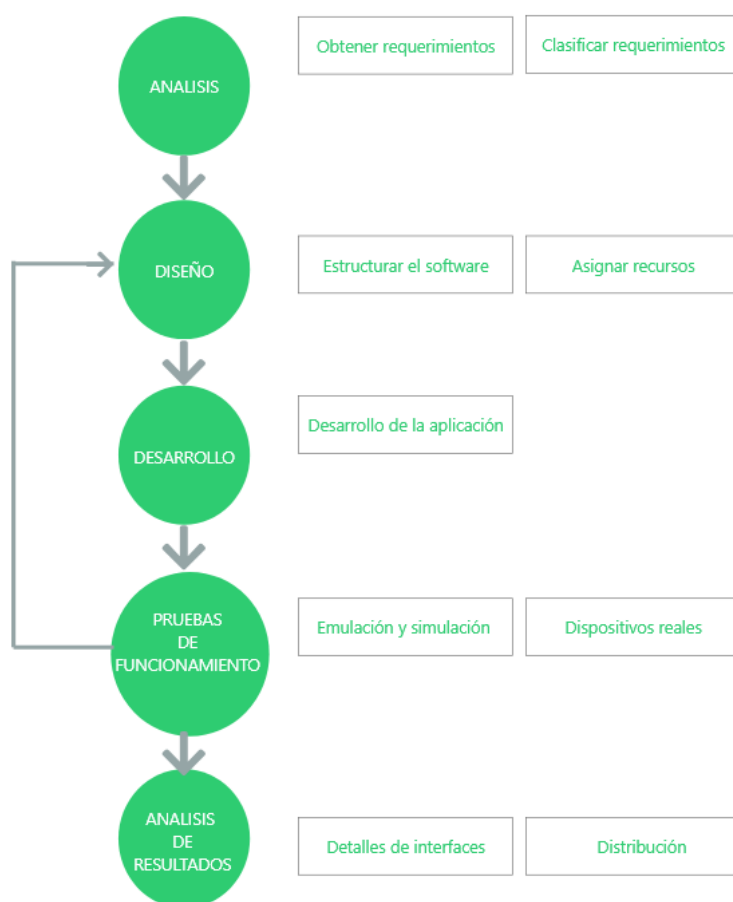


Figura 13. Metodología propuesta para el proyecto.

Fuente: Elaboración propia

## **Análisis:**

En esta fase se analizan las peticiones o requerimientos de las personas o entidad para la cual se desarrolla el servicio móvil, el propósito es definir las características del mundo o entorno de la aplicación. Se realizan dos tareas: obtener requerimientos, clasificar los requerimientos.

- **Obtener requerimientos:** se sugiere hacer una serie de entrevistas al cliente, para que manifieste los síntomas del problema o necesidades que se pretenden solucionar con las tecnologías móviles, o simplemente, para que señale las características que debe tener la aplicación.
- **Clasificar los requerimientos:** una vez identificados los requerimientos que debe tener el software, se procede a clasificarlos. Dichos requerimientos se pueden clasificar en funcionales y no funcionales.

## **Diseño:**

El objetivo de esta etapa es plasmar el pensamiento de la solución mediante diagramas o esquemas, considerando la mejor alternativa al integrar aspectos técnicos, funcionales, sociales y económicos. A esta fase se retorna si no se obtiene lo deseado en la etapa prueba de funcionamiento. Se realizan dos actividades en esta fase: estructurar el software, definir tiempos y asignar recursos.

- **Estructurar el software:** se deben utilizar algunos diagramas de Modelado de Lenguaje Unificado, Unified Modeling Language (UML), casos de uso, diagramas etc. según las necesidades del proyecto, modelando el sistema desde varias perspectivas
- **Asignar recursos:** se asignan los recursos para realizar cada actividad, se deben seleccionar las herramientas para el desarrollo de la aplicación móvil.

## **Desarrollo**

El objetivo de esta fase es implementar el diseño en un producto de software. En esta etapa se realizan las siguientes actividades:

- **Desarrollo de la aplicación en Flutter.**

Por qué usar Flutter para desarrollar la aplicación móvil.

- **Instalación de Flutter.**

Instalar Flutter en el computador.

- **Plugin de Flutter en Visual Studio Code.**

Agregar Plugin de Flutter en Visual Studio Code

## **Pruebas de funcionamiento**

El objetivo de esta fase es verificar el funcionamiento de la aplicación en diferentes escenarios y condiciones; para esto se realizan las siguientes tareas:

- **Emulación y simulación:** se realizan pruebas simulando el escenario y emulando el dispositivo móvil, explorando todas las utilidades y funciones de la aplicación, introduciendo diferentes datos, inclusive erróneos, para medir la funcionalidad y el nivel de robustez del software. Si se encuentran algunas fallas, se debe regresar a la etapa de codificación en la fase de desarrollo para solucionar los problemas, si las pruebas son satisfactorias se procede a la etapa de pruebas con dispositivos reales.
- **Dispositivos reales:** deben hacerse pruebas de campo en equipos reales para medir el desempeño y el rendimiento del aplicativo. Si se encuentran fallas en el tiempo de ejecución, si el software no cumple con los requerimientos especificados.

## **Análisis de los resultados**

Esta etapa consta de dos pasos, el detalle de las interfaces de la aplicación y su distribución.

- **Detalles de las interfaces:** Se detallará de forma muy minuciosa las interfaces finales de la aplicación móvil, su funcionamiento y como debe de ser utilizado, se puede entender que es similar a un manual de usuario.
- **Distribución:** se define el canal de comercialización de la aplicación, con el propósito de adecuar la aplicación al medio de distribución.

### 3.2. Análisis.

#### 3.2.1. Obtener requerimientos.

##### 3.2.1.1. Diseño de la investigación y procedimiento.

Esta es una investigación de tipo aplicada orientada a demostrar la validez de ciertos métodos, instrumentos y técnicas para dar respuesta a un problema con innovación tecnológica y producto. Según el desarrollo del tema estudiado es transversal.

Su procedimiento consiste en:

1. La primera etapa consistió en realizar la elaboración de la encuesta para su posterior aplicación.
2. Se procede a investigar en los datos del INEGI sobre la población del municipio de Misantla.
3. Por último, se aplicó la encuesta a personas que cumplen con los criterios de inclusión donde se realizará una visita a las personas y otra en línea, la información quedará en anonimato.

##### 3.2.1.2. La población y la muestra.

La población de este estudio es de tipo finita ya que está conformada por un número limitado de elemento la cual se aplica al municipio de Misantla, entidad Veracruz de Ignacio de la Llave que está conformada por un total de 62,919 habitantes [43] de las cuales se tomarán en cuenta a 30441 personas que se encuentran en un rango de edad entre los 20 y 59 años, de sexo tanto femenino como masculino.

La muestra será obtenida referencialmente a través de la fórmula de muestra finita:

$$n = \frac{N * Z_a^2 * p * q}{(e^2 * (N - 1)) + Z_a^2 * p * q}$$

*Figura 14. Ecuación para obtener del tamaño de la muestra.*

*Fuente: Elaboración propia*

En donde:

$n$ = Tamaño de la muestra buscada (número de encuestas que vamos a hacer).

$N$ = Es el tamaño de la población o universo (número total de posibles encuestados) siendo 30441 personas que representan los individuos que cumplen con los criterios de inclusión para este estudio.

$Z$ = Es una constante que depende del nivel de confianza que asignemos. El nivel de confianza indica la probabilidad de que los resultados la investigación sean ciertos, para ello el investigador asigna el valor estimado de confianza a través del grado de certeza que quiera de la investigación. En la *Tabla 3* se muestra los valores de  $Z$  alfa, dado el nivel de confianza.

Nivel de confianza	99.7%	99%	98%	96%	95%	90%	80%	50%
$Z_{\alpha}$	3	2.58	2.33	2.05	1.96	1.645	1.28	0.674

*Tabla 3. Nivel de confianza.*

*Fuente: Elaboración propia*

$e$ = Error de estimación máximo aceptado se refiere a la cantidad de error de nuestro aleatorio resultado de la elaboración de una investigación, el valor es asignado por el investigador.

$p$ = Es la proporción de individuos que poseen en la población la característica de estudio.

$q$ = Es la proporción de individuos que no poseen esa característica, es decir,  $1-p$ .

La proporción de individuos que poseen en la población las características de estudio ( $p$ ) indica la probabilidad de que ocurra el evento con éxito mientras que la proporción de individuos que no poseen esa característica ( $q$ ) indica la probabilidad de que el evento no ocurra. Cuando no se conoce el valor de  $p$  o  $q$ , el investigador deberá asignar el mismo peso de ambos eventos.

En lo cual asumimos que queremos un nivel de confianza del 1.645 que pertenece al 90% mientras que consideramos que la probabilidad de éxito y fracaso es del 50%, y deseamos un

error de estimación del 10% para 30441 individuos que pertenece al tamaño de la población para esta investigación. A continuación, en la *Figura 15* se muestra el cálculo para determinar el tamaño de muestra:

Parámetro	Valores	Tamaño de muestra "n" =
N	30.441	
Z	1.645	
P	50.00%	
Q	50.00%	
e	10.00%	67.50

*Figura 15. Determinar el tamaño de muestra.*

*Fuente: Elaboración propia*

### 3.2.1.3. Criterios de inclusión y exclusión.

Quedaron incluidos en este estudio las personas que viven en el Municipio de Misantla, de ambos sexos, con edades comprendidas entre los 20 a 59 años y tanto como cuenten con una mascota, así como aquellas que no tienen una mascota, pero han dado su consentimiento para la participación de la investigación y se excluyeron del estudio las personas que no presentaron las características contempladas. En la cual se utiliza un listado de preguntas que se presenta de forma física o electrónica a fin de que sean contestadas.

### 3.2.1.4. Instrumentos.

Para la recolección de la información, como técnica principal se aplica la encuesta de forma personal y a través de Internet, el cual es elaborado por los investigadores y que se muestra en *anexos*, los cuales para una mejor obtención de la información cuenta entre dos a cuatro respuestas de elección.

La encuesta consta de nueve ítems con la finalidad de determinar:

- La cantidad de personas que cuentan con alguna mascota.
- La cantidad de personas que cuentan con internet en casa o un Smartphone.
- La cantidad de personas que estarían dispuestas a adquirir un dispensador inteligente.

### 3.2.2. Clasificar requerimientos.

#### 3.2.2.1. Requerimientos no funcionales o de información.

<b>IDENTIFICACION DEL REQUERIMIENTO</b>	<b>RI-01</b>
<b>NOMBRE DEL REQUERIMIENTO</b>	Splashscreen
<b>CARACTERISTICAS</b>	El usuario podrá visualizar el nombre de la aplicación y el nombre de la empresa que lo desarrollo, así como el logo.
<b>DESCRIPCION DEL REQUERIMIENTO</b>	No habrá necesidad de que el usuario este registrado o inicie sesión
<b>DATOS ESPECIFICOS</b>	1. Nombre de la aplicación 2. Logo 3. Nombre de la empresa 4. CircleProgrees (Automático 6 s.)
<b>INTERVALO TEMPORAL</b>	Presente

*Tabla 4. SplashScreen.*

*Fuente: Elaboración propia.*



### 3.2.2.2. Requerimientos funcionales.

<b>IDENTIFICACIÓN</b>		<b>RF-01</b>
<b>NOMBRE REQUISITO FUNCIONAL</b>		<b>INICIAR SESIÓN</b>
<b>REQUISITOS ASOCIADOS</b>		RI-01 Autenticación del sistema. RF-02 Error de datos
<b>DESCRIPCIÓN</b>		La aplicación debe dejar iniciar sesión a los usuarios
<b>PRECONDICIÓN</b>		El usuario debe estar registrado en la base de datos de la empresa.
<b>SECUENCIA NORMAL</b>	<b>PASO</b>	<b>ACCIÓN</b>
	01	El usuario deberá introducir en el campo de texto con la etiqueta “Nombre de usuario”
	02	El usuario deberá introducir en el campo de texto con la etiqueta “Contraseña”.
	03	Una vez introducido los datos, el usuario deberá dar clic en el botón de “iniciar sesión”
<b>POST-CONDICIÓN</b>		Si los datos no son correctos se mostrará un mensaje con el error.

Tabla 5. Iniciar sesión.

Fuente: Elaboración propia.

<b>IDENTIFICACIÓN</b>		<b>RF-02</b>
NOMBRE REQUISITO FUNCIONAL	ERROR DE DATOS	
REQUISITOS ASOCIADOS	RF-01 iniciar sesión	
DESCRIPCIÓN	La aplicación debe dejar iniciar sesión a los usuarios y si los datos no son correctos deberá enviar un mensaje con el error.	
PRECONDICIÓN	El usuario debe llenar los datos del formulario del inicio de sesión	
SECUENCIA NORMAL	<b>PASO</b>	<b>ACCIÓN</b>
	01	El usuario deberá dar clic en el botón “Regresar” para poder cerrar el mensaje y regresar al formulario de inicio de sesión.
POST-CONDICIÓN	Se mostrará el login del sistema	

*Tabla 6. Error de datos.*

*Fuente: Elaboración propia*

<b>IDENTIFICACIÓN</b>		<b>RF-03</b>	
NOMBRE REQUISITO FUNCIONAL		INICIO	
REQUISITOS ASOCIADOS		RI-01 Autenticación del sistema. RF-01 Iniciar sesión RF-02 Error de datos	
DESCRIPCIÓN		La aplicación deberá tener una pantalla de inicio donde muestre los datos más relevantes	
PRECONDICIÓN		El usuario deberá iniciar sesión para poder visualizar el contenido	
SECUENCIA NORMAL		<b>PASO</b>	<b>ACCIÓN</b>
		01	El usuario dará clic en el botón “Alimentar” cada vez que desee proporcionar alimento a su mascota

*Tabla 7. Inicio.*

*Fuente: Elaboración propia*

<b>IDENTIFICACION</b>		<b>RF-04</b>	
NOMBRE REQUISITO FUNCIONAL		MENU PRINCIPAL	
REQUISITOS ASOCIADOS		RI-01 Autenticación del sistema. RF-01 Iniciar sesión	
DESCRIPCIÓN		La aplicación deberá tener un menú para poder navegar en la aplicación de forma cómoda y sencilla	
PRECONDICIÓN		El usuario deberá iniciar sesión para poder visualizar el contenido	

SECUENCIA NORMAL	PASO	ACCIÓN
	01	El usuario dará clic en el icono del menú, el cual se encuentra en la parte posterior izquierda de color blanco, se desplazará la ventana del menú, el cual tiene la apariencia de una carta.
	02	La primera opción del menú es “Inicio”, cuando el usuario seleccione esta opción del menú lo va a dirigir a la pantalla principal.
	03	La segunda opción del menú es “Modo automático”, cuando el usuario seleccione esta opción, lo va a dirigir a la pantalla de modo automático.
	04	La tercera opción del menú principal es “Perfil de mascota”, cuando seleccionen esta opción del menú principal, el usuario será dirigido a la pantalla de perfil de mascota.

*Tabla 8. Menú Principal.*

*Fuente: Elaboración propia*

<b>IDENTIFICACIÓN</b>		<b>RF-05</b>
NOMBRE REQUISITO FUNCIONAL	MODO AUTOMÁTICO	
REQUISITOS ASOCIADOS	RI-01 Autenticación del sistema. RF-01 Iniciar sesión	
DESCRIPCIÓN	La aplicación deberá tener una interfaz para poder programar el dispensador de alimento de forma automática.	
PRECONDICIÓN	El usuario deberá iniciar sesión para poder visualizar el contenido, también podrá programar de forma automática las horas en las que se va a dispensar el alimento de la mascota.	
SECUENCIA NORMAL	<b>PASO</b>	<b>ACCIÓN</b>
	01	El usuario ingresara la clave de su dispensador en el cuadro de texto con la etiqueta “Clave”
	02	Una vez que el usuario ingreso la clave de su dispensador, seleccionara la primera opción que es la primera dosis a dispensar de alimento.
	03	El usuario seleccionará el icono de flecha que estará en la parte derecha de la etiqueta “Primera dosis”
	04	Aparecerá un timePicker donde el usuario deberá elegir la hora en la que desea dispensar el

		alimento a su mascota, en el mismo widget estarán los botones de “Ok”, “cancelar”
	05	El usuario deberá repetir la acción con las tres opciones más, los pasos son exactamente iguales.
	06	Una vez que el usuario seleccione todos los horarios correspondientes deberá dar clic en el botón “Guardar”, para que los cambios se queden guardados en la base de datos.

*Tabla 9- Modo automático.*

*Fuente: Elaboración propia*

<b>IDENTIFICACIÓN</b>	<b>RF-06</b>
NOMBRE REQUISITO FUNCIONAL	PERFIL DE MASCOTA(tablas)
REQUISITOS ASOCIADOS	RI-01 Autenticación del sistema. RF-01 Iniciar sesión
DESCRIPCIÓN	La aplicación deberá tener una interfaz para poder visualizar los datos de la mascota, así como unas tablas de información acerca de la buena alimentación de los perros y gatos.
PRECONDICIÓN	El usuario deberá iniciar sesión para poder visualizar el contenido, y seleccionar “Perfil de mascota” en el menú principal para poder acceder

SECUENCIA NORMAL	<b>PASO</b>	<b>ACCIÓN</b>
	01	El usuario podrá visualizar las tablas de información nutricional de las mascotas”
	02	Cuando el usuario se deslice al final de la pantalla encontrará un botón, el cual lo dirigirá al perfil de su mascota.

*Tabla 10. Perfil de mascota (Tablas).*

*Fuente: Elaboración propia*

<b>IDENTIFICACIÓN</b>	<b>RF-07</b>	
NOMBRE REQUISITO FUNCIONAL	PERFIL DE MASCOTA	
REQUISITOS ASOCIADOS	RI-01 Autenticación del sistema. RF-01 Iniciar sesión	
DESCRIPCIÓN	La aplicación deberá tener una interfaz para poder visualizar los datos de la mascota, nombre, edad, peso, etc.	
PRECONDICIÓN	El usuario deberá iniciar sesión para poder visualizar el contenido, y seleccionar “Perfil de mascota” en el menú principal para poder acceder	
SECUENCIA NORMAL		

	PASO	ACCIÓN
	01	El usuario podrá visualizar en un recuadro en la parte central de la pantalla el nombre de la mascota, la cantidad de alimento que va a dispensar.
	02	En la parte inferior derecha habrá un botón para poder dar de alta a una mascota.

*Tabla 11. Perfil de mascota.*

*Fuente: Elaboración propia*

IDENTIFICACIÓN		RF-08
NOMBRE REQUISITO FUNCIONAL	ELIMINAR MASCOTA	
REQUISITOS ASOCIADOS	RI-01 Autenticación del sistema. RF-01 Iniciar sesión RF-07 Perfil de mascota	
DESCRIPCIÓN	La aplicación deberá tener una interfaz para poder eliminar a una mascota.	
PRECONDICIÓN	Tener una mascota registrada	
SECUENCIA NORMAL	PASO	ACCIÓN
	01	El usuario deberá estar en el perfil de mascota



	02	Al hacer clic en el nombre de la mascota, se mostraran dos opciones editar y eliminar
	03	El usuario deberá seleccionar eliminar para poder dar de baja a la mascota.
	04	La aplicación móvil notificará que dicha acción esta correcta
		El usuario regresará al perfil de la mascota y se mostrarán los datos de la mascota.
POST-CONDICIÓN	Se eliminará la mascota del sistema	
EXCEPTIONES	PASO	ACCIÓN
	01	El usuario podrá cancelar la acción de eliminar a la mascota
	02	La aplicación móvil notificará cualquier problema al eliminar a la mascota

*Tabla 12. Eliminar mascota.*

*Fuente: Elaboración propia*

### 3.2.2.3. Elaboración de las tablas de porciones de alimento.

Para que el presente dispensador sea dosificable es importante saber la cantidad de ingesta de alimento para la mascota, esto es importante ya que las mascotas deben ser alimentados en cantidades proporcionales para que reciban la energía y nutrientes necesarios, pero, la cantidad de comida varía de acuerdo al tipo de mascota, así como sus características siendo: la etapa, su tamaño y en algunos casos como su edad en meses y peso. A continuación, se muestran las porciones de alimento en gramos para cada tipo de mascota siendo en la *Figura 26* para gatos y en la *Figura 27* para los perros; según información obtenida de Purina Excellent Cat Chow [44] y Dog Chow [45].

CANTIDAD DE ALIMENTO				
TABLA DE INFORMACIÓN DE PURINA EXCELLENT				
De acuerdo a las características de tu mascota				
Tabla de información para gatos				
Etapa	Tamaño	Edad en meses	Peso	Cantidad (gramos)
Gatito		3 semanas – 2 meses		25 – 45 gramos
		2 meses – 6 meses		45-60 gramos
		6 meses – 1 año		60 – 75 gramos
Adulto	Mediano		2 – 3.5 G Kg	35- 60 gramos
	Grande		3.5 – 6.5 Kg	60-110 gramos
Adulto 7+	Mediano		2 – 3.5 G Kg	35- 60 gramos
	Grande		3.5 – 6.5 Kg	60-120 gramos

*Figura 16. Porciones de alimento para gatos.*

*Fuente: Elaboración propia*


CANTIDAD DE ALIMENTO				
TABLA DE INFORMACIÓN DE PURINA EXCELLENT				
De acuerdo a las características de tu mascota:				
<div>  <div>Tabla de información para perros</div> </div>				
Etapas	Tamaño	Edad (En meses)	Peso (Kg)	Cantidad
Cachorro	Minis	Destete – 3 meses	1- 5 Kg	50- 90 gramos
		3 – 5 meses	1- 5 Kg	60-80 gramos
		5- 8 meses	1- 5 Kg	60- 80 gramos
		8- 12 meses	1- 5 Kg	45- 120 gramos
	Pequeño	3 – 5 meses	1- 5 Kg	50- 70 gramos
		5-8 meses	5-10 Kg	90- 150 gramos
		8- 12 meses	5- 10 Kg	120- 200 gramos
	Mediano		10-25 Kg	200 – 360 gramos
			25 – 35 Kg	360- 450 gramos
	Grande		35 45 Kg	450 – 540 gramos
Adulto	Gigante		45 Kg +	560 + 30g por cada 5kg adicional
	Minis		1- 5 Kg	45 – 120 gramos
	Pequeño		5-10 Kg	120 – 200 gramos
	Mediano		10-25 Kg	200 – 360 gramos
			25 – 35 Kg	260 – 450 gramos
	Grande		35 – 45 Kg	450 – 540 gramos
Senior	Gigante		45 Kg +	560 + 30 por cada 5kg adicional
	Minis		1 – 5 Kg	35 – 110 gramos
	Pequeño		5 – 10 Kg	110 – 170 gramos
	Mediano		10 – 25 Kg	170 – 310 gramos
			25 – 35 Kg	310 – 390 gramos
	Grande		35 – 45 Kg	390 – 460 gramos
	Gigante		45 Kg +	460 + 30 g por cada 5Kg adicional

Figura 17. Porciones de alimento para perros.

Fuente: Elaboración propia.

### 3.3. Diseño.

#### 3.3.1. Estructurar el software.

##### 3.3.1.1. Modelo funcional.

El modelo funcional explica de forma concreta y detallada lo que debe realizar la aplicación móvil y como debe ejecutarlo: Aplicación móvil Pet-Home:

Se inicia la aplicación y se muestra el splashscreen que dura alrededor de 6 segundos, posteriormente se realiza un login, donde el usuario ingresa los datos correspondientes para acceder y ver el resto del contenido de la aplicación. Cuando se verifique que se trata del usuario administrador (categoría que se asigna) podrá alimentar de forma manual a la mascota con solo presionar el botón que se encuentra dentro de la interfaz principal, además de ser capaz de ingresar las horas en que se dispensa las dosis diarias de alimento, así como gestionar los datos de las mascotas como son altas, bajas y modificaciones, al finalizar el usuario puede cerrar sesión.

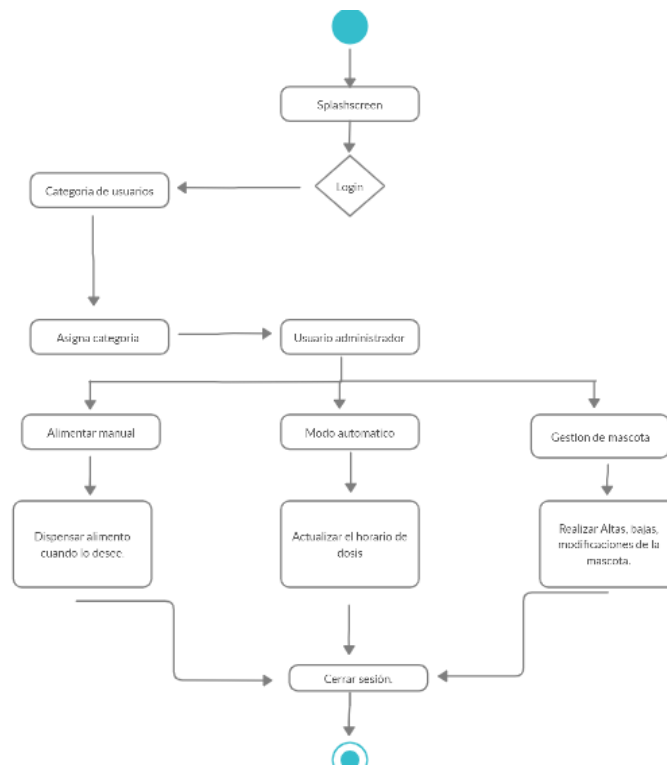


Figura 18. Modelo funcional.

Fuente: Elaboración propia.

### 3.3.1.2. Casos de uso.

Un caso de uso consiste en especificar la comunicación el comportamiento de un sistema mediante su interacción con los usuarios y/o sistemas.

- Al iniciar la aplicación el usuario cuenta con la posibilidad de Iniciar sesión.
- Una vez que el usuario inicie sesión, puede agregar, eliminar o editar los datos de su mascota.
- Configurar los horarios de las dosis de alimento a dispensar.
- Activar de forma remota el dispensador de alimento para proporcionar la dosis a su mascota siempre que el usuario lo requiera.



Figura 19. Casos de uso.

Fuente: Elaboración propia.

### 3.3.2. Asignar recursos.

#### 3.3.2.1. Herramientas para el desarrollo.

Nombre del software	Área a utilizar.
<b>Mockflow</b>	Maquetación.
<b>Flat UI Palette</b>	Colores
<b>Adobe XD</b>	Diseño UI
<b>Visual Studio Code</b>	Entorno de desarrollo
<b>Dart</b>	Lenguaje de programación.
<b>Flutter</b>	Plataforma de desarrollo

Tabla 13. Herramientas de desarrollo.

Fuente: Elaboración propia.

### 3.3.2.2. Maquetación de la aplicación.

La maquetación es una de las principales claves en el proceso de conceptualización, diseño y desarrollo de aplicaciones móviles. Elaborar un prototipo o un boceto navegable de una aplicación permite materializar de forma aproximada una idea, acercándose al resultado final, así mismo acelerar la fase de desarrollo y poder evitar problemas futuros e involucrar también a los clientes. Dentro del desarrollo de aplicaciones móviles, hay tres tipos de maquetas: *Sketch*, *Wireframes* y *Mockup*.

El tipo de maquetación que se utilizó para el desarrollo de la aplicación móvil fue *Wireframes*, ya que es una de las formas de maquetación más completas que las mencionadas anteriormente, esta forma de maquetación se hace en papel y en digital, contiene información adicional en cada una de las pantallas, por ejemplo; disposición de los botones, contenidos, llamadas a la acción, movimientos de navegación entre pantallas, esto es habitual en proyectos digitales. *Wireframes* es una representación estática, se suele entregar en formato de imagen. A continuación, se muestra el diseño de la maquetación de la aplicación móvil, la cual controlara de forma remota el dispensador, diseñada en la plataforma mockflow.

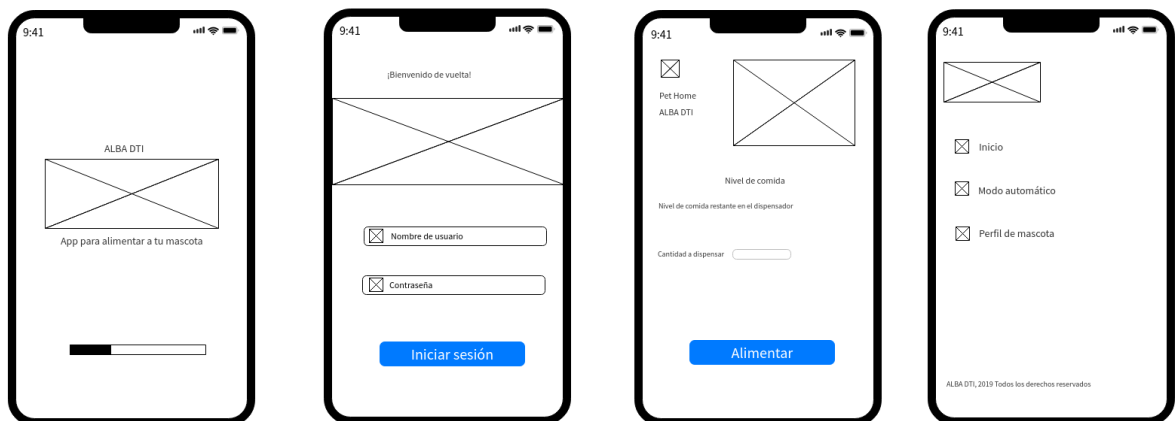


Figura 20. Maquetación Pet Home parte 1.

Fuente: Elaboración propia.

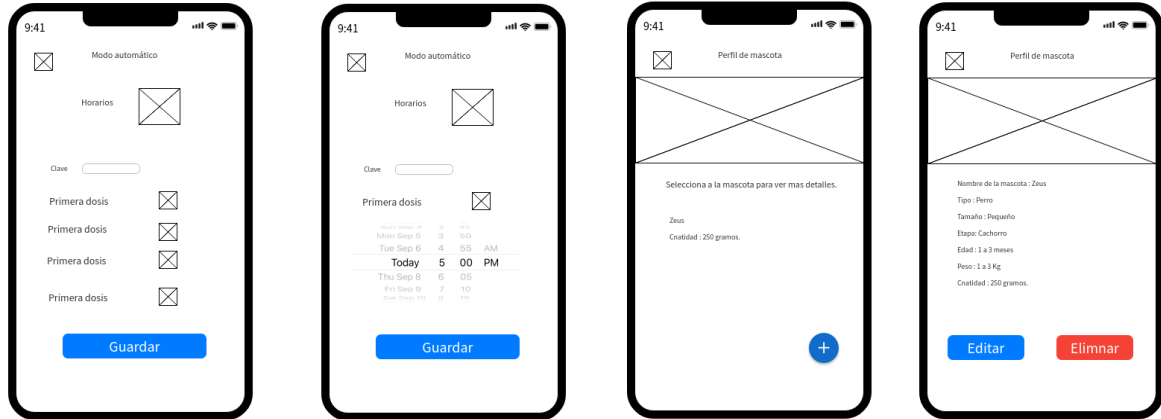


Figura 21. Maquetación Pet Home parte 2.

Fuente: Elaboración propia.

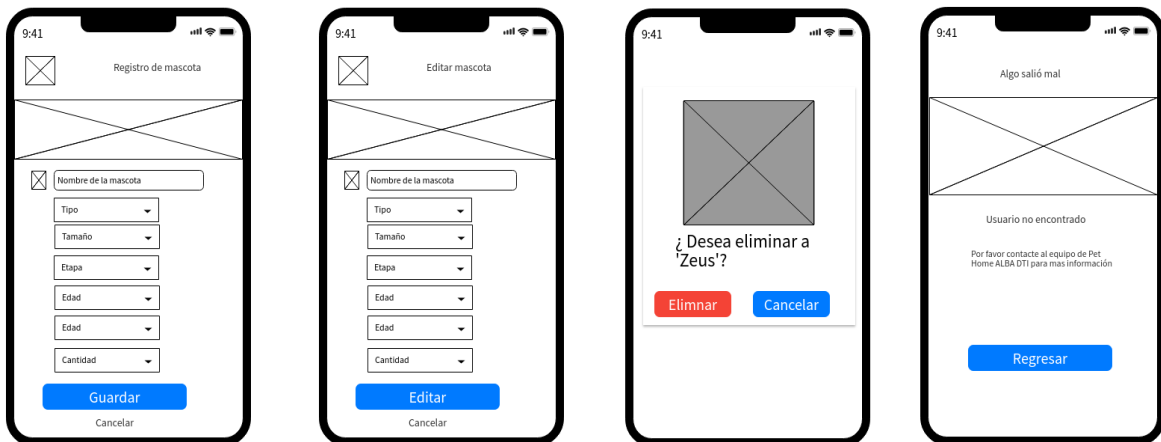


Figura 22. Maquetación Pet Home parte 3.

Fuente: Elaboración propia.

### 3.3.2.3. Selección de colores.

En la selección de colores se han usado los tonos de dos colores principales, el primero fue el azul que tiene algunas propiedades como son [46];

- El color azul permite muchas combinaciones con diferentes colores
- Es utilizado para ser relacionado con salud, tecnología o para perfiles profesionales.

En general se debe tener las siguientes consideraciones para la elección de las tonalidades de los colores [46] . Para el fondo se debe colocar un tono bastante claro que permita la lectura,

entonces se ha escogido dos tonos uno blanco cuyo identificador hexadecimal es “#FFFFFF”, el segundo tono es un lila cuyo identificador hexadecimal es “#8391DC”. · Las cabeceras o barras de menú deben tener un color de contraste llamativo para guiar la vista del usuario, es por ello que se usa el color azul que hará un contraste con el blanco y el lila, que debe tener un tono un poco más cargado para resaltar en donde sea necesario. Se ha escogido el color “#4066D3”. · Los botones y cuadros de texto deben tener colores llamativos, se ha usado de igual manera el azul, pero con tonos un poco más bajos para no cansar la vista ya que esos botones y cuadros de texto se ven constantemente en la aplicación móvil, Para el color de las letras se ha escogido el gris en general, pero también se puede usar el mismo azul donde el contraste sea adecuado. En la Tabla 14 se aprecia la paleta de los colores que se han usado.




Cabecera y barras de menú.	botones	Cuadros de texto	Botón (especial)	Fondo	Detalles.
#8391DC 	#4066D3 	#4066D3 	#E14626 	#FFFFFF 	#FBB03B 
#4066D3 		#544F51 		#8391DC 	

Tabla 14. Colores a utilizar.  
Fuente: Elaboración propia.

### 3.3.2.4. Diseño UI de la aplicación.

El diseño UI (Diseño de interfaces del usuario) es aquel que trabaja directamente con el medio en que cada persona interactúa con una aplicación, software o programa. Se asocia de forma directa en la interacción entre el usuario y el dispositivo. Entre mejor y más practica sea la interacción, más reconocido será la aplicación como una experiencia de usuario más amigable y fácil. El diseño UI de la aplicación Pet- Home fue realizada con la herramienta de diseño Adobe XD, el contenido de cada interfaz será explicado de una forma más detallada en el apartado de resultados.



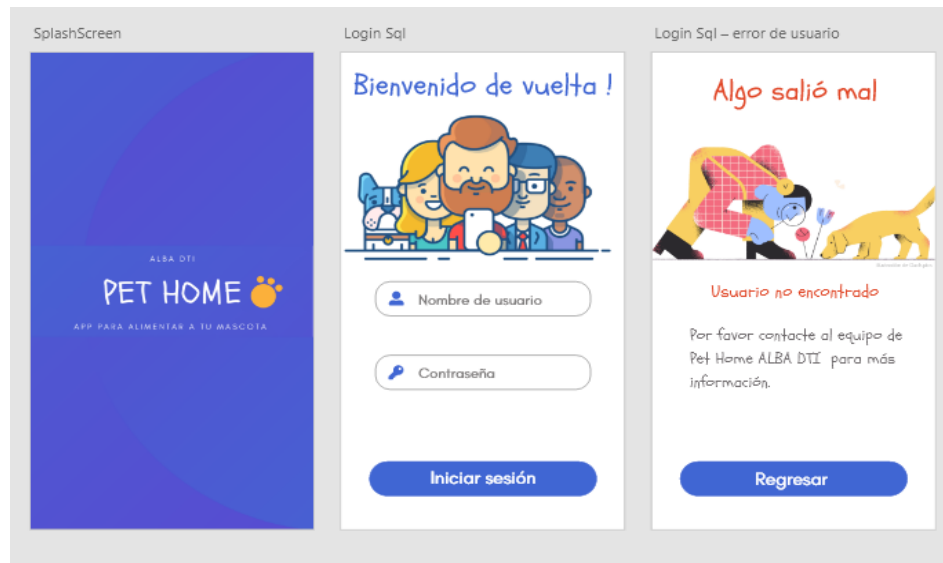


Figura 23. Diseño UI Splashscreen, Login, error de datos.

Fuente: Elaboración propia.

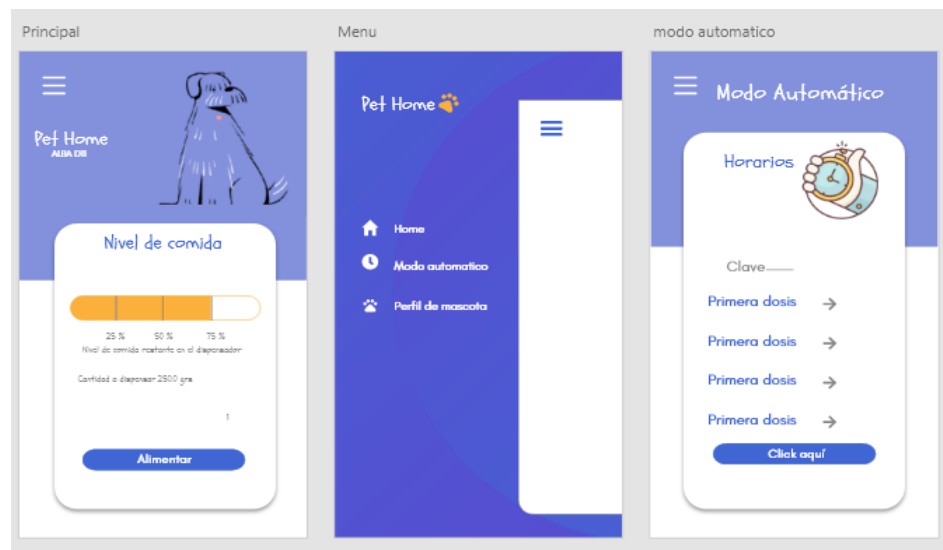


Figura 24. Diseño UI Menú y modo automático.

Fuente: Elaboración propia.

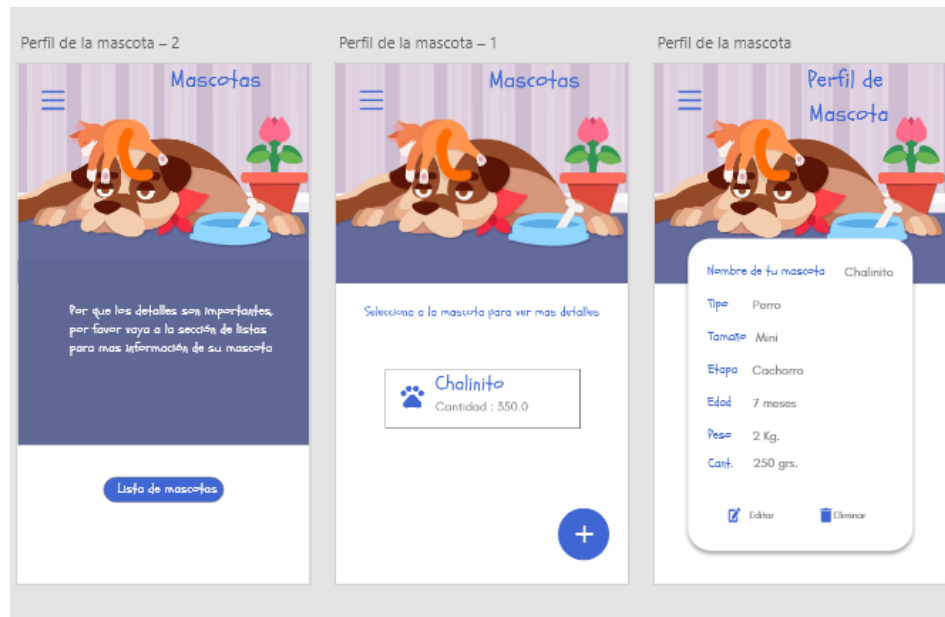


Figura 25. Diseño UI Perfil de mascotas.

Fuente: Elaboración propia.

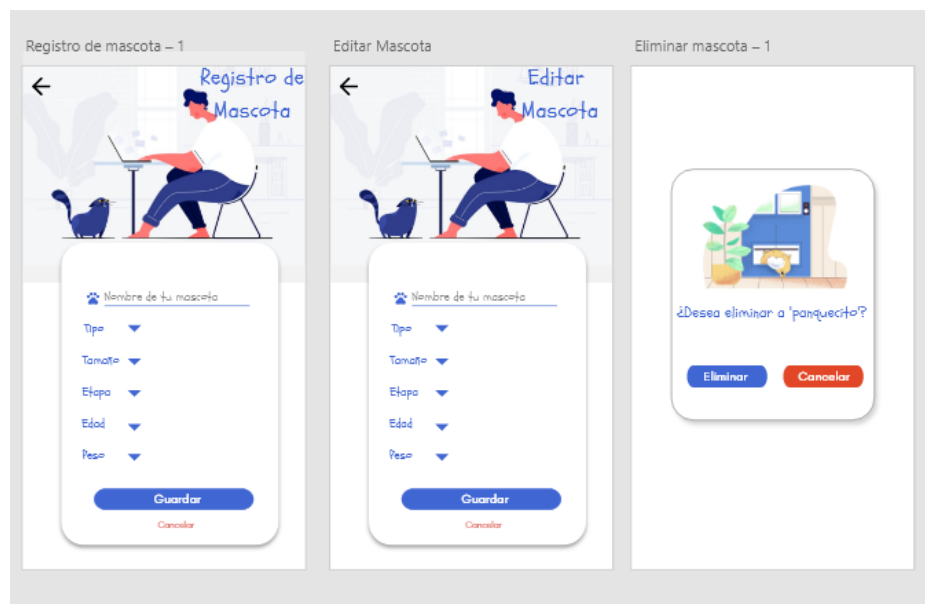


Figura 26. Diseño UI Registro, editar y eliminar mascotas.

Fuente: Elaboración propia

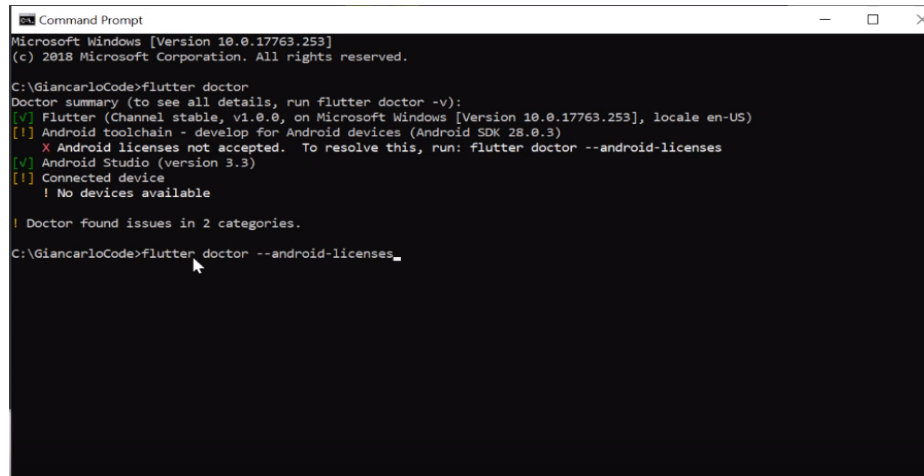
### 3.4. Desarrollo.

#### 3.4.1. Desarrollo de la aplicación en Flutter.

Flutter es un framework desarrollado por google para desarrollar aplicaciones nativas en Android y iOS, incorpora al lenguaje de programación orientado a objetos Dart, el cual es propiedad de google, en su momento fue desarrollado para competir con *JavaScript*, pero se incorporó en *Flutter*, y ha tenido buena aceptación por la comunidad de desarrolladores. La primera versión estable de este *framework* salió en diciembre del 2018, aunque ya existía desde el 2016 las versiones beta, la filosofía de *Flutter* es que todo es un *widget*, botones, cajas de texto, *list view*, etc., Destaca por su rápido desarrollo de aplicaciones, el llamado *HotReload* el cual hace que los cambios sean de segundos realizados a la aplicación una vez que está en emulación, y su rendimiento nativo, tal como lo indica su página web.

#### 3.4.2. Instalación de Flutter.

Para usar *Flutter* se descarga el SDK que proporciona la página oficial de *Flutter* [47]. Se crea una carpeta en el disco local y se ejecuta archivo *flutter\_console.bat*, y desde ahí se ejecuta los comandos de inicio, el primer comando es *Flutter doctor*, verifica el entorno y muestra un reporte del estado de la instalación de *Flutter*, y revisa la salida con cuidado de otros programas que se pudieran necesitar o tareas a realizar. Por otro lado, se ejecuta el comando *Flutter doctor -android-licenses*, para dar licencia a *Android Studio*, ya que para el desarrollo de la aplicación se puede utilizar como IDE. Algunas veces recomienda ejecutar otros comandos recomendados para actualizar Flutter.



```
Microsoft Windows [Version 10.0.17763.253]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\GiancarloCode>flutter doctor
Doctor summary (to see all details, run flutter doctor -v):
[✓] Flutter (Channel stable, v1.0.0, on Microsoft Windows [Version 10.0.17763.253], locale en-US)
[!] Android toolchain - develop for Android devices (Android SDK 28.0.3)
    X Android licenses not accepted. To resolve this, run: flutter doctor --android-licenses
[✓] Android Studio (version 3.3)
[!] Connected device
    ! No devices available

! Doctor found issues in 2 categories.

C:\GiancarloCode>flutter doctor --android-licenses_
```

Figura 27. Consola para ejecutar comandos de Flutter.

Fuente: Elaboración propia.

### 3.4.3. Plugin de Flutter en Visual Studio Code.

Como editor de código para programar en *Flutter* se utiliza *Visual Studio Code*, aunque es posible utilizar *Android Studio* también. Para ello se dirige al menú de *Visual Studio Code* y en el apartado de extensiones se busca el *plugin* de *Flutter*.

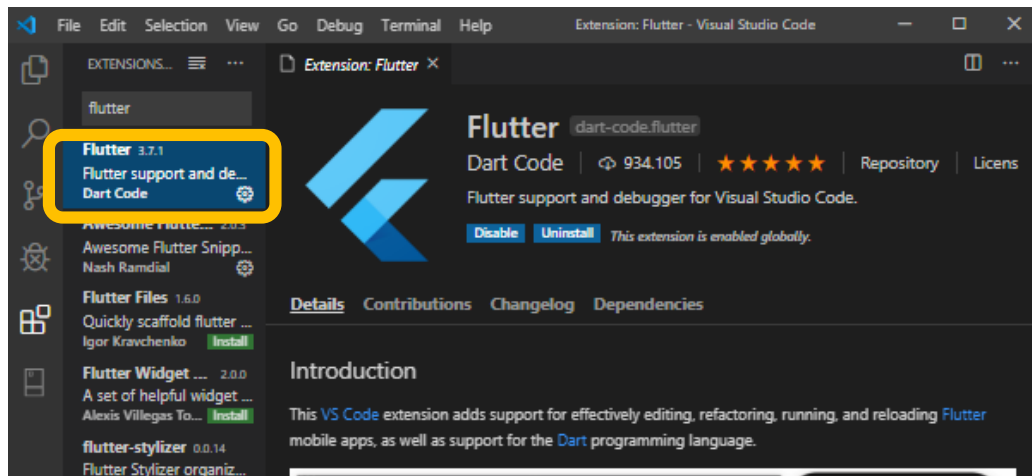


Figura 28. Agregar Plugin de Flutter.

Fuente: Elaboración propia

De esta manera el plugin de Flutter ya está instalado en nuestro editor Visual Studio Code. Para poder crear un nuevo proyecto en Flutter desde nuestro editor, solo se necesitan oprimir la combinación de teclas `Ctrl + shift + p` y se abrirá el menú emergente del editor, se seleccionará *Flutter New Project* y se le dará un nombre.

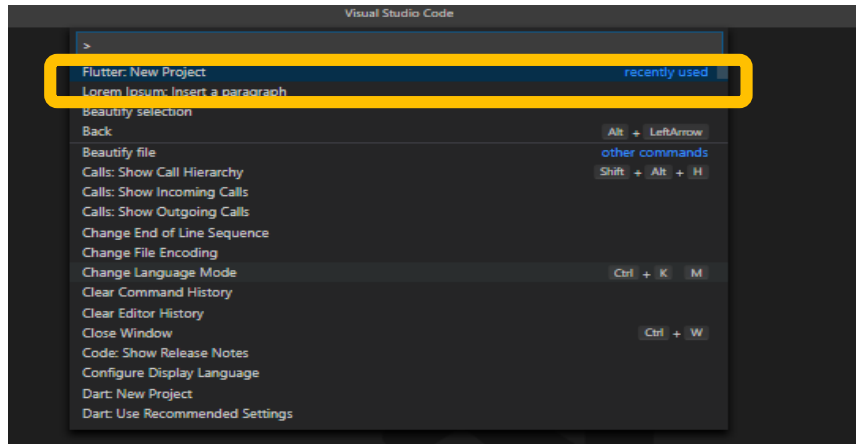


Figura 29. Iniciar un proyecto en Flutter.

Fuente: Elaboración propia

### 3.5. Pruebas de funcionamiento.

#### 3.5.1. Emulación y simulación.

Las pruebas realizadas se hicieron en un dispositivo físico, conforme se iban desarrollando las interfaces se iban probando para poder arreglar posibles errores, diseño, etc., para poder hacer la emulación en el dispositivo se tuvo que configurar el teléfono a modo “Opciones del desarrollador”.

#### 3.5.2. Dispositivos reales.

Los dispositivos donde fue probada la aplicación se muestran en la tabla:

Modelo Smartphone	Procesador	Sistema Operativo	Memoria RAM	Pantalla
<b>Huawei p20</b>	Kirin 970 con NPU	Android 8.1 Oreo – EMUI 8.1	4GB	5,8” LCD 18:7:9. FHD+ (2240 x 1080 píxeles) Full View Display 2.0
<b>Samsung Galaxy S8</b>	Snapdragon 835 / Exynos 8895	Android 7.0	4GB	5.8", 1440 x 2960 pixels

<b>LG Q6 Prime</b>	Snapdragon 435 1.4GHz	Android 7.1.1	3GB	5.5", 1080 x 2160 pixels
<b>Motorola Moto G8 Plus</b>	Snapdragon 665 2GHz	Android 9.0	4GB	6.3", 1080 x 2280 pixels
<b>Huawei Nova 5T</b>	ocho núcleos (Kirin 980)	9.0 Pie (EMUI 9.1)	8GB	6.2“(IPS, 2,340x1,080, 412ppp)

*Tabla 15. Dispositivos móviles utilizados en pruebas de aplicación móvil.*

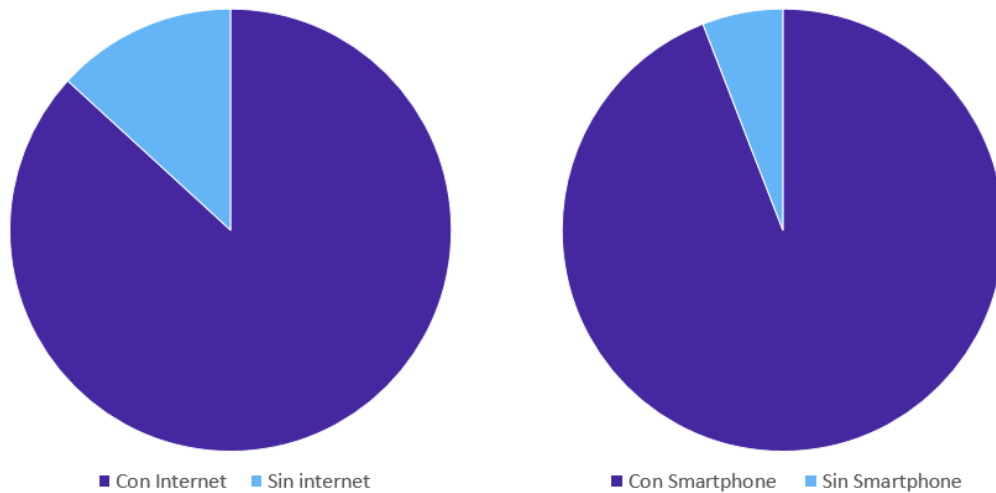
*Fuente: Elaboración propia*

## Capítulo IV. Análisis de los resultados.

### 4.1. Datos Pet-Home ALBA

A continuación, se analizan los resultados obtenidos de la encuesta pertenecientes al apartado 3.1.4. que se aplicó a las 68 personas resultado del tamaño de la muestra:

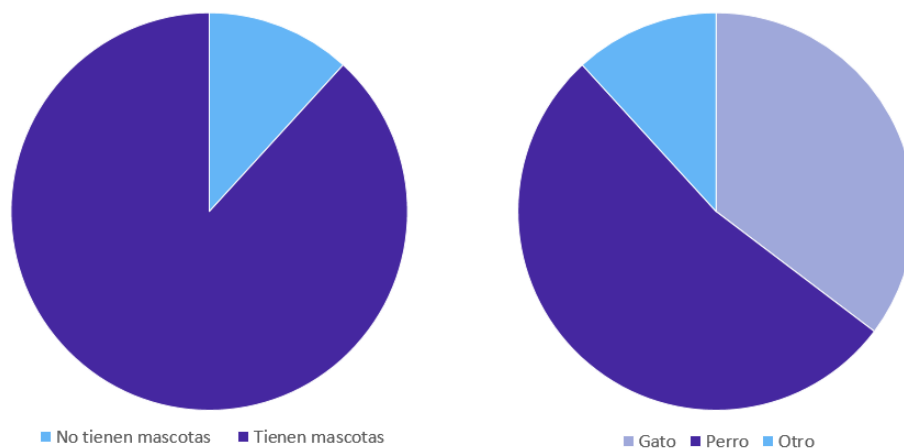
1. En la *Figura 30* se determina la cantidad de personas que cuenta con internet en sus hogares y que disponen con al menos un Smartphone, a lo cual el 86.8% de los encuestados cuentan con internet y un 94.1% poseen un Smartphone mientras que el 13.2% no tiene internet y el 5.9% no disponen de un Smartphone.



*Figura 30. Resultado de la población que cuenta con Internet y un Smartphone.*

*Fuente: Elaboración propia*

2. En la *Figura 49* se determina la cantidad de personas que cuentan con una mascota en sus hogares y de qué tipo son, sin embargo, en caso de que el individuo no tenga alguna mascota se preguntó qué tipo de mascota le gustaría tener, a lo cual, el 88.2% dice sí contar con una mascota mientras que el 11.8% indicó no tener una mascota por otra parte la mascota más popular con un 52.9% son los perros seguidos de un 35.3% que dicen tener un gato y por último un 11.8% indican tener algún otro tipo de mascota.



*Figura 31. Resultado de la población que cuentan con una mascota y que tipo es.*

*Fuente: Elaboración propia*

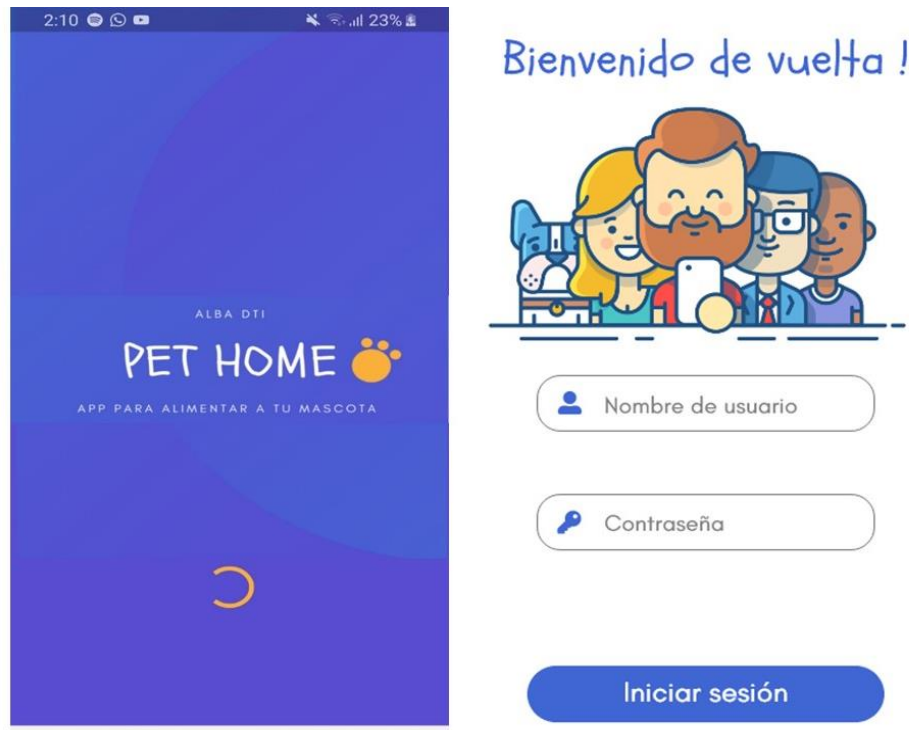
## **4.2. Aplicación móvil.**

### **4.2.1. Pantallas de inicio: Splashscreen y login.**

La primera pantalla Splashscreen se muestra cuando arranca la aplicación móvil, tiene una duración de carga de seis segundos y en la cual el usuario no interactúa de forma directa, después del tiempo establecido cambia de forma automática a la pantalla de inicio de sesión la cual pide ingresar los siguientes datos:

1. Nombre de usuario: Una vez registrado el cliente en la base de datos del Pet home ALBA-DTI se le asigna un nombre de usuario el cual se debe introducir.
2. Contraseña: Decidida por el usuario, se ingresa en el campo respectivo para tener acceso al contenido de la misma.
3. Iniciar sesión (Botón): Una vez ingresado los datos requeridos del usuario, se debe dar clic en el botón.





*Figura 32. Pantallas de inicio: Splashscreen y Login.*

*Fuente: Elaboración propia*

#### **4.2.2. Pantalla de error de datos.**

Se muestra cuando los datos ingresado en la pantalla anterior son erróneos este es la ventana / vista que surgirá, debe asegurarse que los datos ingresados sean correctos. Para volver a la pantalla de inicio de sesión, solo deberá dar clic en el botón regresar.

## Algo salió mal



### Usuario no encontrado

Por favor contacte al equipo de  
Pet Home ALBA DTI para más  
información.

Regresar

*Figura 33. Pantalla error de datos.*

*Fuente: Elaboración propia*

#### **4.2.3. Pantalla principal y menú.**

La pantalla principal, aquí es donde el usuario tendrá más interacción. En esta pantalla se muestra el nivel de alimento disponible en el dispensador, así como la opción para que el usuario ingrese la cantidad de alimento que dispensará de acuerdo a los datos de la mascota (para más información ver el apartado 4.3.5. *Perfil mascota: Tabla de información*), un botón de alimentar al cual al dar clic hace funcionar el dispensador al suministrar la cantidad de dosis de alimento a la mascota y en la parte izquierda se encuentra un icono referencia al menú, el cual se podrá navegar a diferentes pantallas dentro de la aplicación, para acceder a las opciones del menú el usuario deberá seleccionarlo y se desplazará la pantalla en efecto Card mostrando las opciones:

1. Home / inicio: Cuando el usuario seleccione esta opción del menú, lo va a redirigir a la pantalla principal.
2. Modo automático: Cuando el usuario seleccione esta opción en el menú de la aplicación podrá tener acceso a la pantalla que lo llevara al modo automático, donde podrá programar los horarios para dispensar el alimento.
3. Perfil de la mascota: Cuando el usuario seleccione esta acción del menú de la aplicación, podrá ver los datos de su mascota.

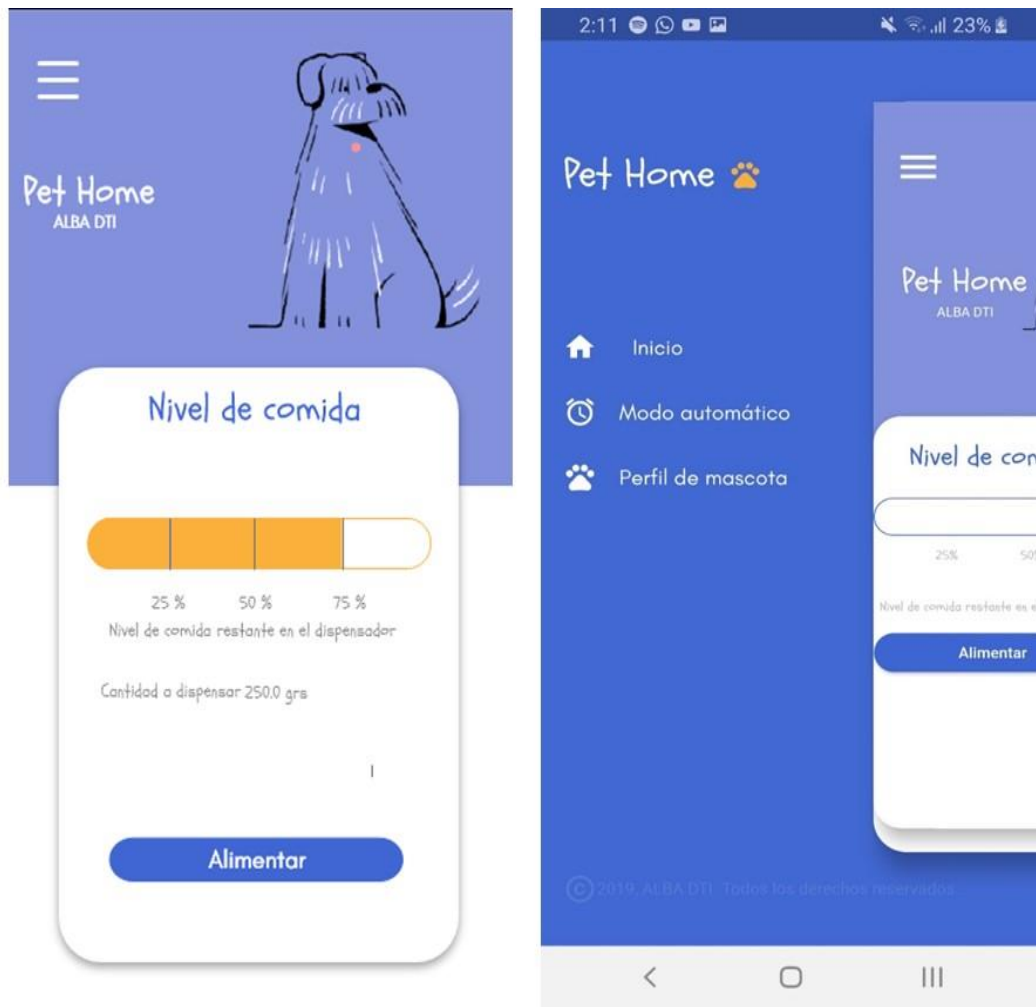


Figura 34. Pantallas después de inicio: Pantalla principal y menú.

Fuente: Elaboración propia

#### 4.2.4. Modo automático

Pantalla que tiene la función de programar el horario de dosis en la que el usuario desee dispensar el alimento de su mascota. Los campos son:

1. |Clave: El usuario debe ingresar la clave del dispensador que viene incluida en él.
2. Dosis: Primera, segunda, tercer y cuarta dosis se ejecutan los mismos pasos para configuración. El usuario da clic sobre la flecha y se mostrará un reloj en el cual se selecciona la hora y minutos para dispensar el alimento, se da clic en el botón *Ok*, que se muestra debajo del reloj.
3. Guardar: Una vez que el usuario ha seleccionado los horarios para dispensar da clic en el botón *Guardar* para almacenar los cambios efectuados.

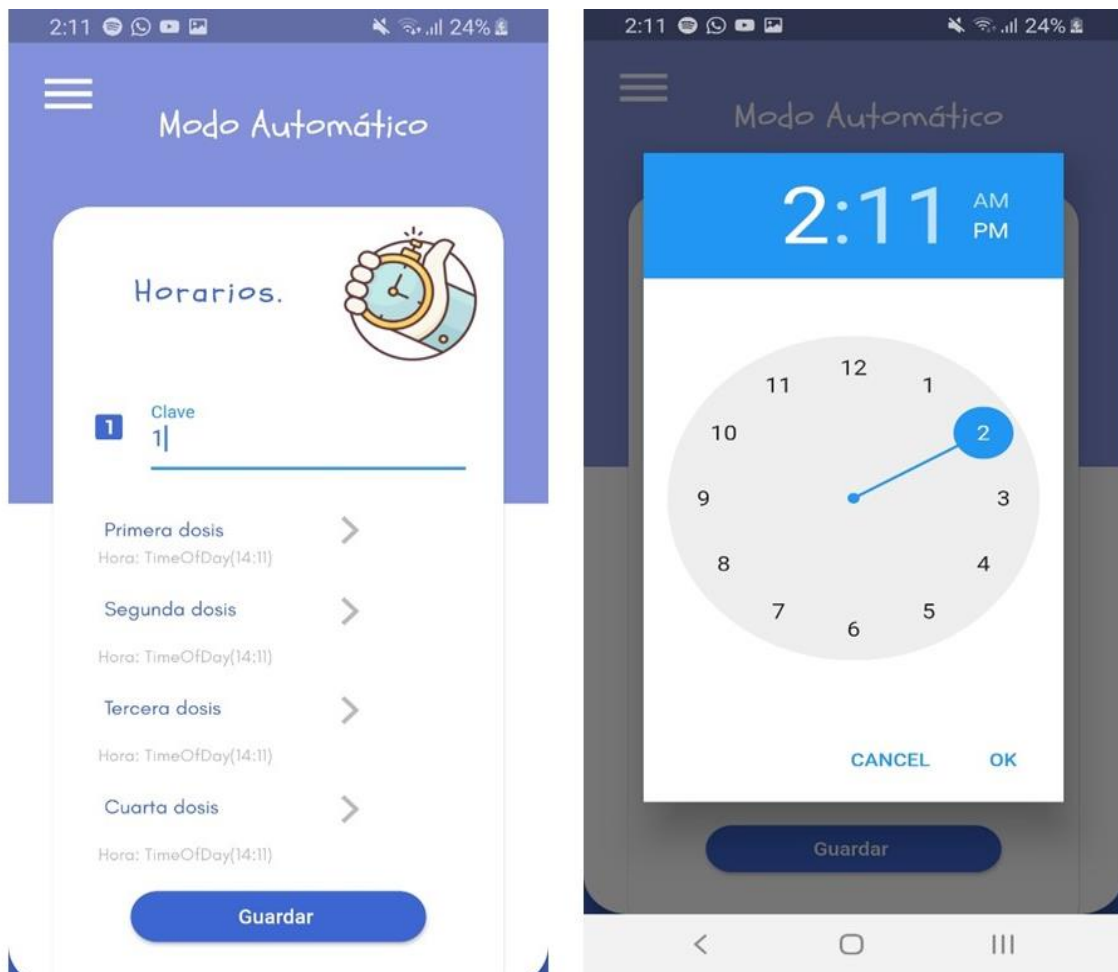


Figura 35. Pantalla de modo automático.

Fuente: Elaboración propia.

4.2.5. Perfil mascota: Tabla de información y lista de mascota

Estas pantallas surgen cuando el usuario selecciona *Perfil de la mascota*. En esta pantalla se mostrará una tabla en la cual dependiendo de las características de la mascota se recomienda la cantidad de alimento. En la parte inferior en el centro se encuentra un botón el cual llevará a las listas y detalles de la mascota que el usuario registre donde se muestra el nombre de la mascota y la cantidad de alimento que se va a dispensar, para ver los detalles de la mascota solo se debe seleccionar el nombre de la mascota, así como se muestra un botón circular en la parte inferior derecha donde él puede agregar a la mascota.

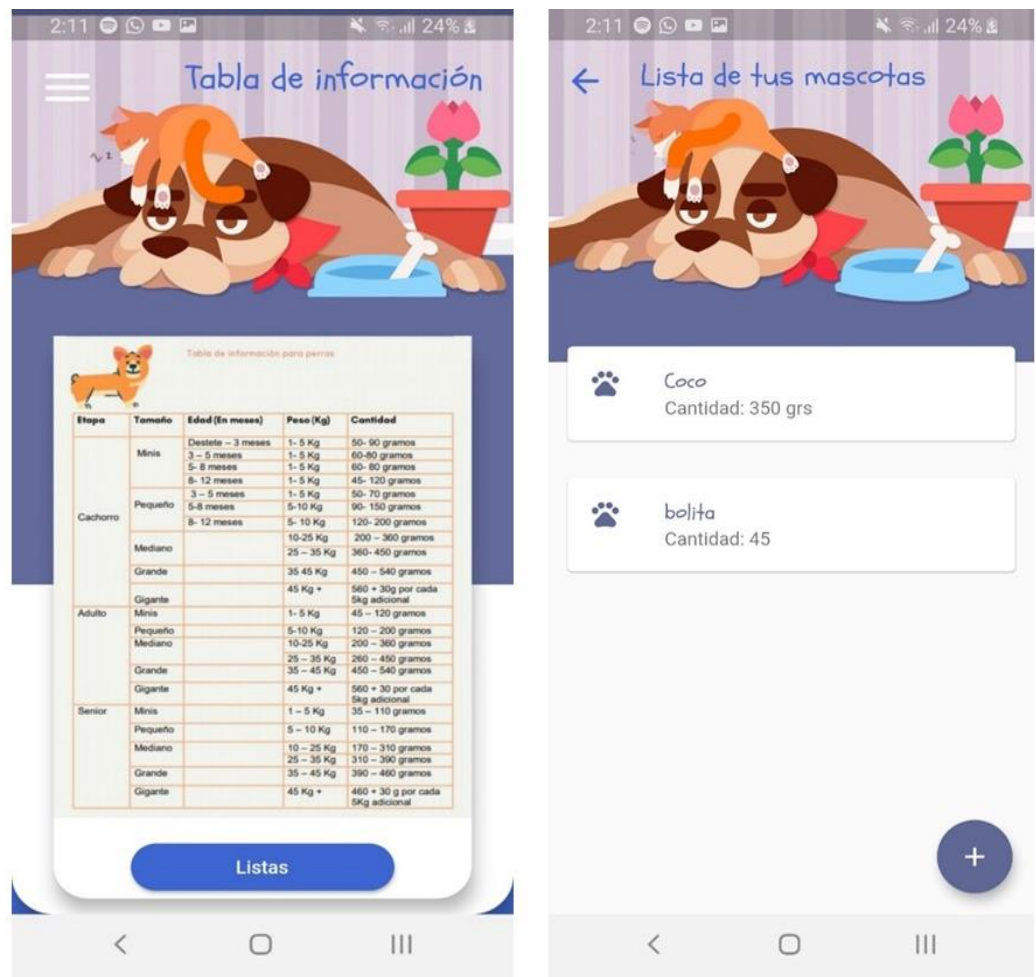


Figura 36. perfil mascota: Tabla de información y lista de mascota.

Fuente: Elaboración propia.

#### 4.2.6. Perfil mascota: Registro y detalle de la mascota

Pantalla de registro donde el usuario ingresa la información de la mascota la clave con la que se registra la mascota, nombre de la mascota, clave dispensador, tipo de mascota siendo las opciones gato y perro de acuerdo a la selección los demás datos cambiarán, tamaño de la mascota según la opción las opciones varían con respecto a la edad, de acuerdo a la edad se muestra el peso correspondiente, así como las cantidades de alimento ideales a dispensar. Por último, se cuenta con dos botones Guardar donde una vez finalizado el registro se almacenan los datos en la base de datos y el botón Cancelar el cual no guarda el registro. Algo que se debe saber es que el registro se debe hacer de forma ordenada, dato por dato y no saltado ya que las opciones van de acuerdo al dato anterior. Mientras que la pantalla de detalles de la mascota surge cuando el usuario selecciona el nombre de la mascota mostrada en la lista de mascotas. En esta interfaz se muestran los datos mostrados del registro: Nombre, tipo, tamaño, edad, peso y la cantidad de alimento. En la parte inferior se encuentran dos botones donde se podrá editar o eliminar la mascota.

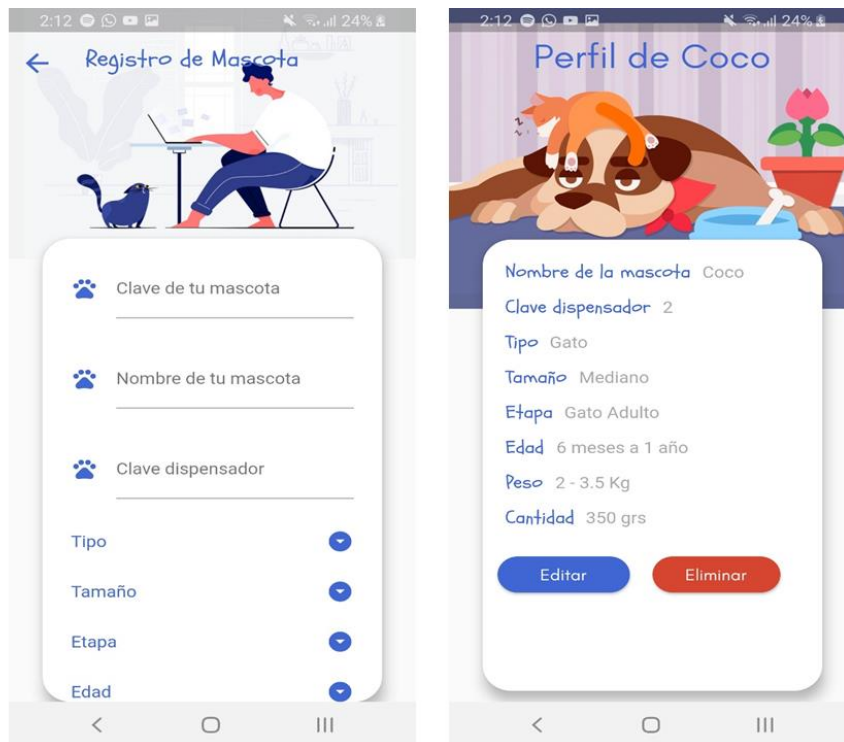


Figura 37. Perfil mascota: Registro y detalle de la mascota.

Fuente: Elaboración propia

#### 4.2.7. Perfil mascota: Editar y eliminar mascota

Cuando el usuario seleccione el botón de editar, surge en pantalla *Editar Mascota* donde se puede actualizar los datos de la mascota como el tamaño, etapa, edad, peso y la cantidad deseada a suministrar por dosis, así como poder guardar los cambios o ya sea cancelar la edición de los datos de la mascota. Pero, cuando el usuario selecciona el botón de eliminar surge una alerta en el cual muestra “¿Estás seguro de eliminar a [Nombre de la mascota]?”, el usuario debe confirmar en caso de querer eliminar o bien, cancelar la operación de eliminar.

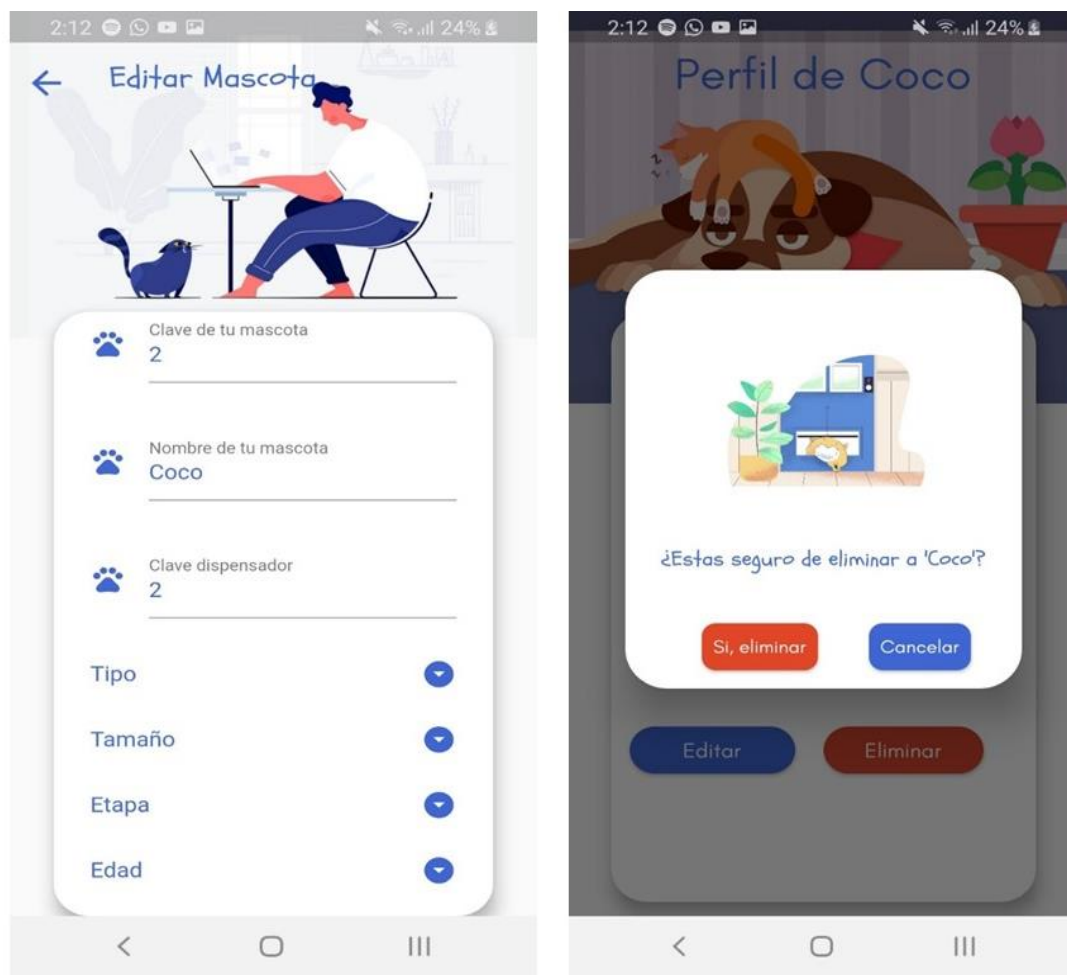


Figura 38. Perfil mascota: Editar y eliminar mascota.

Fuente: Elaboración propia

## Conclusiones.

Una vez finalizado el proyecto podemos concluir que se han alcanzado los principales objetivos previamente establecidos y una descripción personal de la realización de que comprendió el desarrollo de este proyecto.

- Se diseñó y creó una base de datos capaz de almacenar la información de los clientes, los dispensadores y las mascotas además de guardar la información en históricos para que en caso de pérdida de información se facilite la recuperación de dichos datos.
- Se desarrolló un servidor en la plataforma Raspberry Pi completamente funcional, como Broker MQTT se hizo uso de Mosquitto, servicios de LAMP como servidor web dinámico y se creó un host e instaló un cliente NO-IP para acceder de manera remota a los recursos del proyecto al ser servicios Open source el servidor fue de bajo coste. Además, que permite que los proyectos sean escalables al permitir conexión de módulos adicionales ya sea por sus pines o por comunicación Wi-Fi como el módulo NodeMCU, etc.
- Con los servicios que ofrece LAMP como servidor web dinámico se crearon usuarios para acceder de manera segura a la base de datos, así como los permisos para guardar la información proveniente del sensor HC-SR04 en la base de datos para indicar el nivel de alimento.
- La conexión del módulo NodeMcu ESP8266 a Wi-Fi se logró al implementar la librería de WiFiManager obteniendo beneficios como una conexión a red de manera rápida y sin necesidad de tocar una línea de código; permitiendo al usuario conectar el dispensador por el mismo.
- El desarrollo del nodo lector y el circuito electrónico permitió el suministro del alimento y al ser programados en la plataforma de Arduino que se libera bajo la licencia de código abierto no presentó coste alguno, de tal manera que los actuadores ejecutarán la tarea de suministrar y el sensor obtuvo el nivel de alimento del contenedor del dispensador en tiempo real; permitiendo al usuario la facilidad de suministrar el alimento en porciones adecuadas de acuerdo a la información mascota.
- El desarrollo la aplicación móvil logró controlar de forma remota el dispensador de alimento y de manera automática al indicar los horarios a dispensar. Al hacer uso de



Flutter, se logra un diseño atractivo e intuitivo para los usuarios, obteniendo un producto competitivo ante desarrollos híbridos.

Se concluye que la hipótesis no es la esperada ya que uno de los fines era lograr un producto de bajo costo y no se logró, ya que al incluir un servidor en este caso la Raspberry incrementa su valor, una forma para resolver este problema es adquirir los servicios de un alojamiento MQTT en la nube.

La conclusión del trabajo en ámbito personal por una parte presentó un reto académico desde su fase inicial hasta su fase final que fue solventado por medio de la investigación y la contribución de personas con conocimientos sobre las tecnologías utilizadas y, por otra parte, resultó ser un proyecto enriquecedor permitiendo ser una herramienta de aprendizaje e introducción para temas que en la actualidad son tendencias como el Internet de las cosas y el desarrollo de aplicaciones en Flutter comprobando la aplicabilidad del protocolo MQTT en distintos entornos y lenguajes de programación, además de que posibilitó reforzar las competencias adquiridas durante la carrera obtenidas es distintas materias de electrónica, base de datos, ingeniería de software, sistemas operativos y redes haciendo que este proyecto merezca la pena buscar otras alternativas tecnológicas para dar solución a un problema.

## Referencias

- [1] A. & B. C. Ortega, «Prospectiva de las Tecnologías de Información y Comunicación: La explotación racional de informaciones tecnológicas,» *MULTICIENCIAS*, vol. 8, n° 2, p. 227, 2008.
- [2] C. Á. Arreola, «En México, 57 de cada 100 hogares tienen alguna mascota,» *La Jornada Maya*, 30 Enero 2019.
- [3] INEGI, «Estadísticas a propósito del día mundial del internet (17 de Mayo) Datos nacionales,» 15 Mayo 2019. [En línea]. Available: [https://www.inegi.org.mx/contenidos/saladeprensa/aproposito/2019/internet2019\\_Nal.pdf](https://www.inegi.org.mx/contenidos/saladeprensa/aproposito/2019/internet2019_Nal.pdf). [Último acceso: 28 Septiembre 2019].
- [4] A. e. a. Adriansyah, «Design of Pet Feeder using Web Server as Internet of Things Application,» de *Conference: 2nd International Conference on Electrical Engineering and Informatics (Icon-EEI 2016)*, At: Pekanbaru, Riau, Indonesia, 2016.
- [5] A. Muñoz Chaparro, «Base de datos objeto-relacionales,» de *Oracle 11g PL/SQL*, Madrid, RC Libros, 2012, p. 258.
- [6] K. Ashton, «That Internet of Things Thing,» 2009. [En línea]. Available: <https://www.rfidjournal.com/articles/view?4986>.
- [7] Á. Arias, Bases de Datos con MySQL, 2014.
- [8] M. P. é. a. Zea Ordóñez, «PostgreSQL,» de *Administración de bases de datos con PostgreSQL*, Alicante, ÁREA DE INNOVACIÓN Y DESARROLLO, S.L, 2017, p. 12.
- [9] WAMP, «wampserver,» 24 Agosto 2019. [En línea]. Available: <http://www.wampserver.com/en/>. [Último acceso: 10 Septiembre 2019].

- [10 «Laragon,» 24 Agosto 2019. [En línea]. Available: <https://laragon.org/>. [Último acceso: 10 septiembre 2019].
- [11 R. E. a. F. Erick, «Setting up LAMP: Getting Linux, Apache, MySQL and PHP Working Together,» de *Chapter I. Introducing LAMP*, San Francisco , SYBEX, INC, 2004, p. 2.
- [12 «Adobe Illustrator,» 21 Septiembre 2019. [En línea]. Available: <https://www.adobe.com/mx/products/illustrator.html>. [Último acceso: 14 septiembre 2019].
- [13 «ADOBE PHOTOSHOP,» 21 Septiembre 2019. [En línea]. Available: <https://www.adobe.com/mx/products/photoshop.html>. [Último acceso: 14 septiembre 2019].
- [14 ADOBE, «Adobe XD,» 21 Septiembre 2019. [En línea]. Available: <https://www.adobe.com/mx/products/xd.html?promoid=3NQZBBTZ&mv=other>. [Último acceso: 14 Septiembre 2019].
- [15 Platzi, «Platzi,» [En línea]. Available: [https://platzi.com/multiplataforma/..](https://platzi.com/multiplataforma/)
- [16 Next-U. [En línea]. Available: [https://nextu.com/blog/apps-nativas-vs-apps-hibridas/..](https://nextu.com/blog/apps-nativas-vs-apps-hibridas/)
- [17 K. Tracy, *Mobile Application Development Experiences on Apple's iOS and Potentials*, 2012.
- [18 R. A.W, *Tabla de comparación aplicaciones*, vol. 18, D. IESA, Ed., 2013, pp. 80-81.
- [19 I. formación, «Imagina Formación,» [En línea]. Available: <https://www.imaginaformacion.com/que-es-xamarin-como-crear-un-proyecto/>.

- [20 IONIC, «IonicFramework,» [En línea]. Available:  
] <https://ionicframework.com/enterprise/why-ionic>.
- [21 Flutter, «Flutter. dev,» [En línea]. Available: <https://flutter.dev/>.  
]
- [22 J. Castillo, Desarrollo de aplicaciones Android con Android Studio: Conoce Andoird  
] Studio, p. 14.
- [23 A. Del Sole, «Introducing Visual Studio Code,» de *Visual Studio Code Distilled:  
] Evolved Code Editing for Windows, macOS, and Linux*, Cremona, IT, Apress, 2018, p.  
2.
- [24 Platzi, «Curso de Flutter Avanzado en Platzi,» [En línea]. Available:  
] <https://platzi.com/cursos/flutter-avanzado/>.
- [25 F. é. a. Luna, PROGRAMACION WEB Full Stack, Buenos Aires, ARG: RedUsers,  
] 2018.
- [26 J. é. a. Novillo-Vicuña, «Protocolos de comunicación,» de *Arduino y el internet de las  
] cosas*, Alcoy, ALC, Área de innovación y desarrollo, S.L, 2018, p. 35.
- [27 Icaza Álvarez, D. O. ét al, «Dispensador automático de alimento para mascotas,»  
] *FIGEMPA: Investigación y Desarrollo*, vol. 2, n° 2, pp. 99-106, 2017.
- [28 TG Berhan ét al, «Programmable Pet Feeder,» *International Journal of Scientific  
] Engineering and Research (IJSER)*, vol. 3, n° 11, pp. 99-104, 2015.
- [29 A. G, Giggiola del Pilar ét al, «Happy pet Dispesador electrónico para mascotas de agua  
] y comida,» Universidad Peruana de Ciencias Aplicadas UPC, Lima, 2018.
- [30 K. Aasavari y G. (Mohite), Vaishali, «Automatic Pet Feeder,» SSRN, Mubai, IN, 2018.  
]

- [31 M. R, Alberto ét al, «Dispensador automático de alimento seco para mascotas caninas,»  
] *Proyectos de Ingeniería Multidisciplinaria Una muestra del talento mexicano*, vol. I, pp. 69-74, 2017.
- [32 L. –A, Nelson Andrés ét al, «Diseño de dispensador automático de alimento para  
] mascotas, usando una celda de carga,» *Revista Científica UNINCCA*, vol. VIII, nº 1, pp. 29-41, 2017.
- [33 Vania, & Karyono, Kanisius & Nugroho, Hargyo, «Smart dog Feeder design using  
] Wireless communication, MQTT and Android Client,» de *de 2016 International Conference on Computer, Control, Informatics and its Applications*, Tangerang, IDN, 2016.
- [34 A. A., Sandra Nathaly , «Desarrollo de un prototipo para dispensar alimento y agua  
] para mascotas con tecnología IOT monitoreado y controlado a través de un dispositivo móvil,» Escuela Politécnica Nacional (EPN), Quito, EC, 2019.
- [35 C. Own, H. Shin and C. Teng,, «The study and Application of the IoT in Pet System,»  
] *Advances in Internet of Things*, vol. III, nº 1, pp. 1-8, 2013.
- [36 de la Cal. C., Javier, «Control domótico basado en el protocolo MQTT,» Universidad  
] de Valladolid, Valladolid, 2019.
- [37 C. C. Mario y G. L. Joan , «Monitorización de sensores con Arduino utilizando el  
] protocolo MQTT,» Universitat Politècnica de Catalunya, Catalunya, 2019.
- [38 P. T, C. Laura , «Diseño e implementación de un sistema dispensador de alimento para  
] mascota, controlado vía ethernet,» Escuela Superior de Ingeniería mecánica y eléctrica unidad Culhuacan, D.F, MEX, 2013.
- [39 B. –Jr, Angelo A. ét al, «Arduino-based Food and Water Dispenser for Pets with GSM  
] Technology Control,» *IJSET*, vol. 4, nº 231-234, p. IV, 2015.

- [40] L. Q, John David y R. A, Daniel , «Dispensador automático de comida para mascotas, programable y controlado remotamente,» Facultad de Ingeniería UNIVALLE, Santiago de Cali, COL, 2013.
- [41] Asaner, Ugur & Elibol, Armagan, «Low-Cost IoT Design and Implementation of a Remote Food and Water Control System for Pet Owners,» *HJSE*, vol. V, n° 4, pp. 317-320, 2018.
- [42] Cañamas, Jeremie, «Automatización y asistencia remota para cuidado de mascotas,» Escola Tècnica Superior d'Enginyeria Informàtica (UPV), Valencia, ESP, 2017/2018.
- [43] Unidad de microrregiones, «INEGI. Censo de Población y Vivienda 2010.,» 2013. [En línea]. Available: <http://www.microrregiones.gob.mx/zap/poblacion.aspx?entra=nacion&ent=30&mun=109>. [Último acceso: 03 Junio 2019].
- [44] «Purina Cat Chow,» Purina Latam, 24 Julio 2019. [En línea]. Available: <https://www.purina-latam.com/mx/catchow>. [Último acceso: 29 Julio 2019].
- [45] «Purina Latam,» 24 Julio 2019. [En línea]. Available: <https://www.purina-latam.com/mx/dogchow>. [Último acceso: 29 Julio 2019].
- [46] A. Romo, «Gamas de colores: Usos, combinaciones y aplicación web.,» 2018. [En línea]. Available: <https://neoattack.com/gamas-de-colores-para-paginas-web/>.
- [47] Flutter, «Flutter.dev,» [En línea]. Available: <https://flutter-es.io/docs/get-started/install/windows>. [Último acceso: 31 Julio 2019].

[CÓDIGO FUENTE, MÉTODOS, PROCESOS, PASOS Y DESARROLLO ES DE USO CONFIDENCIAL Y DE USO EXCLUSIVO PARA LA EMPRESA ALBA DTI]

## Anexos

### Cuestionario Pet-Home ALBA

El presente instrumento forma parte del trabajo de investigación titulada: "Prototipo dispensador de alimento para mascotas Pet-Home ALBA".

La información que nos proporcione será utilizada para analizar y determinar la factibilidad del prototipo. Solo le tomará unos minutos. Agradezco anticipadamente su valiosa colaboración.

1. ¿Cuenta con internet en casa?

☐ Si ☐ No

2. ¿Cuenta con un Smartphone?

☐ Si ☐ No

3. ¿Tiene mascotas en su hogar?

☐ Si ☐ No

4. ¿Qué tipo de mascota es?

Si no tiene una mascota, ¿Qué tipo de mascota le gustaría tener?

☐ Gato ☐ Perro ☐ Otro

5. ¿Suele estar mucho tiempo fuera de casa?

☐ Si ☐ No

6. ¿Motivo por el cual está fuera de casa?

☐ Escuela ☐ Trabajo ☐ Otros motivos ☐ Casi no salgo

7. ¿Ha obtenido un dispensador de alimento?

☐ Si ☐ No

8. ¿Estaría dispuesto a comprar un dispensador inteligente que alimente a su mascota cuando no se encuentre en casa?

☐ Si ☐ No

9. ¿Cuánto dinero está dispuesto a pagar por un dispensador inteligente?

☐ \$400 a \$700 ☐ \$800 ☐ \$1000 o más ☐ No interesado