



UNIVERSIDAD DE TARAPACÁ  
*Universidad del Estado*



# Sistema de monitoreo y control de alimentador y bebedero

## “Smart Feeder”

INTEGRANTES: RENÉ AYCA  
YAZUSKA CASTILLO  
CLAUDIO CARVAJAL  
ISRAEL TEBES

DOCENTE: DIEGO ARACENA

ASIGNATURA: PROYECTO 2

# INTRODUCCIÓN

La creciente cantidad de personas con mascotas genera el problema de asegurar su alimentación durante las ausencias del dueño. Para solucionarlo, se propone un Dispensador Automático IoT que, mediante una Raspberry Pi 4 Model B y sensores, dispensa comida o agua con precisión. Y Una aplicación móvil que permite programar horarios y supervisar el sistema a distancia. Así, se automatiza la alimentación y se mejora el bienestar de la mascota mediante un prototipo sustentable hecho con materiales reciclados.

# OBJETIVO

## GENERAL

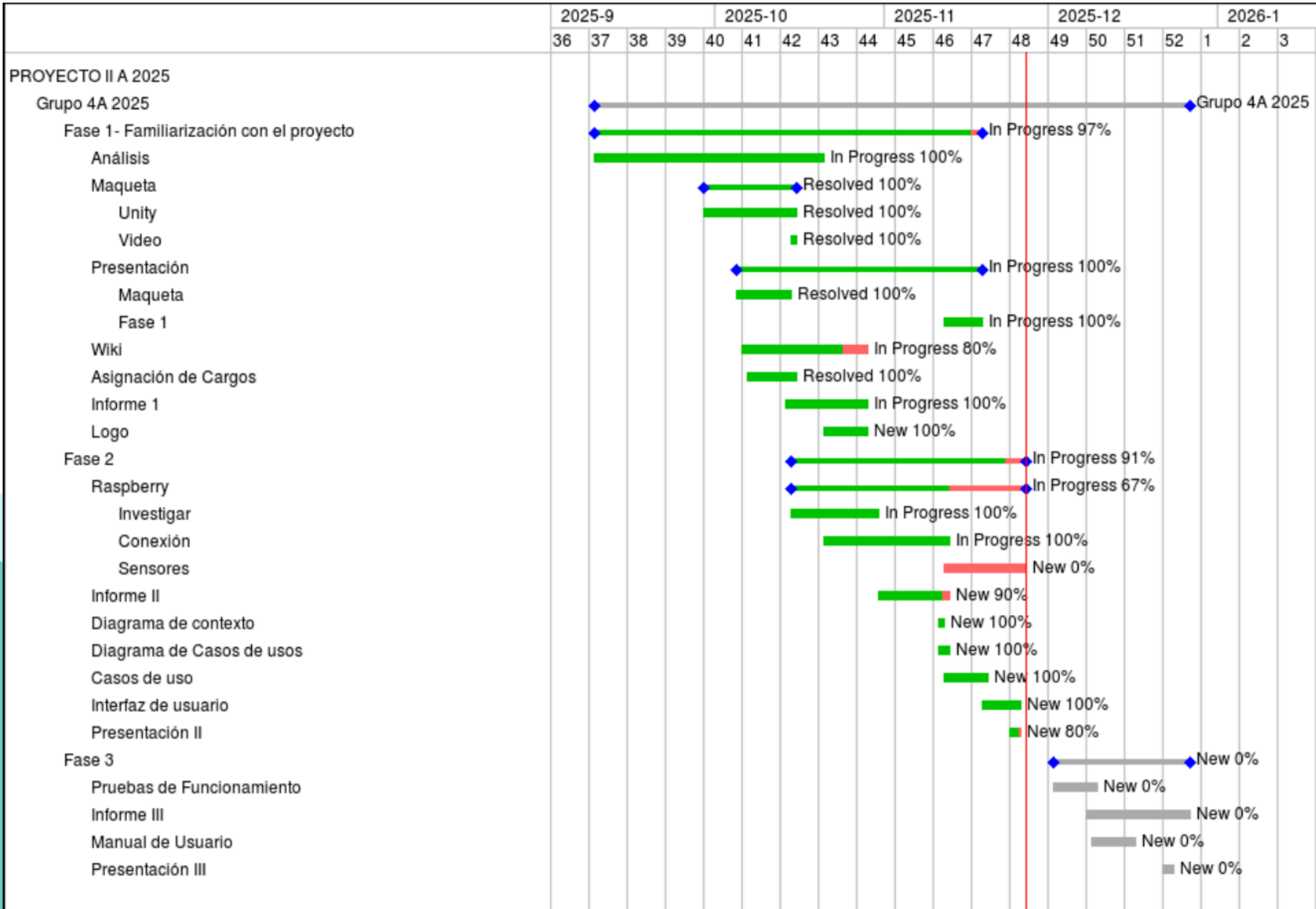
Desarrollar un dispensador automático IoT que se encargue de la alimentación e hidratación precisa para las mascotas. Permitiendo el control y monitoreo remoto por parte del dueño, con el fin de garantizar el bienestar del animal durante largas ausencias.

# OBJETIVO

## ESPECÍFICO

- Adquirir los conocimientos necesarios para el desarrollo del sistema.
- Diseñar maqueta y diagrama del dispensador, definiendo ubicación de los componentes y seleccionando e incorporando componentes hardware clave para el funcionamiento del dispensador.
- Planificar el desarrollo del proyecto para un avance eficiente.
- Desarrollar una aplicación móvil que permita la conexión con el sistema, la programación de horarios de alimentación y el monitoreo del estado de la comida y el agua.
- Documentar el desarrollo, resultados y conclusiones del proyecto realizado.
- Realizar pruebas para asegurar que el monitoreo y control automatizado funcione correctamente.

# CARTA GANTT

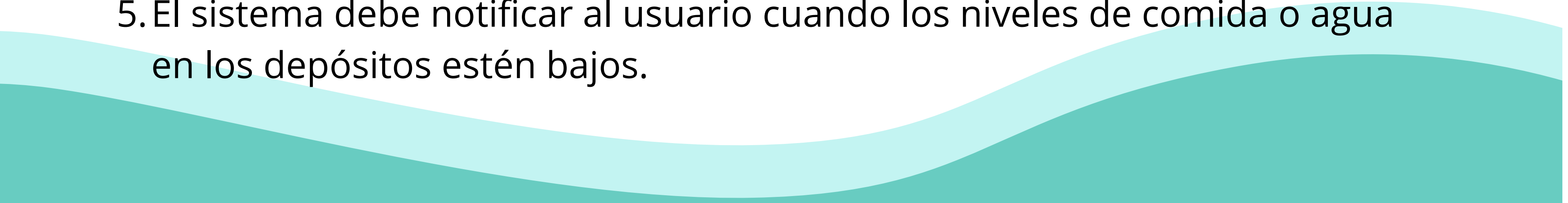


# Planificación de estimaciones

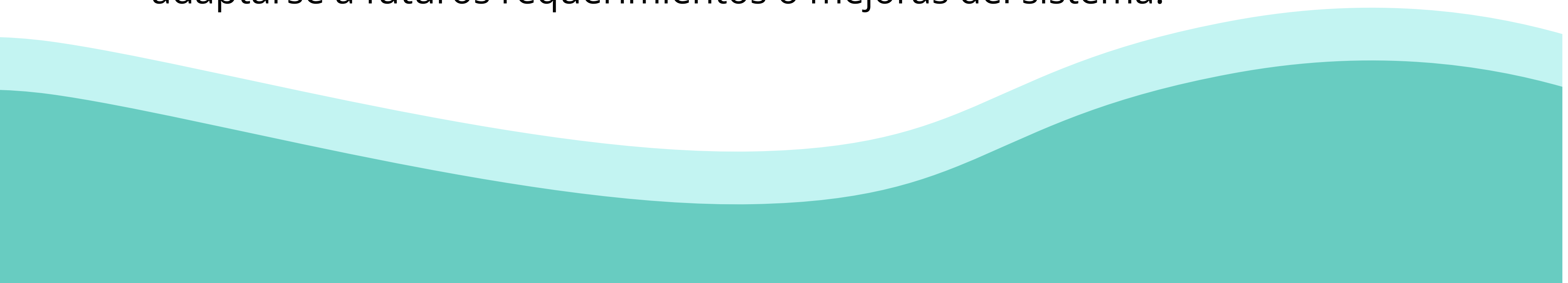
Producto	Cantidad	Costo por unidad	Costo total
Vivobook 16X	1 (30%)	\$619.990	\$185.997
HP Victus Gaming15	1 (30%)	\$859.990	\$257.997
HP Victus Gaming16	1 (30%)	\$1.299.990	\$389.997
HP 348 G7	1 (30%)	\$799.990	\$239.997
Raspberry Pi	1	\$116.990	\$116.900
Sensor de Peso	2	\$5.500	\$11.000
Sensor Ultrasónico	2	\$2.590	\$5.180
Cámara Raspberry	1	\$4.753	\$4.753
Válvula solenoide	1	\$5.229	\$5.229
Tornillo sin fin	1	\$5.500	\$5.500
Total			\$1.222.550

Elemento	Costo
Hardware	\$1.222.550
Software	\$16.960
Recursos Humanos	\$8.552.192
Costo total del proyecto	\$9.791.702

# REQUERIMIENTOS FUNCIONALES

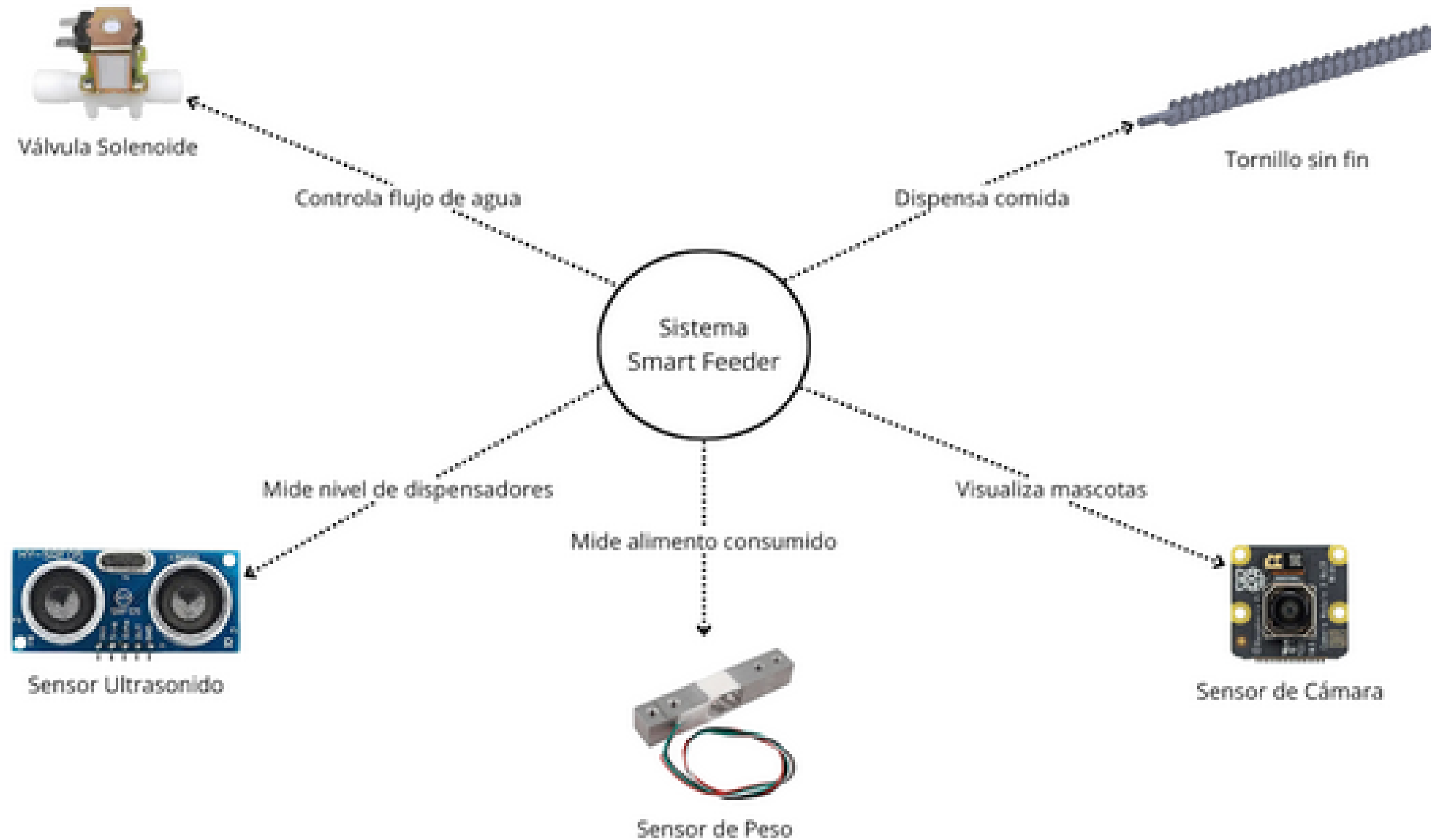
1. El sistema debe permitir al usuario programar horarios de alimentación a través de una aplicación móvil.
  2. El sistema debe dispensar una cantidad de alimento predefinida por el usuario cuando se cumpla el horario programado o se active manualmente.
  3. El sistema debe medir y registrar la cantidad de alimento dispensado y consumido (en gramos) utilizando el sensor de peso.
  4. La aplicación móvil debe mostrar en tiempo real el estado de los depósitos (nivel de comida y agua) y el historial de consumo.
  5. El sistema debe notificar al usuario cuando los niveles de comida o agua en los depósitos estén bajos.
- 

# REQUERIMIENTOS NO FUNCIONALES

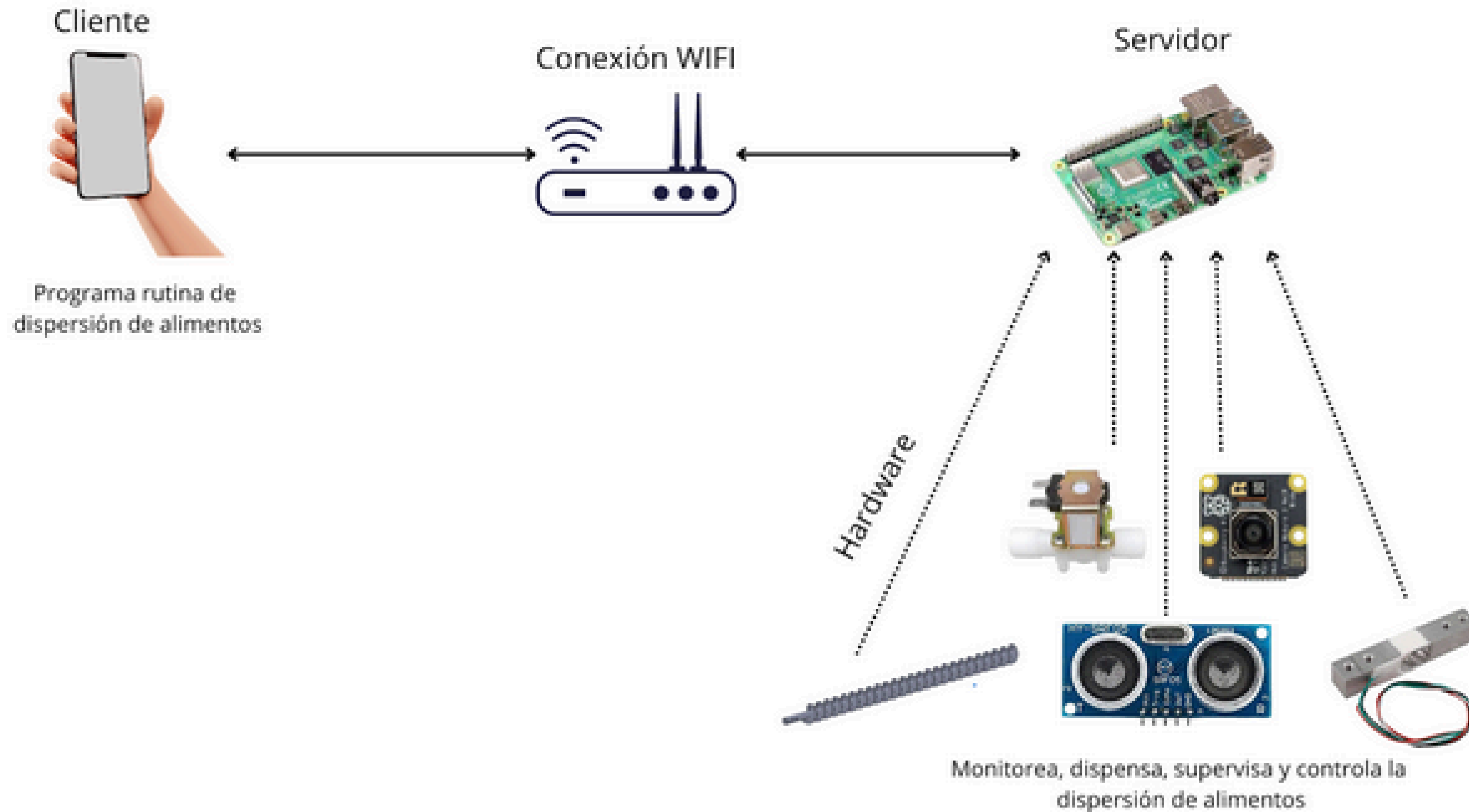
1. La aplicación móvil debe ser intuitiva, permitiendo a un usuario configurar de manera rápida un horario de alimentación.
  2. Minimizar los tiempos de respuesta en la conexión remota para un rápido envío de los parámetros del dispensador y envío de alertas.
  3. El sistema debe mantener su funcionalidad automatizada básica (ejecutar horarios) aunque se pierda la conexión a la red.
  4. Permitir la integración de sensores adicionales o accionadores para adaptarse a futuros requerimientos o mejoras del sistema.
- 



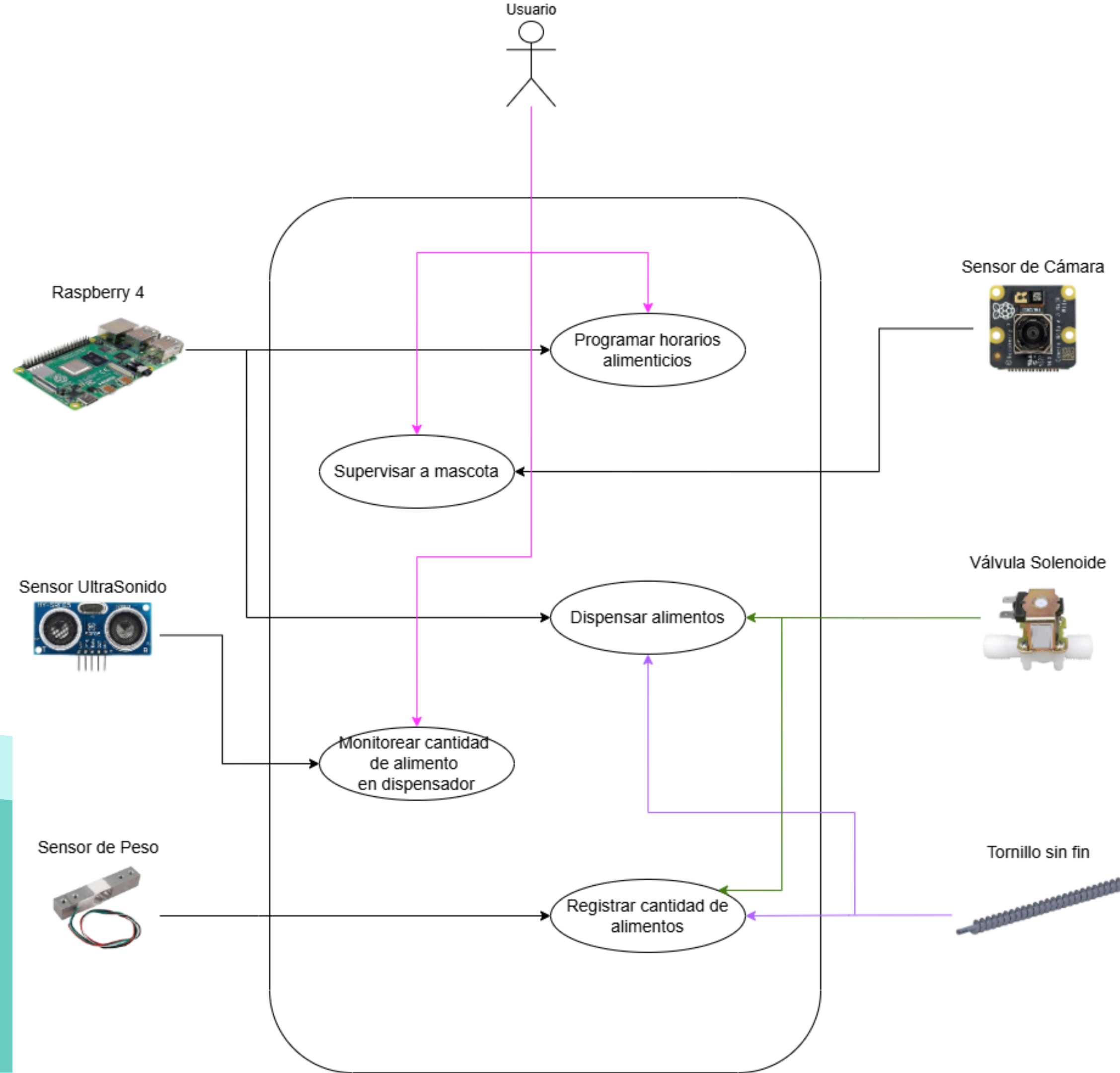
# DIAGRAMA DE CONTEXTO



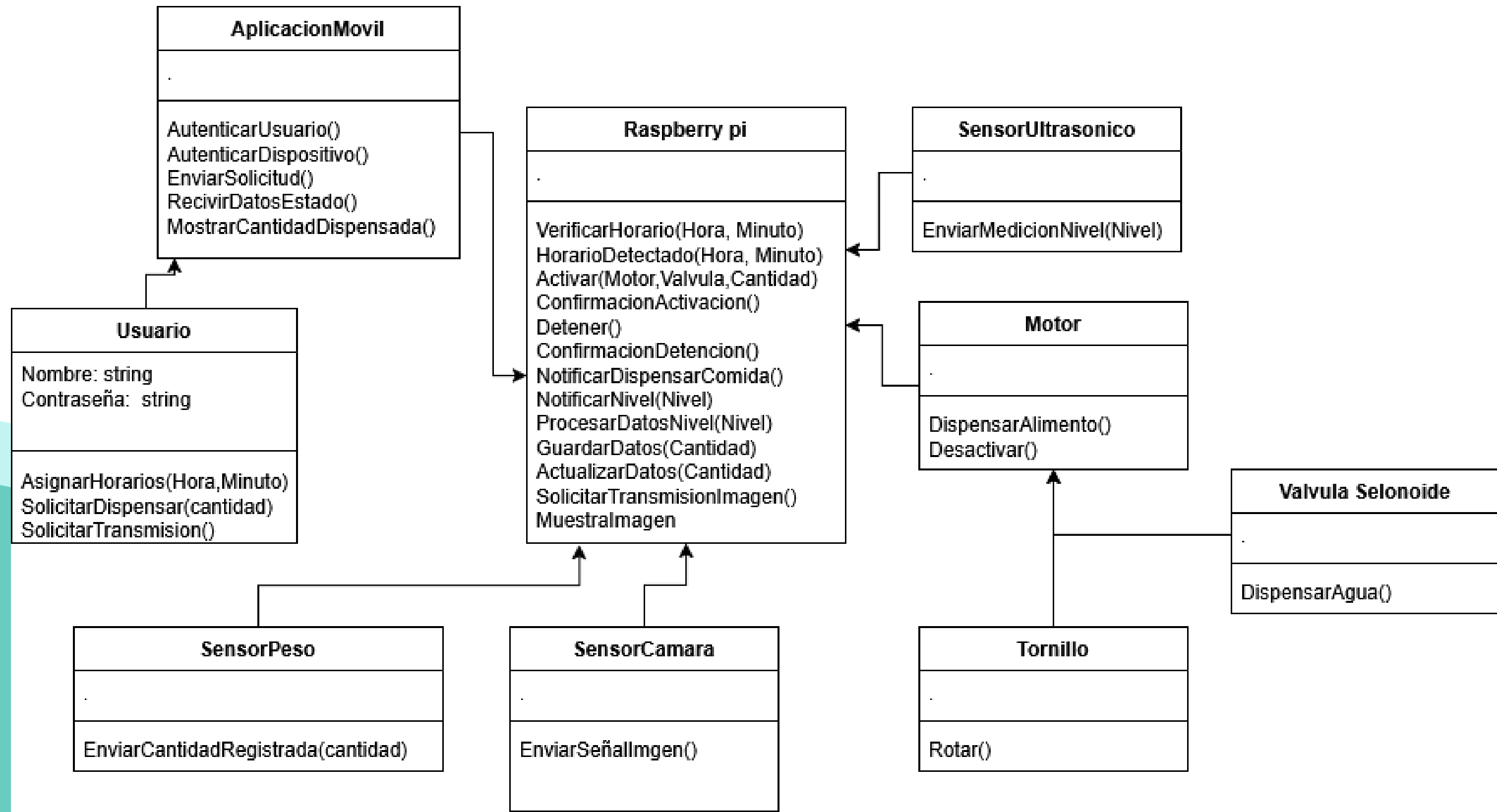
# DESCRIPCIÓN DE ARQUITECTURA



# DIAGRAMA DE CASOS DE USO GENERAL



# DIAGRAMA DE CLASE

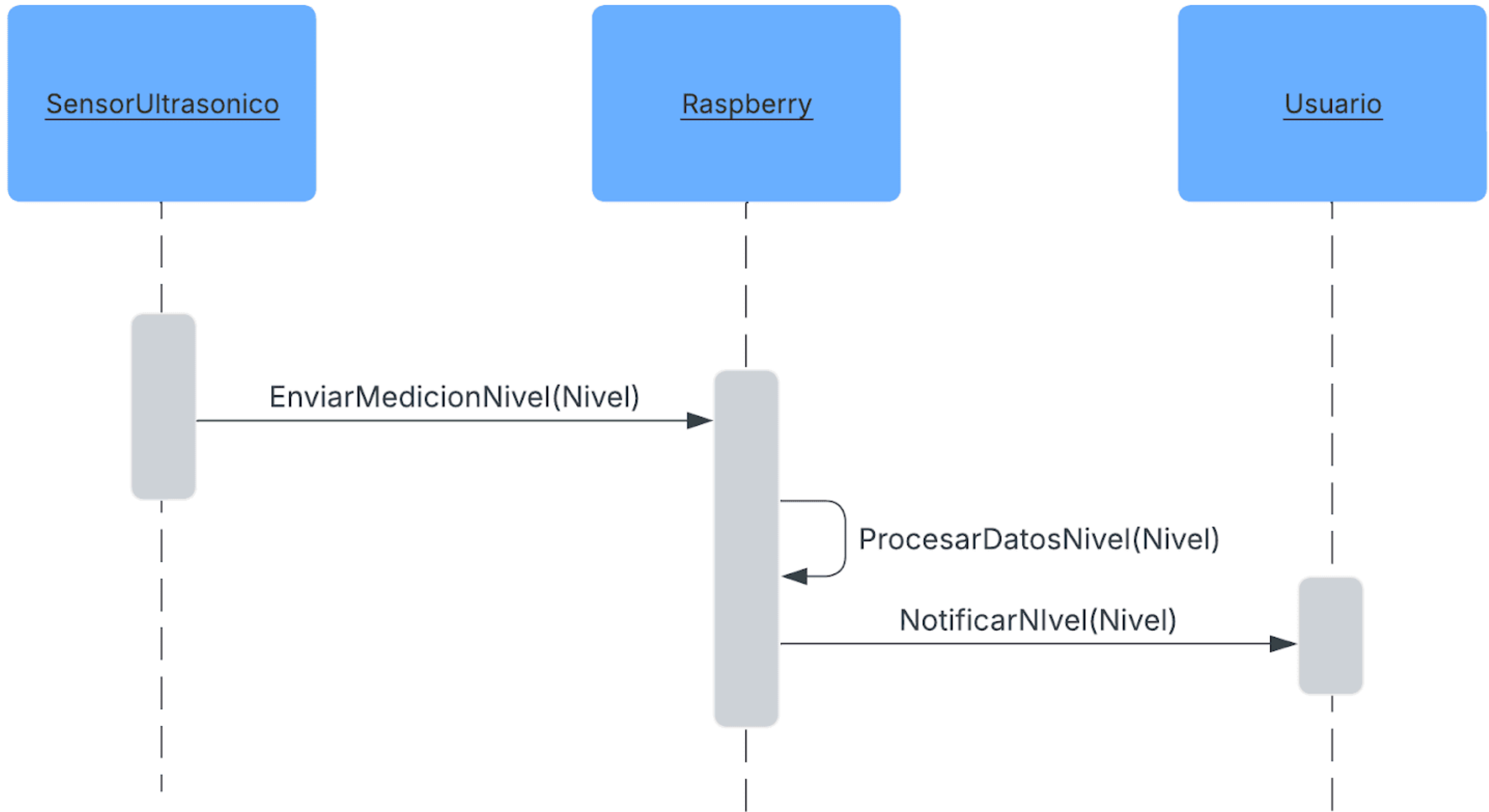


# **CASOS DE USO**



# Monitorear cantidad alimento en dispensador

Nombre Caso de Uso: Monitorear cantidad alimento en dispensador	
Autor/Fecha: Yazuska Castillo / 24-11-2025	
Descripción: El Sensor Ultrasónico envía datos a la Raspberry Pi, la cual determina la cantidad de alimento disponible en el plato y en el depósito.	
Actores: Sensor Ultrasónico	
Precondición: 1- El dispensador debe estar encendido y conectado.  2- Los sensores deben estar funcionando correctamente.  3- El sistema debe estar sincronizado con la aplicación móvil.	
Flujo Principal: Sensor  1. El Sensor Ultrasónico envía a la Raspberry Pi la medición del nivel del depósito de alimentos.	Flujo principal: Sistema  2. La Raspberry Pi recibe los datos de ambos sensores.  3. Procesa los datos para determinar la cantidad de alimento disponible.  4. Se le notifica al usuario cuál es el nivel correspondiente del dispensador.
Postcondiciones: Se notifica en la aplicación móvil.	



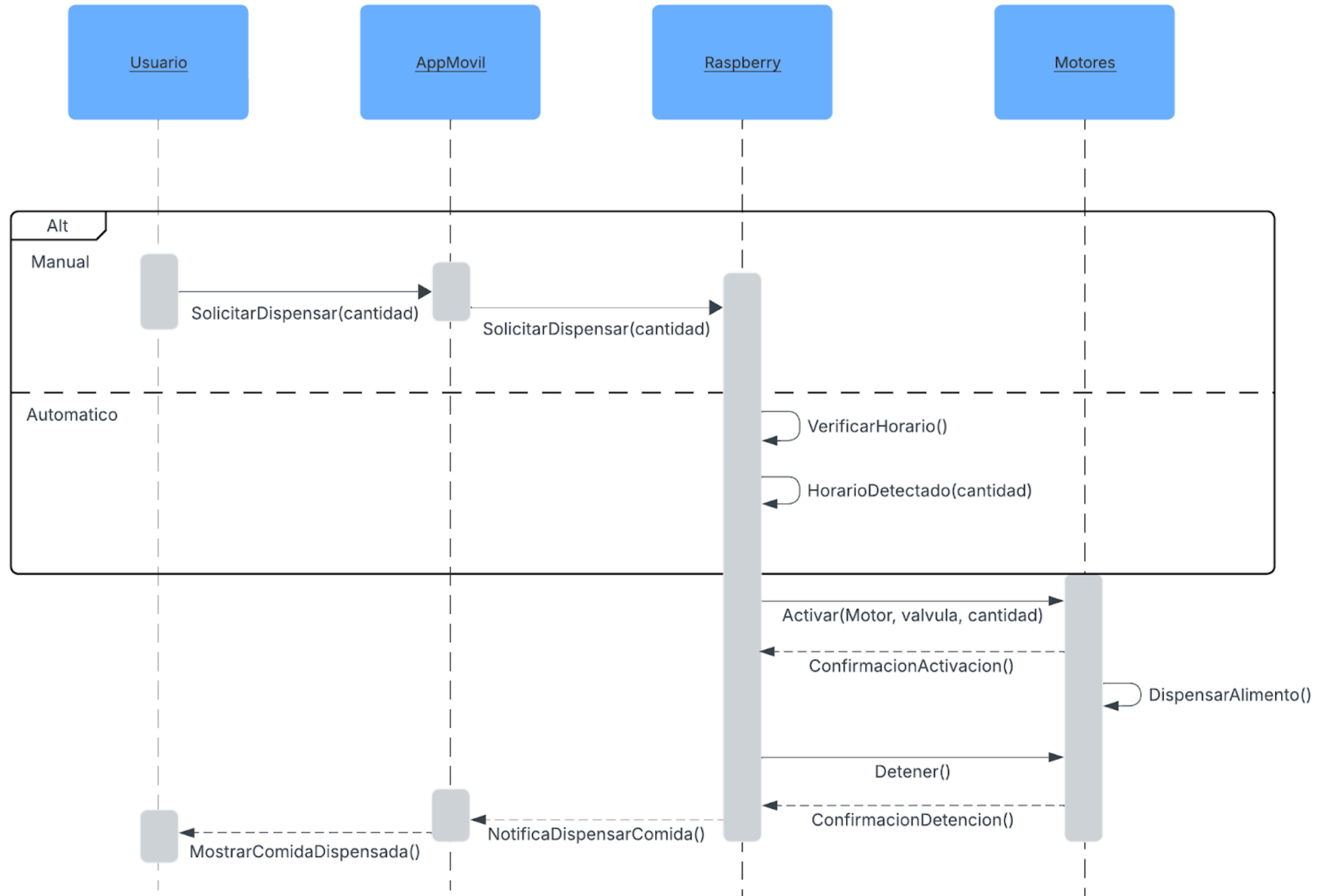
# Dispensar alimento

Nombre Caso de Uso: Dispensar alimento	
Autor/Fecha: Claudio Carvajal / 24-11-2025	
Descripción: El dispensador entrega alimento o agua a la mascota. Esto puede ocurrir de dos formas: 1. Manual, cuando el usuario lo solicita desde la aplicación móvil. 2. Automática, cuando la Raspberry Pi ejecuta un horario previamente programado en el sistema. 3. En ambos casos, la Raspberry controla el motor de tornillo sin fin y la válvula solenoide para liberar la cantidad definida.	
Actores: Válvula solenoide, Tornillo sin fin.	
Precondición: 1- El plato debe estar vacío.  2- El dispensador debe tener alimento disponible.  3- El motor debe estar operativo. 4- Para el flujo manual: la app debe estar conectada al sistema. 5- Para el flujo automático: debe existir un horario programado.	
Flujo Principal: Motores     3. El tornillo sin fin del dispensador de comida en relación a sus giros dispensa comida. La válvula solenoide con respecto a segundos dispensa agua.  5. Una vez dispensados los alimentos, los sensores son detenidos.	Flujo principal: Sistema (Manual)  1. El sistema recibe orden de dispensar alimentos.  2. Activa el motor de tornillo sin fin y válvula solenoide para liberar la cantidad indicada.  4. Detiene el motor cuando se alcanza la cantidad establecida.  6. Envía una notificación de que dispensó alimento a la mascota.
	Flujo alternativo: Sistema (Automático)



# Dispensar alimento

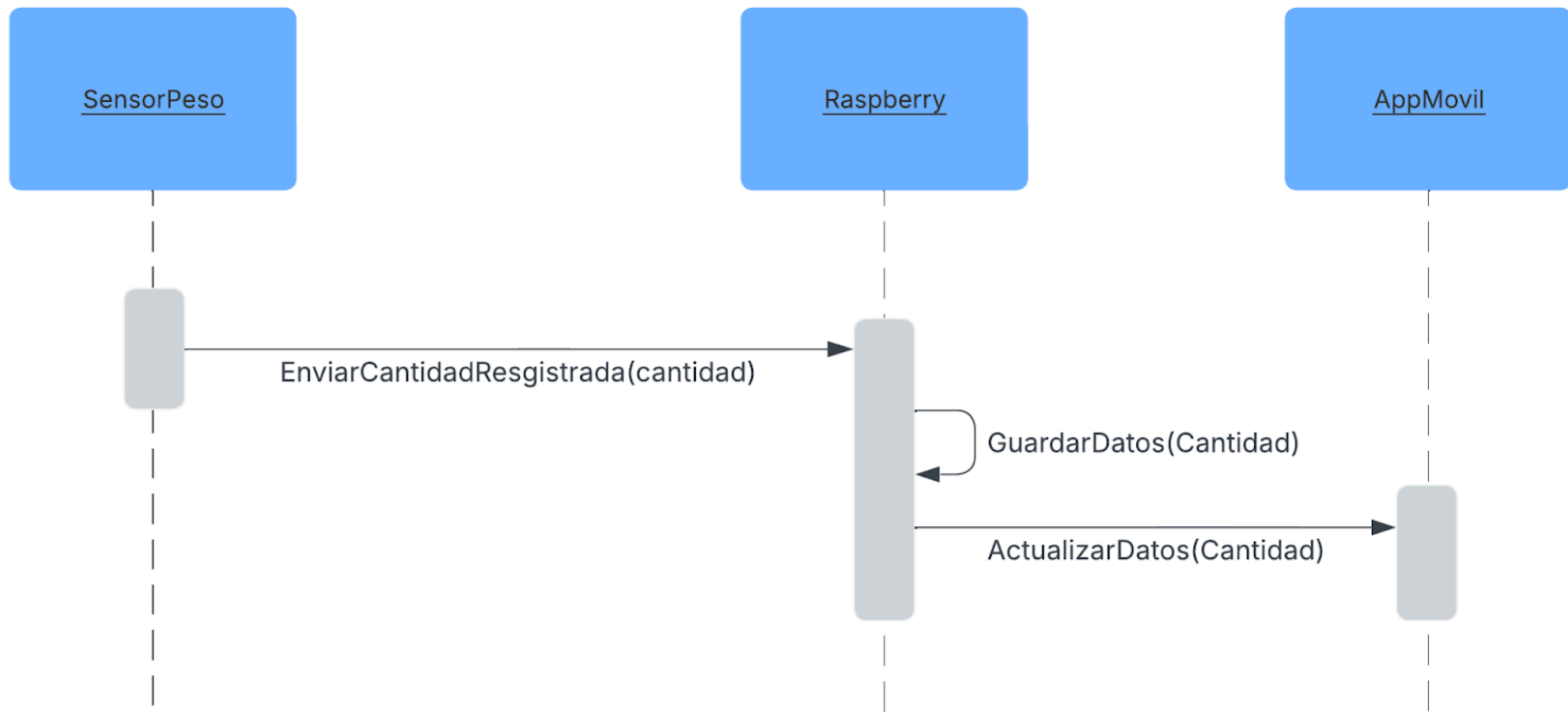
3- El motor debe estar operativo. 4- Para el flujo manual: la app debe estar conectada al sistema. 5- Para el flujo automático: debe existir un horario programado.	
Flujo Principal: Motores     3. El tornillo sin fin del dispensador de comida en relación a sus giros dispensa comida. La válvula solenoide con respecto a segundos dispensa agua.  5. Una vez dispensados los alimentos, los sensores son detenidos.	Flujo principal: Sistema (Manual)  1. El sistema recibe orden de dispensar alimentos.  2. Activa el motor de tornillo sin fin y válvula solenoide para liberar la cantidad indicada.   4. Detiene el motor cuando se alcanza la cantidad establecida.  6. Envía una notificación de que dispensó alimento a la mascota.
Flujo alternativo: Motores     3. El tornillo sin fin del dispensador de comida en relación a sus giros dispensa comida. La válvula solenoide con respecto a segundos dispensa agua.  5. Una vez dispensados los alimentos, los sensores son detenidos.	Flujo alternativo: Sistema (Automático)  1. Raspberry Pi detecta que se alcanzó un horario programado.  2. Activa el motor de tornillo sin fin y válvula solenoide para liberar la cantidad indicada.   4. Detiene el motor cuando se alcanza la cantidad establecida.  6. Envía una notificación de que dispensó alimento a la mascota.
Postcondiciones: El alimento es dispensado correctamente y gracias a otro sensor la cantidad de alimento queda registrada en el sistema.	





# Registrar cantidad de alimento

Nombre Caso de Uso: Registrar cantidad de alimento	
Autor/Fecha: René Ayca / 24-11-2025	
Descripción: El sistema registra la cantidad de alimento presente en el plato mediante los datos enviados por el sensor de peso. Esta información queda almacenada para estadísticas, control y supervisión del cliente.	
Actores: Sensor de peso, Válvula solenoide, Tornillo sin fin.	
Precondición: 1- El sensor de peso debe estar funcionando y conectado.  2- La Raspberry Pi debe estar encendida y comunicándose con el sistema.	
Flujo Principal: Sensor  1. El sensor de peso registra cantidades dispensadas en el plato de la mascota.	Flujo principal: Sistema  2. El sistema recibe las cantidades de alimentos dispensadas y las guarda en la base de datos del sistema.  3. El sistema en la aplicación móvil muestra mediante tablas la cantidad registrada de alimentos.
Postcondiciones: La cantidad de alimento queda almacenada correctamente en el sistema para su consulta futura, los datos pueden utilizarse para historial de consumo o análisis nutricional.	





# Supervisar a la mascota

Nombre Caso de Uso: Supervisar a la mascota

Autor/Fecha: Israel Tebes / 24-11-2025

Actores: Cliente.

Descripción: El cliente utiliza la aplicación móvil para ver en tiempo real lo que capta la cámara del dispensador, con el fin de supervisar la actividad de su mascota.

Precondición:

1- La cámara debe estar en funcionamiento y conectada al sistema.

2- La Raspberry Pi debe estar encendida y comunicándose con la aplicación.

Flujo Principal: Cliente

1. El cliente con la aplicación solicita al sistema la transmisión de la cámara.

Flujo principal: Sistema

2. El sistema recibe la solicitud del cliente.

3. La Raspberry Pi solicita al **sensor de cámara** la captura o transmisión de la imagen.

4. La cámara envía la señal de video o imagen al sistema.

5. El sistema procesa la información recibida.

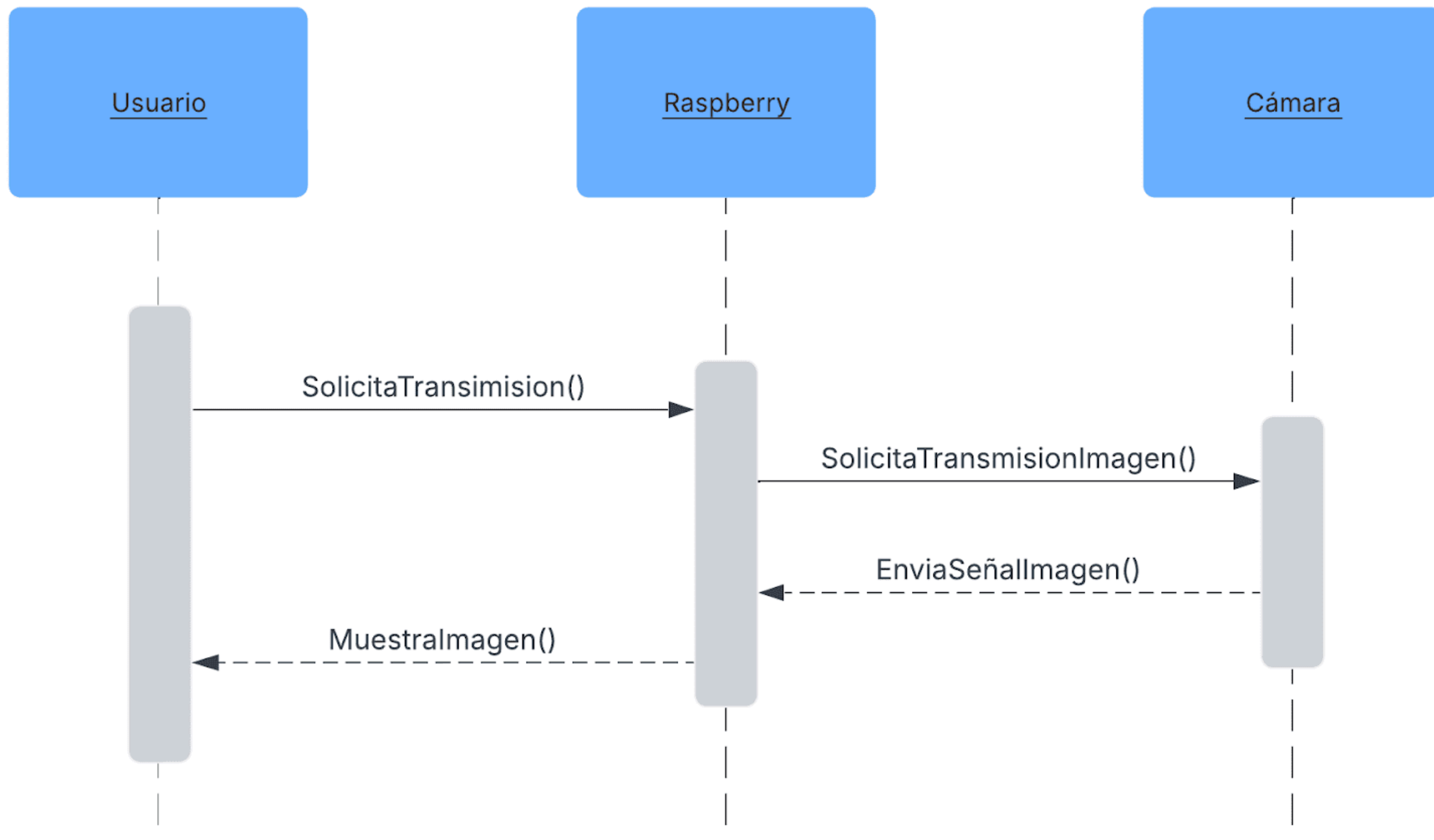
6. Envía la visualización en tiempo real a la aplicación móvil.


7. La aplicación muestra la vista al cliente.



# Supervisar a la mascota

Nombre Caso de Uso: Supervisar a la mascota	
Autor/Fecha: Israel Tebes / 24-11-2025	
Actores: Cliente.	
Descripción: El cliente utiliza la aplicación móvil para ver en tiempo real lo que capta la cámara del dispensador, con el fin de supervisar la actividad de su mascota.	
Precondición: 1- La cámara debe estar en funcionamiento y conectada al sistema.  2- La Raspberry Pi debe estar encendida y comunicándose con la aplicación.	
Flujo Principal: Cliente  1. El cliente con la aplicación solicita al sistema la transmisión de la cámara.	Flujo principal: Sistema  2. El sistema recibe la solicitud del cliente.  3. La Raspberry Pi solicita al <b>sensor de cámara</b> la captura o transmisión de la imagen.  4. La cámara envía la señal de video o imagen al sistema.  5. El sistema procesa la información recibida.  6. Envía la visualización en tiempo real a la aplicación móvil.
Postcondiciones: La imagen o video de la mascota es visualizado por el cliente.	





# Programar horarios alimenticios

Nombre Caso de Uso: Programar horarios alimenticios

Autor/Fecha: René Ayca / 24-11-2025

Descripción: El cliente configura un horario específico para que el dispensador libere alimento de manera automática.

Actores: Cliente.

Precondición:

1- El sistema debe estar en funcionamiento y conectado a la Raspberry Pi.

2- El cliente debe estar autenticado en la aplicación.

Flujo Principal: Usuario

1. El usuario ingresa la hora deseada y la cantidad de alimento a dispensar.

2. Confirma la programación y la aplicación envía los parámetros al sistema.

Flujo principal: Sistema

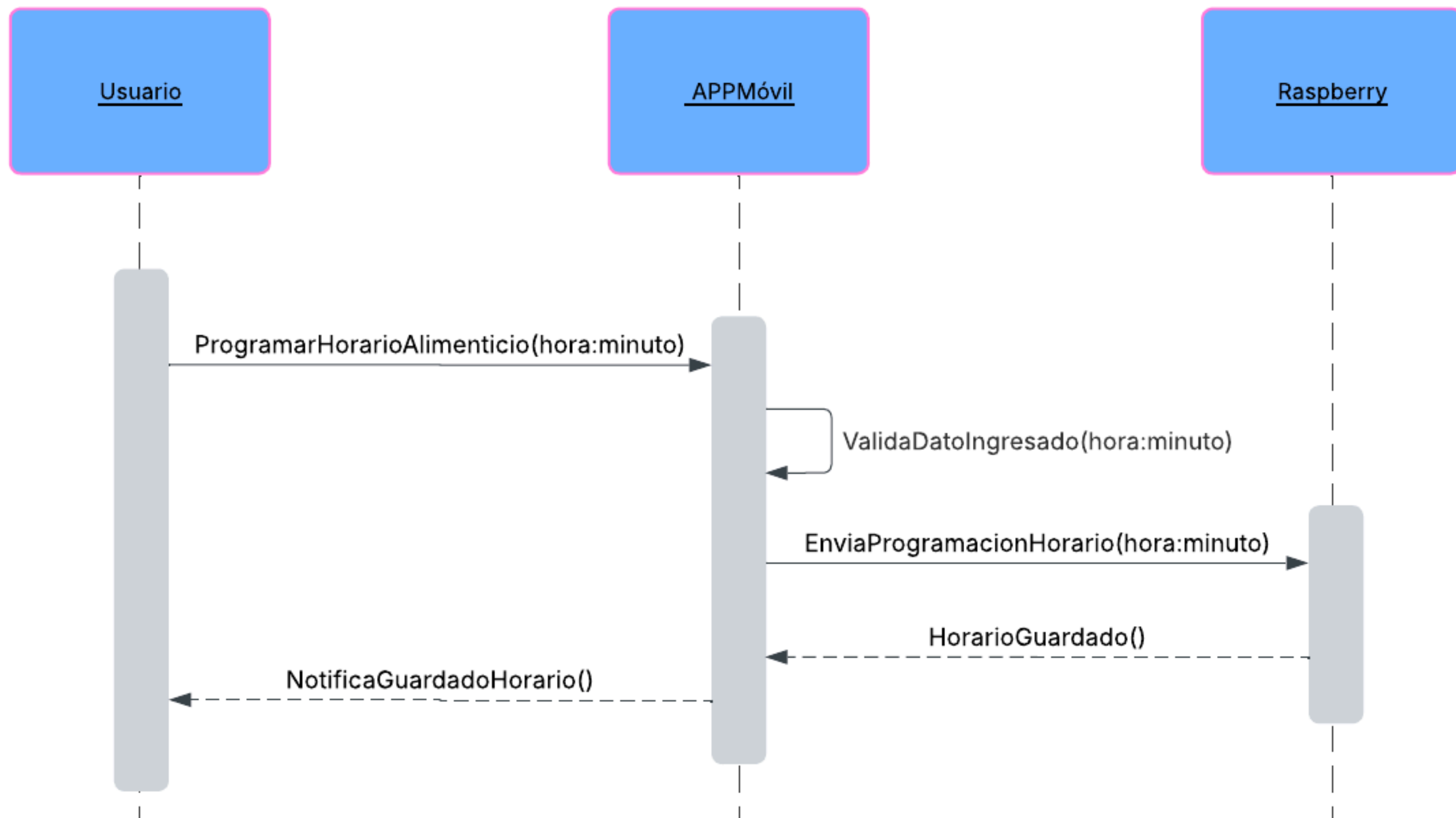
3. El sistema recibe la programación enviada por el cliente.

4. Valida formato, horario y cantidad ingresada.

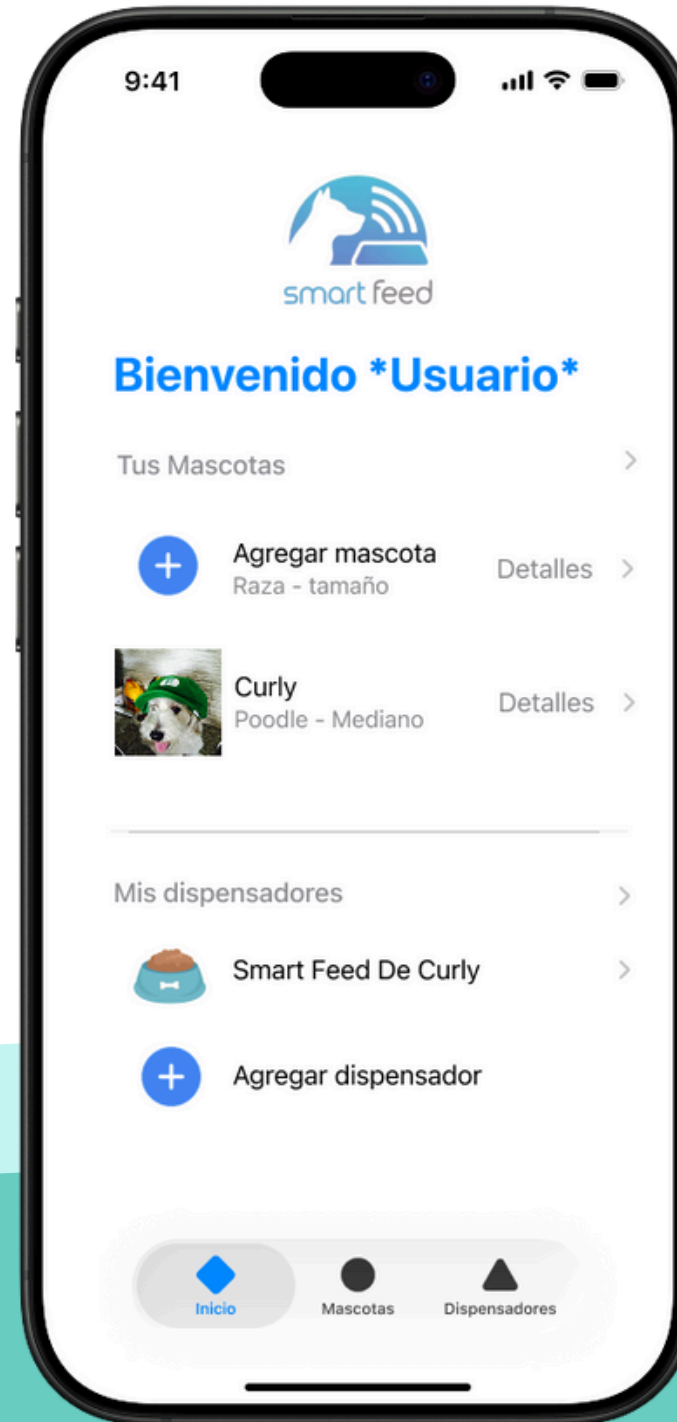
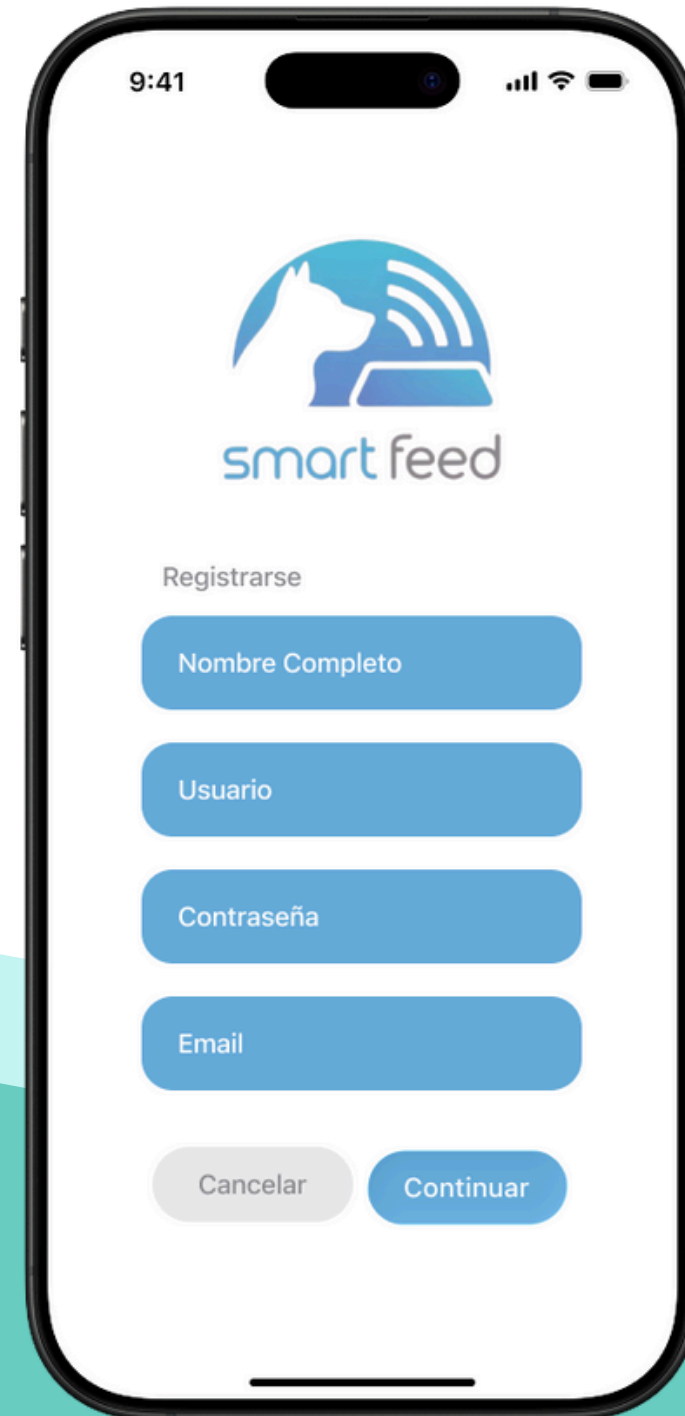
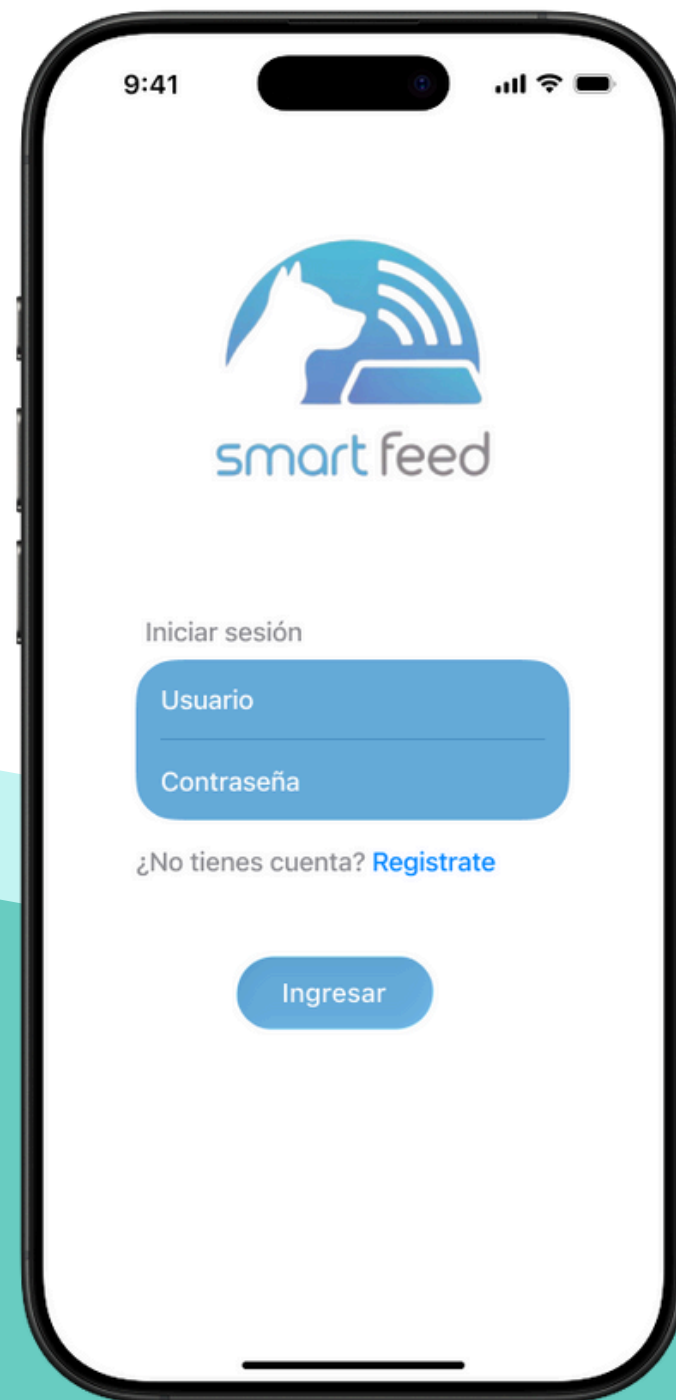
5. Registra el horario en la base de datos.

Postcondiciones: El horario alimenticio queda registrado en el sistema y en la Raspberry Pi, la dispensación automática se ejecutará en la hora programada, el cliente recibe confirmación exitosa del registro.

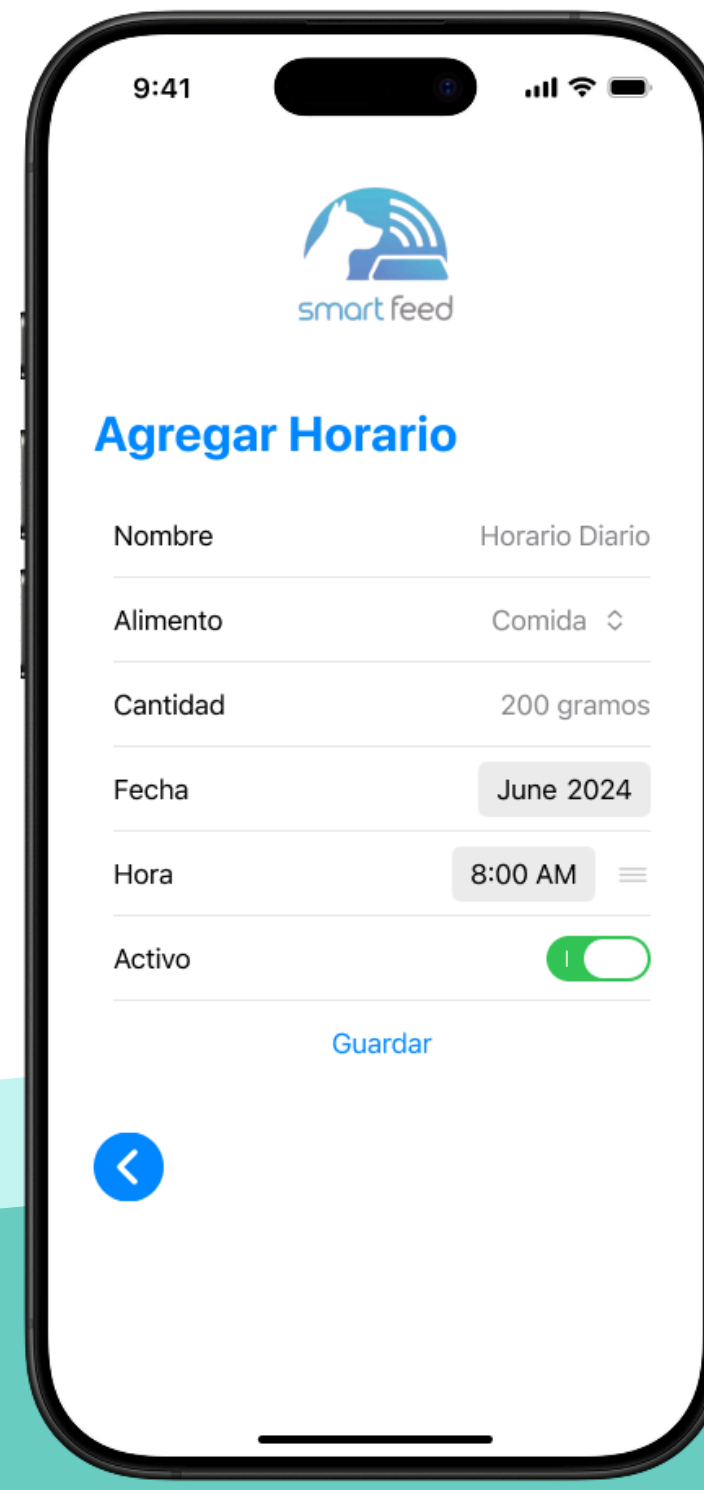
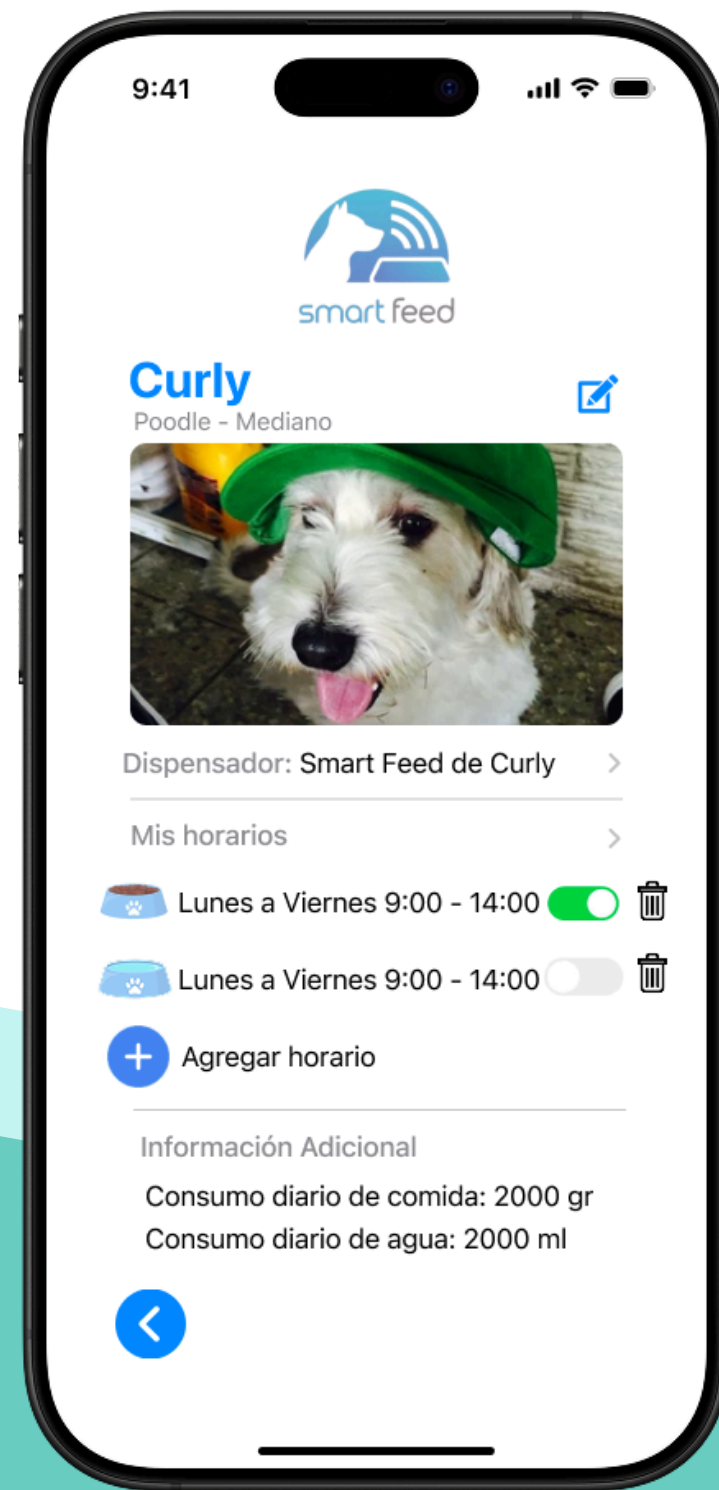
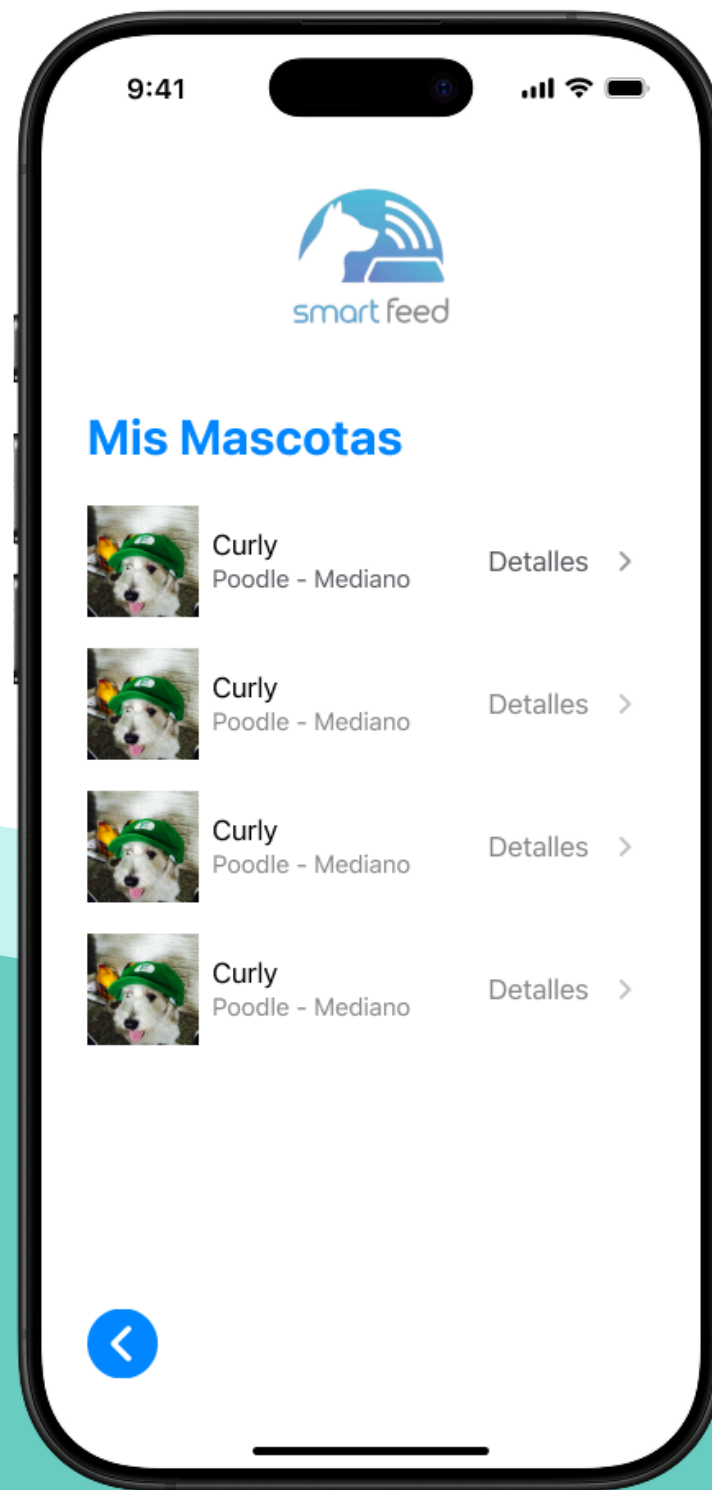




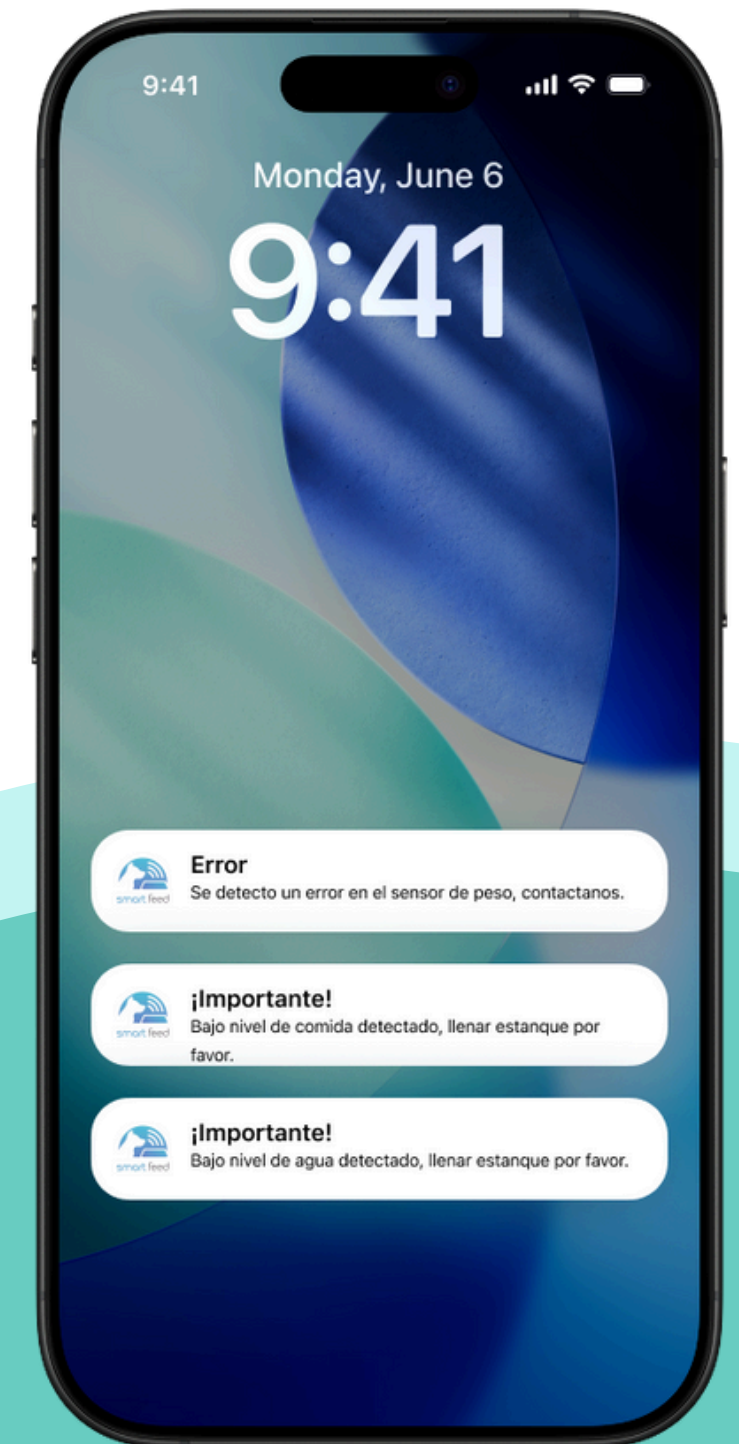
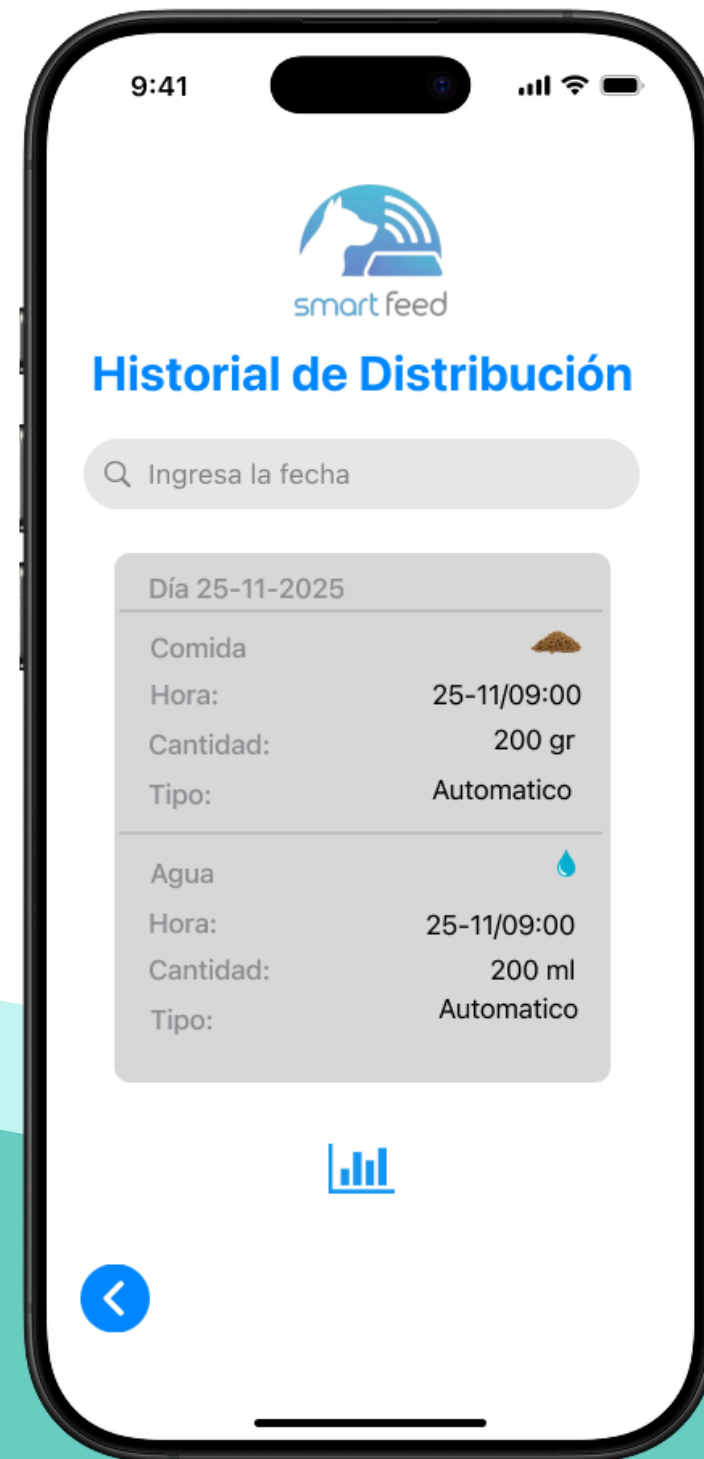
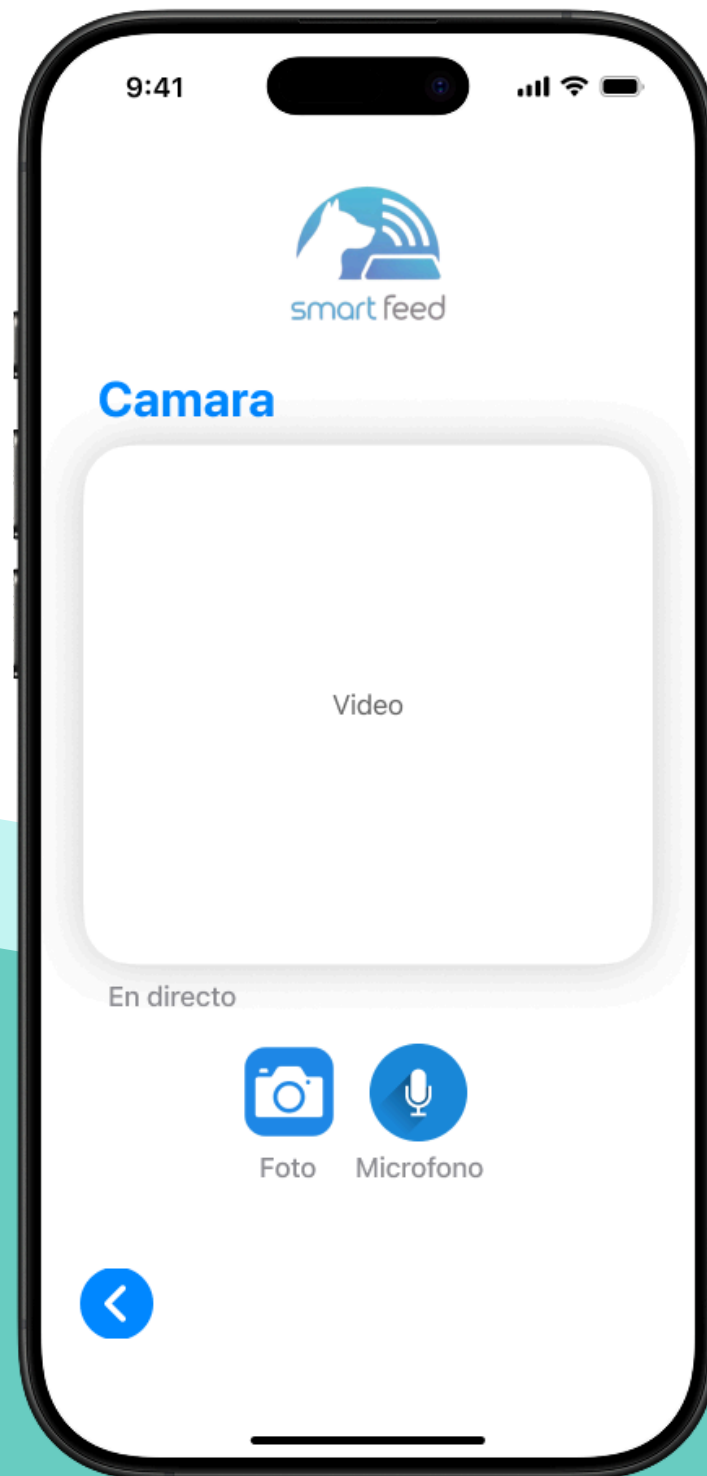
# INTERFAZ



# INTERFAZ



# INTERFAZ



# CONCLUSIÓN

El proyecto Smart Feeder demuestra que es posible automatizar de forma eficiente la alimentación e hidratación de las mascotas mediante tecnología IoT. La integración de sensores, una Raspberry Pi y una aplicación móvil permite monitorear, programar y controlar el sistema a distancia, asegurando el bienestar del animal incluso durante largas ausencias del dueño. El prototipo cumple con los objetivos planteados y sienta una base sólida para futuras mejoras gracias a su diseño escalable y sustentable.

**¡Muchas  
Gracias!**



# Referencia

[1] TALENT.COM (N.D). SALARIO DE JEFE DE PROYECTO EN CHILE. TALENT.COM.  
[HTTPS://CL.TALENT.COM/SALARY?JOB=JEFE+DE+PROYECTO](https://cl.talent.com/salary?job=jefe+de+proyecto)

[2] TALENT.COM (N.D). SALARIO DE PROGRAMADOR EN CHILE. TALENT.COM.  
[HTTPS://CL.TALENT.COM/SALARY?JOB=PROGRAMADOR](https://cl.talent.com/salary?job=programador)

[3] TALENT.COM (N.D). SALARIO DE DISEÑADOR EN CHILE. TALENT.COM.  
[HTTPS://CL.TALENT.COM/SALARY?JOB=DISEÑADOR](https://cl.talent.com/salary?job=diseñador)

[4] TALENT.COM (N.D). SALARIO DE DOCUMENTADOR EN CHILE. TALENT.COM.  
[HTTPS://CL.TALENT.COM/SALARY?JOB=DOCUMENTADOR](https://cl.talent.com/salary?job=documentador)

[5] TALENT.COM (N.D). SALARIO DE ANALISTA EN CHILE. TALENT.COM.  
[HTTPS://CL.TALENT.COM/SALARY?JOB=ANALISTA](https://cl.talent.com/salary?job=analista)

[6] RASPBERRY PI 4 MODELO B / 8GB RAM  
[HTTPS://RASPBERRYPI.CL/PRODUCTO/RASPBERRY-PI-4-MODELO-B-8GB-RAM](https://raspberrypi.cl/producto/raspberry-pi-4-modelo-b-8gb-ram)

[7] GROVEPI+ STARTER KIT - DEXTER INDUSTRIES  
[HTTPS://WWW.DEXTERINDUSTRIES.COM/STORE/GROVEPI-STARTER-KIT/](https://www.dexterindustries.com/store/grovepi-starter-kit/)

# Referencia

**[8] CONEXIÓN GPIO DE RASPBERRY PI 3 | ELECTRÓNICA Y CIENCIA**

**[HTTPS://WWW.ELECTRONICAYCIENCIA.COM/2016/11/CONEXION-GPIO-DE-RASPBERRY-PI-3.HTML](https://www.electronicayciencia.com/2016/11/conexion-gpio-de-raspberry-pi-3.html)**

**[9] RASPBERRY PI Y EL IOT: GUÍA PARA ENTENDER SU PAPEL EN EL INTERNET DE LAS COSAS**

**[HTTPS://MONRASPBERRY.COM/ES/GUIA-DE-RASPBERRY-PI-IOT/](https://monraspberry.com/es/guia-de-raspberry-pi-iot/)**

**[10] SEEED STUDIO (N.D). GROVE – ADC FOR LOAD CELL (HX711). SEEED STUDIO.**

**[HTTPS://WWW.SEEEDSTUDIO.COM/GROVE-ADC-FOR-LOAD-CELL-HX711-P-4361.HTML](https://www.seeedstudio.com/Grove-ADC-for-Load-Cell-HX711-P-4361.html)**

**[11] DEXTER INDUSTRIES (N.D). GROVEPI.**

**[HTTPS://WWW.DEXTERINDUSTRIES.COM/GROVEPI/](https://www.dexterindustries.com/GrovePI/)**

**[12] MADE-IN-CHINA (N.D). STAINLESS STEEL SCREW FLIGHT SPIRAL BLADE HELICAL BLADE SHAFTLESS FOR DRILLING MACHINE.**

**[HTTPS://ES.MADE-IN-CHINA.COM/CO\\_SEITOTECH/PRODUCT\\_STAINLESS-STEEL-SCREW-FLIGHT-SPIRAL-BLADE-HELICAL-BLADE-SHAFTLESS-FOR-DRILLING-MACHINE\\_YSSISYGIUY.HTML](https://es.made-in-china.com/co_seitotech/product_stainless-steel-screw-flight-spiral-blade-helical-blade-shaftless-for-drilling-machine_yssisygiuy.html)**

**[13] DISPENSADOR INTELIGENTE PARA MASCOTAS CON IOT – MONITOREO Y CONTROL REMOTO**

**[HTTPS://WWW.RESEARCHGATE.NET/PUBLICATION/348825241\\_MONITOREO\\_Y\\_CONTROL\\_REMOTO\\_DE\\_UN\\_DISPENSADOR\\_DE\\_ALIMENTO\\_PARA\\_MASCOTAS\\_BASADO\\_EN\\_IOT](https://www.researchgate.net/publication/348825241_monitoreo_y_control_remoto_de_un_dispensador_de_alimento_para_mascotas_basado_en_iot)**