Pet Feeder IoT

75.39 - Aplicaciones Informáticas

Facultad de Ingeniería

Universidad de Buenos Aires

Alumno: Darío Miñones
Padrón: 86644
E-mail: dminones@gmail.com

Introducción y justificación

La correcta alimentación de las mascotas en muy importante para su salud, una parte importante de esta correcta alimentación es hacerlo en horarios y cantidades controladas.

Actualmente muchas veces para los dueños de casa es difícil estar ahí siempre en tiempo y forma para darle a su mascota la comida en el horario que corresponde, es decir surge la necesidad de un dispositivo que dispense de forma automática y precisa el alimento y permita además el monitoreo remoto del estado del mismo.

Fundamentación

Actualmente existen algunos dispensadores de comida automáticos, estos se programan en el dispositivo generalmente con interfaces no muy amigables y siempre funcionando de forma independiente y aislada.

La disminución en los costos de los microprocesadores y su creciente capacidad de procesamiento hace cada vez más facil la incorporación a internet de dispositivos que hasta el momento funcionaban de manera autónoma.
De esta forma es posible

La solución propuesta al problema es una aplicación de internet de las cosas, en este caso un dispensador automático controlado por un microcontrolador que se conecta a internet para actualizar su configuración de horarios y para reportar el estado del dispositivo, esto es el nivel de alimento y las veces que se dispenso con su horario cantidad etc.

Metodología de desarrollo

Para el desarrollo de la aplicación se utilizarán metodologías ágiles, en particular se plantea scrum como punto de partida.

Scrum es una metodología de desarrollo iterativa e incremental, se plantea la iteración en ciclos llamados sprints en los cuales se desarrollan un conjunto de funcionalidades que agreguen valor al producto.
En cada iteración se decide que funcionalidades incluir mediante una depuración del backlog de tareas, luego de cada sprint se revisan las funcionalidades incluidas y el proceso y progreso logrado en esta iteración para ver que aspectos mantener y cuales mejorar en la próxima.

Por las características del proyecto hay que hacer algunas salvedades al proceso base que plantea scrum.

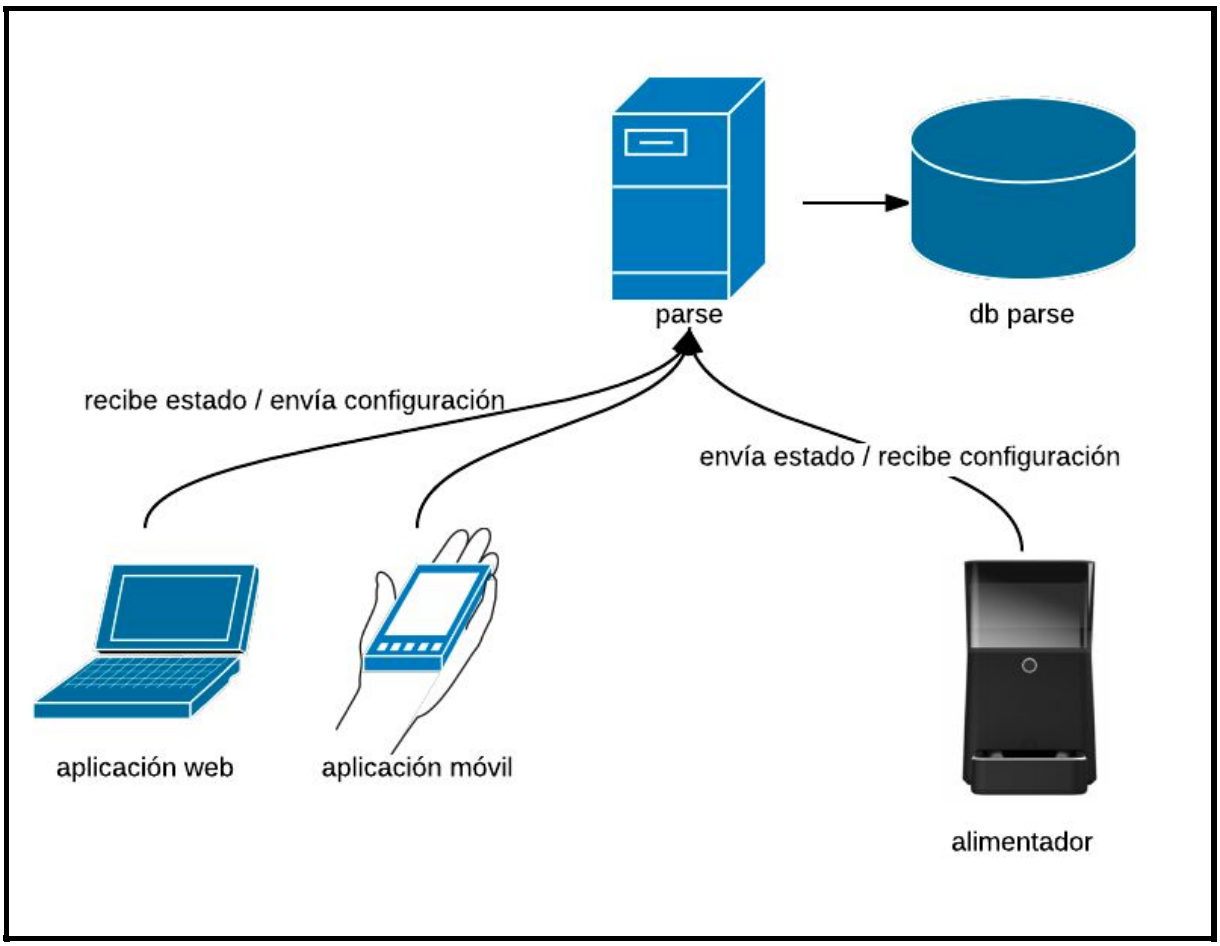
Roles
Al ser un proyecto unipersonal los roles están condensados en la misma persona (Product Owner, Scrum Master, Team) a excepción de los feedback de la cátedra que actuarán en cierta forma como input para el Product Owner.

Reuniones
Las reuniones diarias no tienen sentido pero sí se utilizarán los hitos que las demás reuniones conllevan, es decir, la depuración del backlog, la retrospectiva etc. para el correcto direccionamiento del proyecto.

Infraestructura

El proyecto tiene tres componentes, el dispensador de alimento, el servidor y las aplicaciones móviles para gestión y monitoreo.

Diagrama de Arquitectura



Dispensador de alimento
Para el desarrollo de un prototipo se utilizará la placa arduino que permite la programación de un microcontrolador de manera fácil y rápida de la misma forma que la integración con diferentes protocolos de comunicación como por ejemplo wi-fi.
Para el pase a producción se necesitará la construcción de electrónica según las especificaciones que surjan de la fase de desarrollo.

API de servicios
Se necesita contar con una base de datos en la nube con la que se comunicarán los dispositivos dispensadores para guardar los datos de estado y leer los cambios de configuración y las aplicaciones móviles para gestionar la configuración y monitorear estados.

Se utilizará Parse (parse.com), este servicio ofrece entre otras cosas la posibilidad de contar con una base de datos y su API de servicios en la nube para poder accederla desde cualquier dispositivo, cuenta incluso con SDK para Android, iOS, Javascript e incluso Arduino.
Estas características lo hacen muy conveniente, especialmente para la fase de desarrollo pudiendo focalizarse en las aplicaciones.

Aplicaciones móviles
Se desarrollarán aplicaciones nativas iOS y Android con lo cuál es necesaria el IDE de desarrollo de cada una de estas plataformas, dispositivos de pruebas con cada uno de los sistemas operativos.
Para el pase a producción se utilizarán las tiendas de aplicaciones de cada fabricante.

Aplicación web
También se desarrollará una versión web en angularjs, para esto no es necesario ningún equipamiento especial.
El pase a producción se hará con un servidor en la nube.

Requerimientos funcionales

- Un usuario debe poder configurar los horarios en los que se dispensa alimento desde la aplicación móvil
- Un usuario debe poder verificar que se dispensó correctamente la comida, en que horario, que cantidad, etc.
- El dispositivo tiene que poder funcionar aunque pierda conexión a internet en algún momento.

Requerimientos no funcionales

- El dispositivo tiene que tener un tamaño reducido que permita que se lo coloque cómodamente en un hogar.

Estado del arte

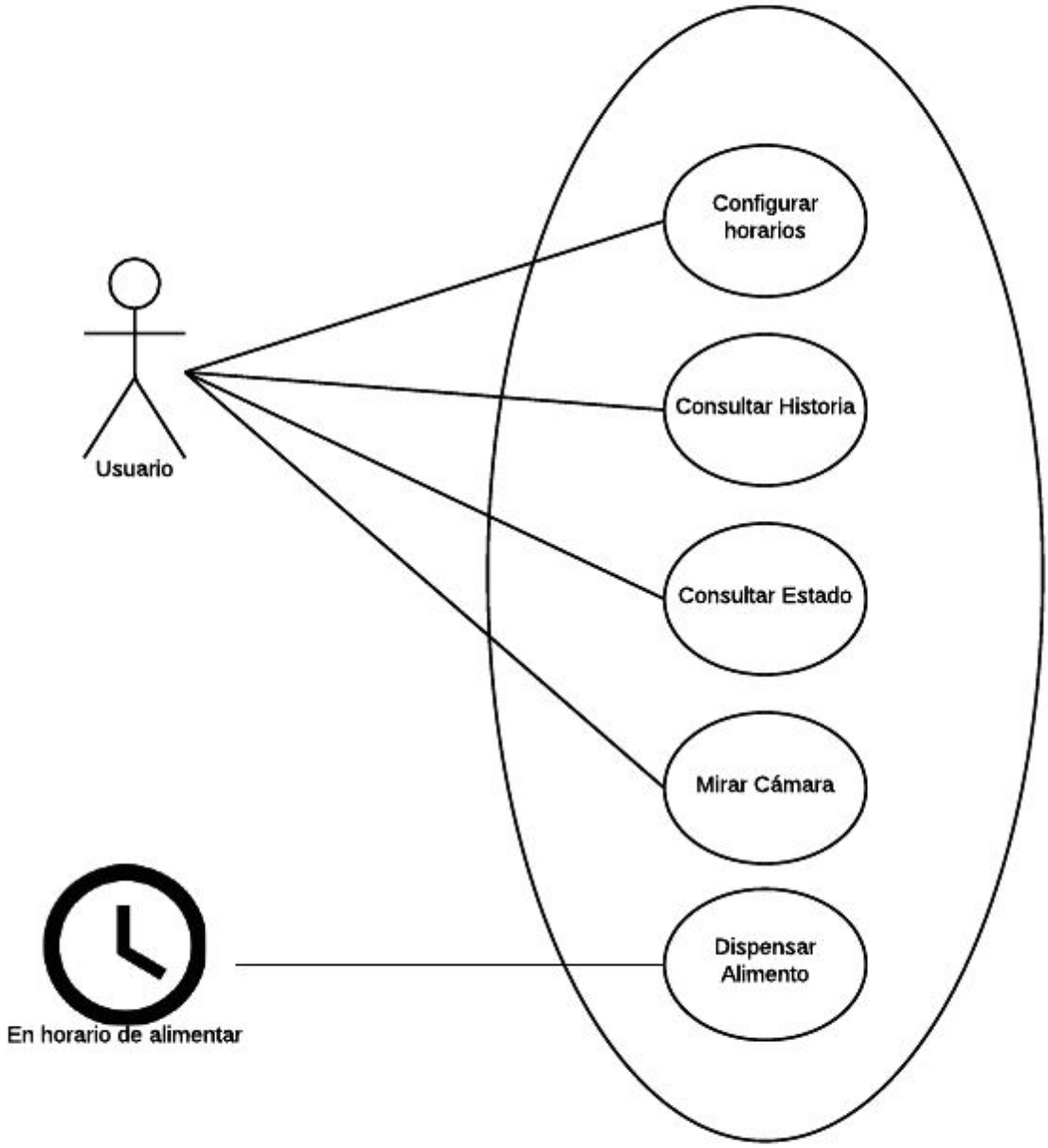
Del análisis del estado del arte de la posible competencia (ver Anexo 1: Estado del arte) se desprende que ninguno de los productos satisface completamente las necesidades del usuario.

Ninguno de los dispositivos envía información del estado en la nube con lo cual no hay forma de saber si se sirvió la comida exitosamente, la cantidad de alimento sobrante, si el dispensador está en un estado válido, etc.
Esta información es esencial para poder monitorear remotamente que la mascota esté comiendo en tiempo y forma y que lo va a poder seguir haciendo.

Por otro lado ninguno tiene aplicaciones para Android y iOS que brinden una buena experiencia a la hora de gestionar la configuración del dispositivo.

Diseño

Diagrama de Casos de Uso



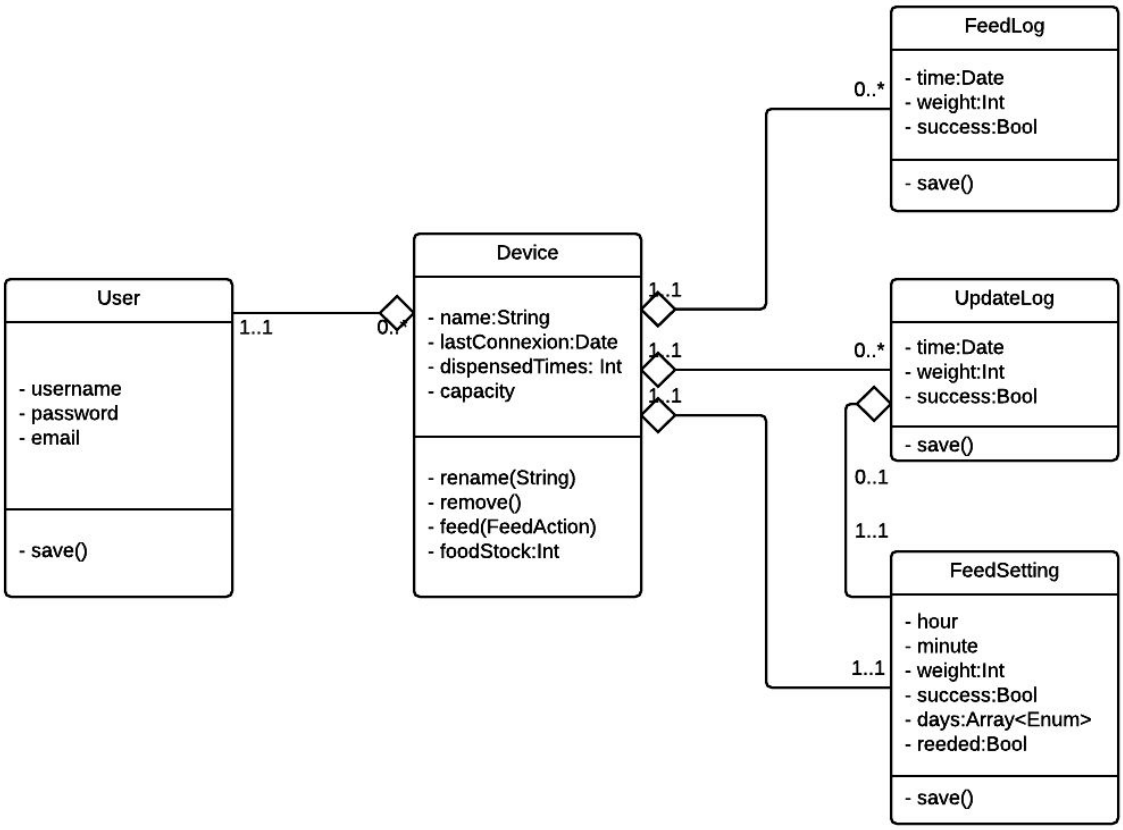
Casos de Uso

Caso de uso: Configurar Horarios	
Descripción: El usuario configura los horarios para cada día de la semana y las cantidades de alimento a servir.	
Actores participantes: Usuario	
Pre-condiciones: El usuario tiene que haber vinculado la aplicación móvil a un dispositivo.	
Flujos	
Flujo Principal	
1	El usuario elige un horario, cantidad de alimento y dias de la semana que aplica
2	El usuario guarda la configuración
Flujos Alternativos	
A1.1	**si no hay**
Flujos de Excepcion	
E1.1	Si no tiene conexión a internet para guardar los datos produce un error
Post-condiciones: La configuración queda guardada en el servidor y vinculada al dispositivo	

Caso de uso: Dispensar Alimento
Descripción: El dispositivo en el horario seleccionado por el usuario dispensa el alimento y notifica al servidor que se logró correctamente o si hubo algun error

Actores participantes: Horario de Alimentar	
Pre-condiciones: Tiene que haber una configuración valida de dispensado de alimento	
Flujos	
Flujo Principal	
1	Llega el horario de servir la comida y el dispensador sirve la comida
2	El dispositivo notifica al servidor que la operación fue exitosa
Flujos de Excepcion	
E1.1	Si no logra dispensar o hay algún error lo notifica al servidor
Post-condiciones: El registro de la operación queda guardada en el servidor para su posterior consulta	

Diagrama de Clases



User
Representa a un usuario de la aplicación, permite identificarlo contra un Device.

Device
Representa un dispositivo, es decir un dispensador.
Mantiene el estado del mismo, su configuración vigente (FeedSetting) y su historial de actualizaciones(UpdateLog, FeedLog)

FeedSetting
Representa una configuración de tiempo y cantidad de comida a dispensar para un dispositivo.

UpdateLog
Representa una actualización de los datos del dispositivo, por ejemplo un cambio en la cantidad de comida o estado de los sensores

FeedLog
Representa un evento de dispensado, es decir hora, cantidad dispensada y estado de la transacción

Todas las clases deberán ser serializables para poder enviarse por la red.

Diagrama de Secuencia

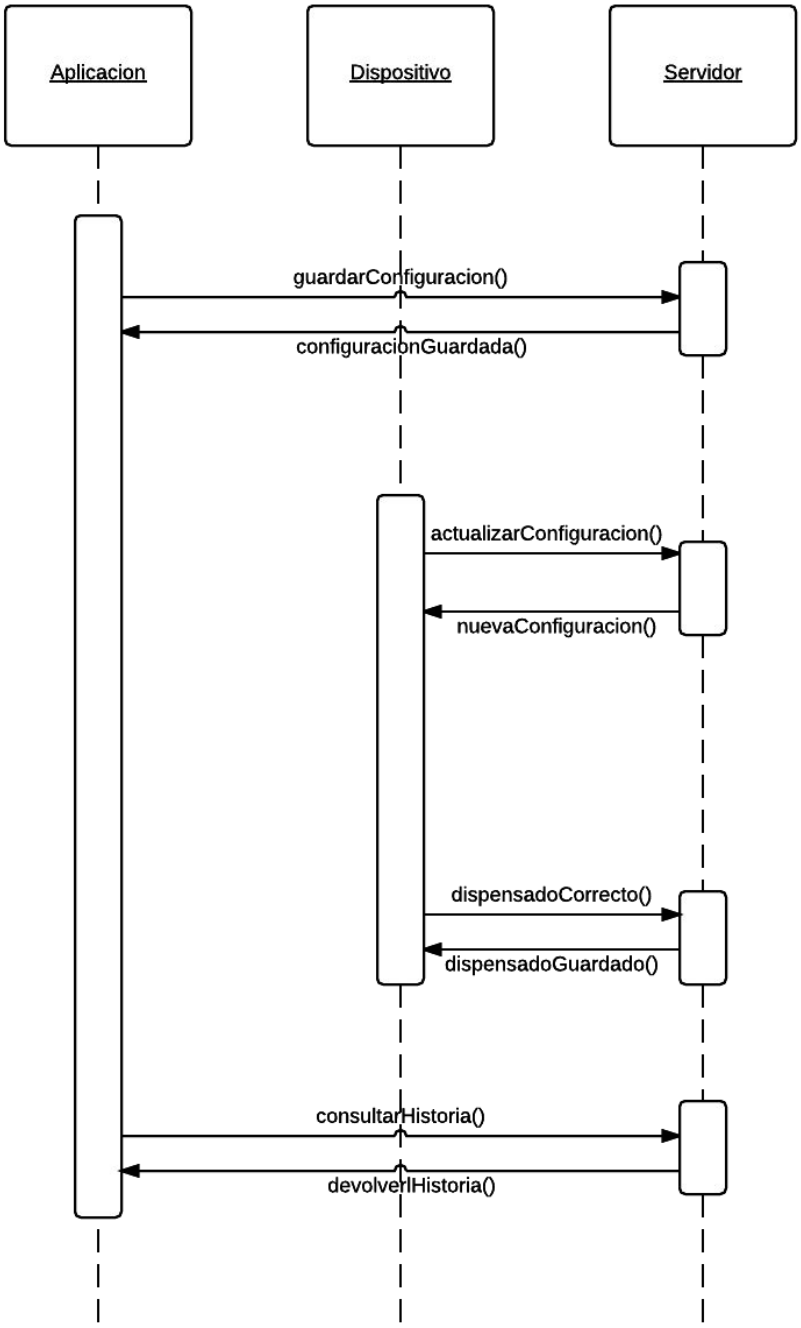
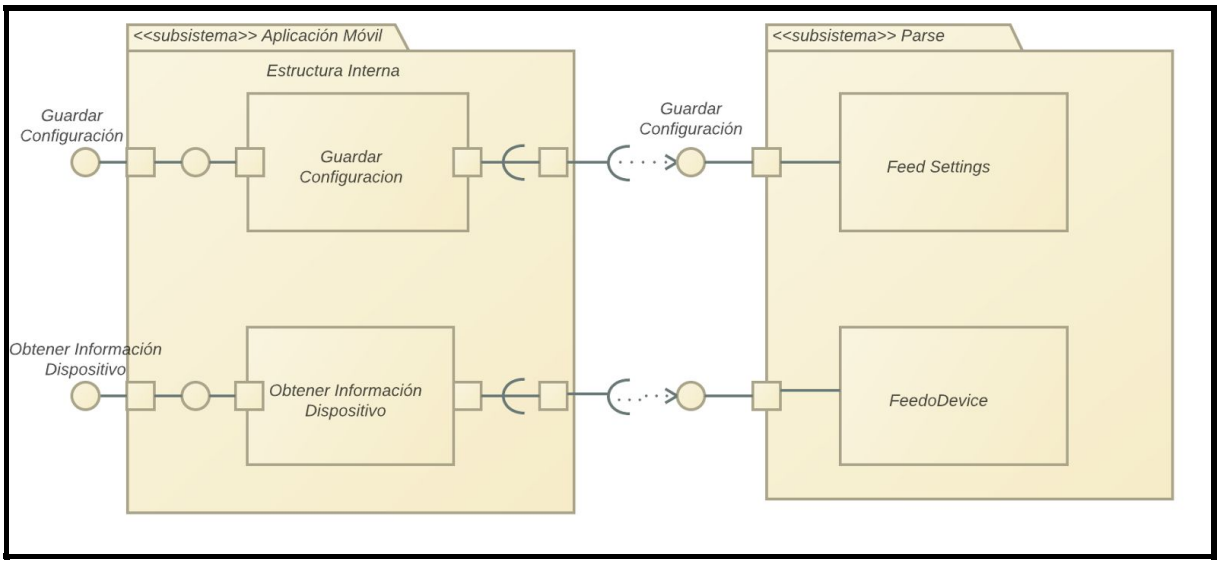


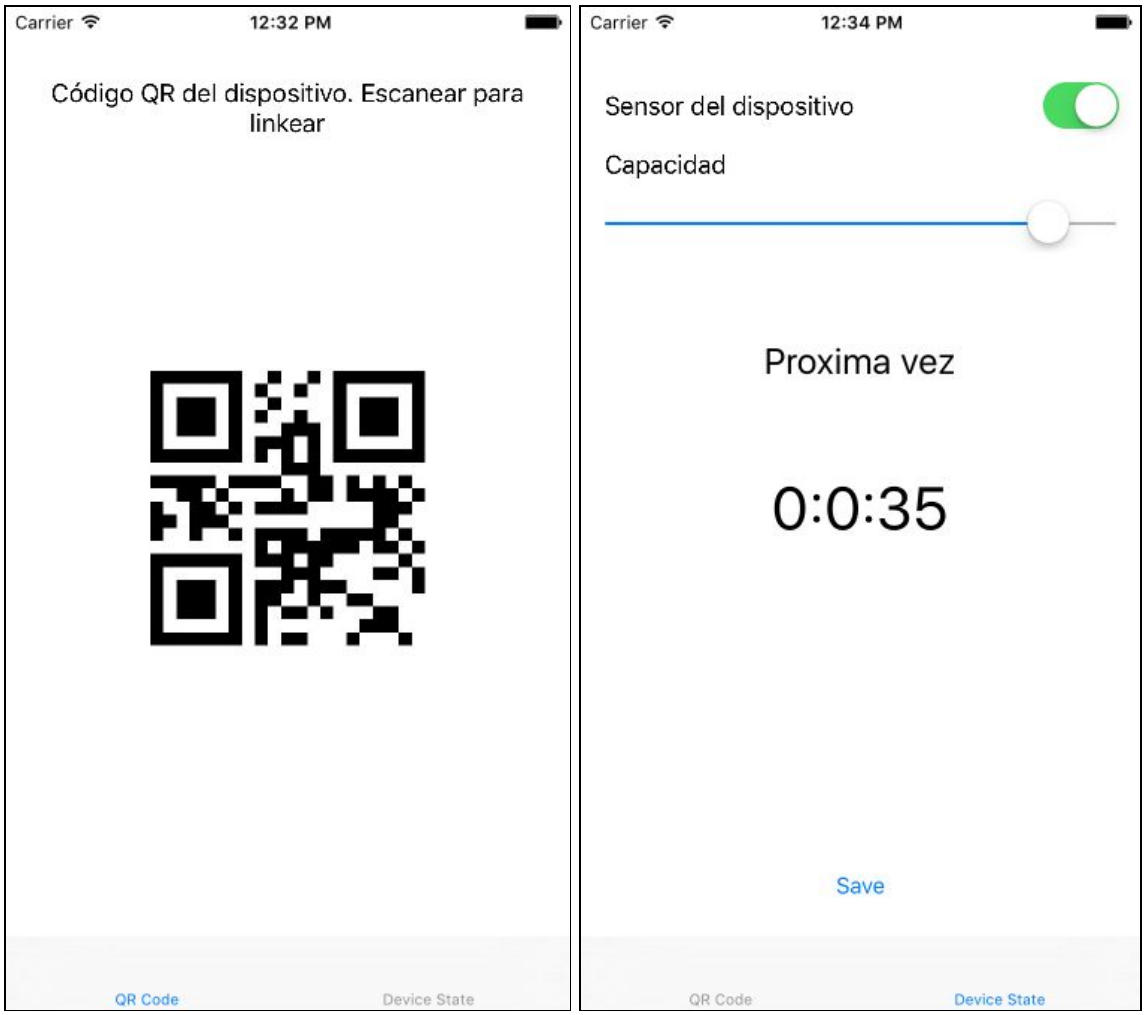
Diagrama de Componentes



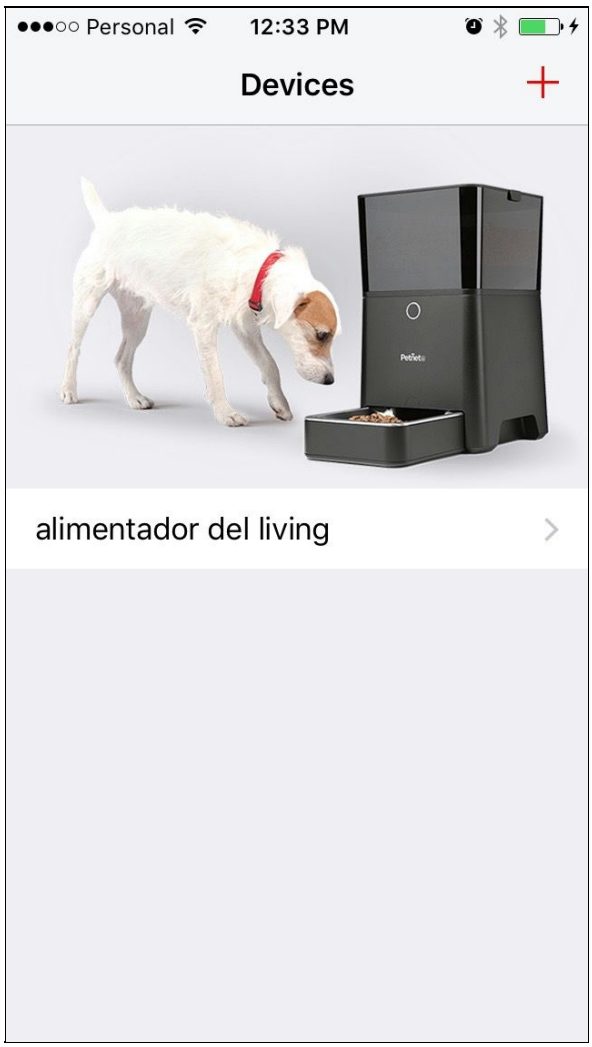
Casos de Prueba

Linkear un dispositivo

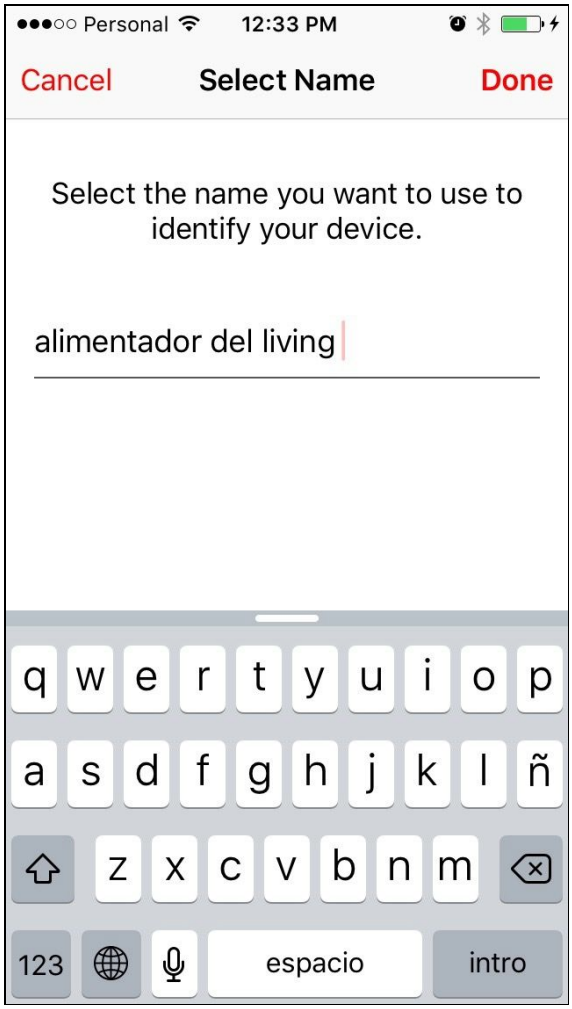
1. En el simulador de alimentador, se muestra el código qr para linkear. En el segundo tab pueden observarse los datos simulados del dispositivo: estado del sensor, capacidad.

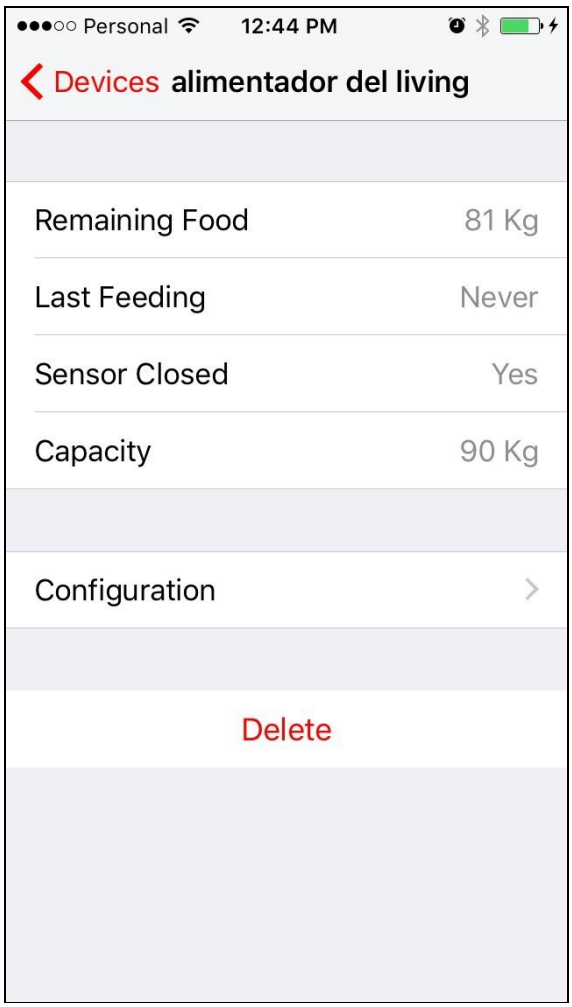


2. En la aplicación presionar + para agregar un dispositivo



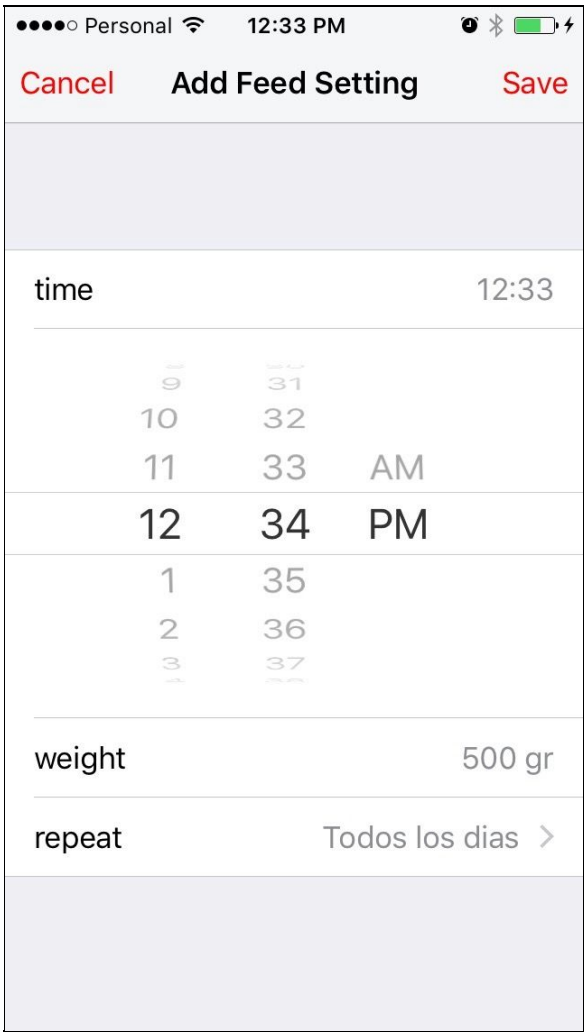
3. **Luego de elegir el nombre ver que los datos coinciden con los del simulador**

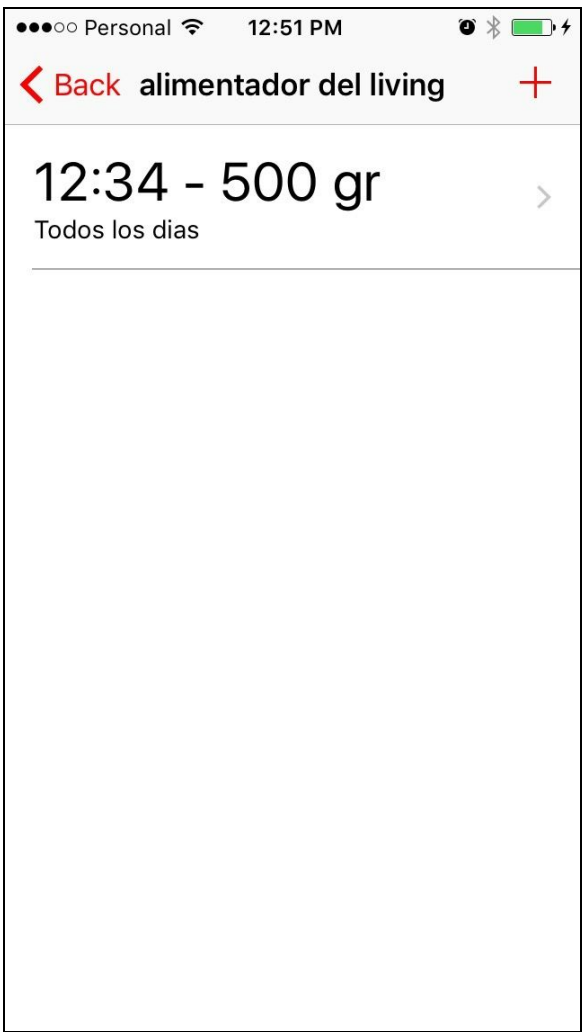




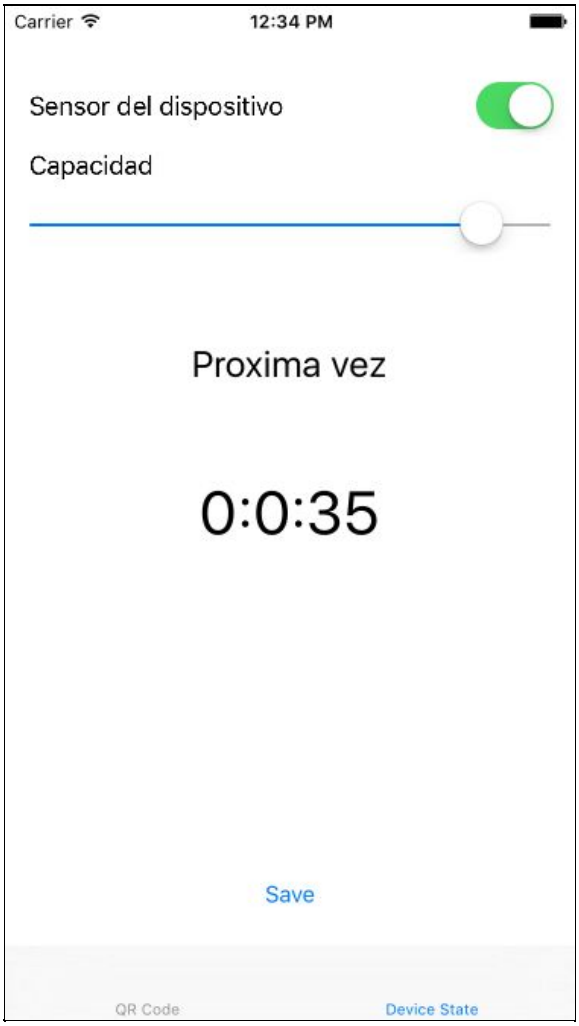
Configurar hora de alimento y verificar su ejecución

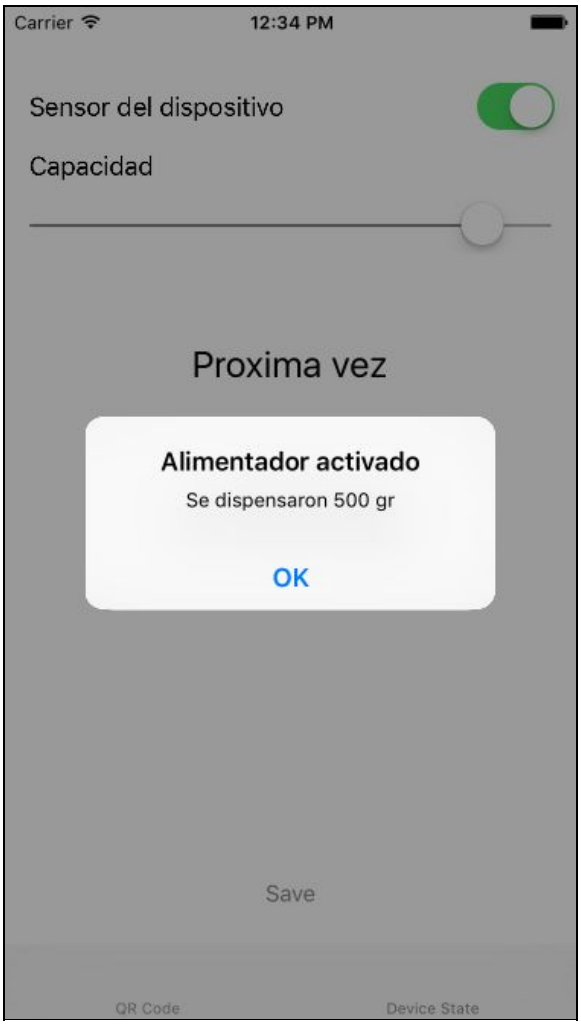
1. **Configurar horarios de alimento en la aplicación**



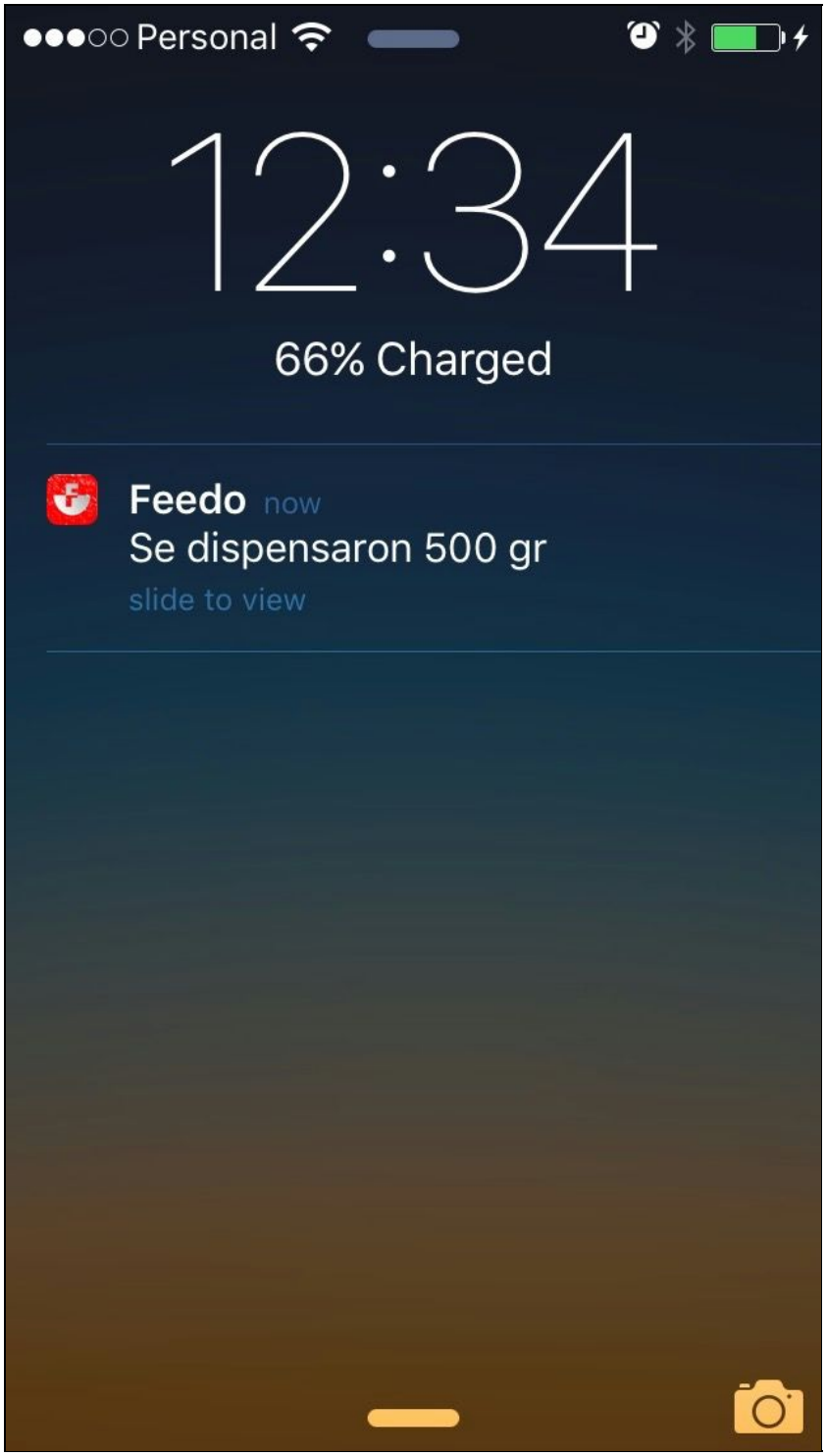


2. Verificar en simulador que se alimenta en ese horario





3. Verificar que llega notificación a la aplicación cliente con la noticia de que se dispensó la comida



Futuras Líneas de trabajo

Quedan abiertas fundamentalmente tres líneas de futuro trabajo, por un lado desarrollar un dispositivo real que sirva mecánicamente el alimento, por otro lado extender el alcance de los clientes a más dispositivos con el desarrollo de una versión HTML, por otro lado mejoras a la funcionalidad actual

Prototipo del dispositivo dispensador

El dispositivo estará basado en la placa Arduino, estas placas ofrecen la posibilidad de conexión wi-fi y además un buen número de entradas y salidas que se usarán para manejar el motor y obtener información de los sensores.
Parse ofrece un SDK para placas Arduino YUN facilitando así la integración

Versión HTML

La versión HTML será desarrollada en AngularJS y podrá ser utilizada tanto desde computadoras desktop como smartphones y tablets.
Usando la estructura REST de la API la integración con este tipo de aplicaciones cliente en Javascript es muy conveniente y se logran muy buenos resultados.

Nuevas Funcionalidades

Una vez concretado el dispositivo físico podrá hacerse uso de este soporte para mejorar la interacción con la mascota.
Dentro de las líneas de mejora se contemplan:

- Detección de mascota mediante wearable RF: de esta forma se podrá identificar a la mascota a la cuál dispensar si hubiera varias.
- Captura de fotos en el momento que se está alimentando
- Poder interactuar a distancia con la mascota mediante voz, luces y sonidos emitidos por el dispositivo.
- Notificaciones cuando cambia un estado crítico del dispositivo. Poder configurar las notificaciones que deseo recibir en este sentido.
- Notificación de que comió o no la comida, luego de cierto tiempo de dispensado.

Anexo 1: Estado del arte

Qpets AF 200
http://www.catfooddispensersreviews.com/qpets-af-200-pet-feeder-review/

Existe una amplia variedad de dispensadores automáticos de alimento, éstos son en su gran mayoría programados desde un panel poco amigable en el mismo dispositivo y fija un cronograma en el cual se va a dispensar para todos los días.
Estos dispositivos compiten con nuestra solución pero son conceptualmente distintos ya que no ofrecen un dispositivo conectado a internet (IoT), por este motivo se incluye uno solo de estos dispositivos representando la categoría.
Este dispensador es programable desde el mismo dispositivo, soporta hasta 4 comidas por día.

PetPal
http://gopetpal.com/

El dispositivo pareciera estar aún en fase, cuenta con wi-fi, cámara y una aplicación mobile para monitorear la cámara y configurar los horarios en los cuales debe dispensar.

Debilidades

- Pareciera estar aún en desarrollo con lo cual no puede accederse a la app ni comprarse el dispositivo.
- La aplicación de configuración parece muy poco amigable y no tiene ningún dato del estado del equipo (cuanto alimento queda, cuanto sirvió, etc).

Feed and Go
http://www.feedandgo.com/

Tiene una forma diferente a la mayoría de los otros dispensadores, tiene seis porciones ya establecidas y el dispensador en el horario indicado gira para permitir a la mascota acceder.

Debilidades

- Tiene bastante limitada la cantidad de alimento (6 porciones)
- Las cantidades no son configurables desde la aplicacion y además hay que servirlas separadamente de forma manual.
- Sólo tiene versión web

Pet Station
http://www.catfooddispensersreviews.com/pet-station-automatic-feeder-with-built-in-camera-review/

Tiene el foco puesto en la interacción con la mascota en forma de videollamada, al llegar el momento de alimentar a la mascota la app ofrece mirar como come.

Debilidades

- No mantiene un registro de la actividad
- La única retroalimentación de lo que sucede con la mascota es la cámara en tiempo real.
- Sólo está en koreano
- La aplicación no está muy lograda.

Tabla Comparativa

Característica	**Qpets AF 200**	**PetPal**	**Feed and Go**	**Pet Station**	**PetFeeder IoT**
Programable (habilidad de programar cuando dispensar)	Si	Si	Si	Si	Si
Wi-Fi	No	Si	Si	Si	Si
App Android	No	Si	No	Si	Si
App iOS	No	Si	No	Si	Si
App Web	No	No	Si	No	Si
Cámara	No	Si	Si	Si	Si
Cantidad de alimento sobrante en la nube	No	No	Si	No	Si
Cantidad de alimento programable	No	Si	No	Si	Si
Registro de las veces en que se accionó en la nube	No	No	No	No	Si
Grabar audio para llamar a tu mascota	Si	Si	Si	No	Si
Video llamada	No	No	No	Si	No

Anexo 2: Bibliografía

Internet de las cosas en wikipedia: https://es.wikipedia.org/wiki/Internet_de_las_cosas
Scrum: http://scrummethodology.com/
Arduino: http://www.arduino.cc/
Web de desarrolladores de Apple: https://developer.apple.com/
Web de desarrolladores de Android: http://developer.android.com/
Web de AngularJS: https://angularjs.org/