
MYKENNEL - APLICACIÓN PARA LA GESTIÓN DE CRIADERO CANINO



TFC UOC- Desarrollo Aplicación iOS

Consultores: Almrall López, Jordi - Ceballos Villach, Jordi

Autor: Sánchez Carrillo, Juan Antonio

11 de marzo de 2015



En memoria de Eros, algún día nos volveremos a ver gordo!

A mis padres, por apoyarme durante toda la carrera y, sobretodo, por confiar en que la terminase.

Y a mi mujer, por aguantarme en los momentos difíciles y estar siempre ahí.

TFC UOC- DESARROLLO APLICACIÓN IOS	1
1.Definición General del Proyecto	6
Introducción	6
1.1.Objetivos Del Trabajo	6
1.2.Entregas	7
1.3.Marco Temporal (Productos Obtenidos)	7
1.4.Recursos E Infraestructura	7
1.4.1.Hardware	7
1.4.2.Software	8
2.Planificación del Proyecto	9
2.1.Planificación Temporal	9
2.2.Descripción De Las Actividades	10
2.2.1.Planificación Del Tfc	10
2.2.2.Análisis Y Diseño	10
2.2.3.Implementación Preliminar Y Test	10
2.2.4.Implementación Final Y Entrega	10
3.Análisis de Requerimientos	11
3.1. Consulta En Redes Sociales	11
3.2. Entrevistas	12
3.2.1.Entrevista 1	12
3.2.2.Entrevista 2	13
3.2.3.Entrevista 3	13
3.3. Conclusiones De Las Entrevistas	14
3.4. Análisis De Competencia	15
3.4.1.Breeding Master	15
3.4.2.Zooeasy	19
3.5. Requerimientos Funcionales	20
3.5.1.Descripción Básica Del Funcionamiento	20

3.6. Requerimientos No Funcionales	22
4.Análisis de Riesgos	23
5.Diseño Conceptual	26
5.1. Casos De Uso	26
5.2.Escenarios De Uso	27
5.2.1.Pantalla Principal	27
5.2.2.Gestión Pruebas De Salud	27
5.2.3.Gestión Prueba De Salud - Añadir/Modificar	28
5.2.4.Gestión Pruebas De Trabajo	28
5.2.5.Gestión Prueba De Trabajo - Añadir/Modificar	29
5.2.6.Gestión Resultados De Show	29
5.2.7.Gestión Resultado De Show - Añadir/Modificar	30
5.2.8.Gestión Propietarios	30
5.2.9.Gestión Propietarios - Añadir/Modificar	31
5.2.10.Gestión Perros	31
5.2.11.Gestión Perros - Añadir/Modificar	32
5.3. Descripción De Los Casos De Uso	33
5.3.1.Pantalla Principal (Pap)	33
5.3.2.Gestionar Propietarios (Gpr)	36
5.3.3.Gestionar Perros (Gpe)	36
5.3.4.Gestionar Pruebas De Salud (Gps)	37
5.3.5.Gestionar Pruebas De Trabajo (Gpt)	37
5.3.6.Gestionar Resultados De Show (Grs)	38
6.Prototipado	39
6.1. Pantalla De Carga	39
6.2. Pantalla Principal	40
6.3. Pantalla Gestión De Pruebas De Salud	41
6.4. Pantalla Gestión De Pruebas De Trabajo	42

6.5. Pantalla Gestión De Resultados De Show	43
6.6. Pantalla Detalle Perro	44
6.7. Pantalla Gestión De Perros	45
6.8. Icono De La Aplicación	46
7. Diseño Técnico	47
7.1. Modelo De Base De Datos	47
7.2. Storyboard	49
8. Implementación	50
8.1. Aplicación Desarrollada	55
9. Conclusion	55
10. Bibliografía	56

1.DEFINICIÓN GENERAL DEL PROYECTO

Introducción

Hace ya un tiempo que hablando con criadores caninos conocidos, me surgieron dudas de como gestionaban su propia cría, como llevaban un control de comidas, vacunas, reservas de cachorros, costes... etc. En todos los casos la respuesta fue la misma: "de cabeza".

Es por ello que me planteé hacer una búsqueda por internet de aplicaciones para la gestión de criaderos caninos , para ver si realmente no había nada para ayudar a los criadores. Me llevé una sorpresa al ver que si había aplicaciones para la gestión de criaderos, como ¹Breeders Assistant o ²ZooEasy, pero básicamente son bases de datos con formularios, no tienen una interfaz gráfica que sea intuitiva. En los casos probados, las herramientas son potentes, pero les falta una interfaz gráfica que haga su uso mucho más intuitivo.

Tras ver los resultados obtenidos en la búsqueda, decidí crear una aplicación desde cero, que sea fácil de gestionar, intuitiva y con la información necesaria. Es por esto que nace el proyecto MyKennel.

1.1.Objetivos del Trabajo

El objetivo de esta aplicación es cumplir con las necesidades de una gestión optima de un criadero canino de manera eficaz y eficiente, sin que esto ponga en peligro la interfaz intuitiva.

Esta aplicación debe ser capaz de, mediante la información proporcionada por el criador, de mostrar los datos referentes a cada camada como: porcentaje de consanguinidad, número de cachorros, datos de los padres, etc. Dentro de cada ejemplar, incluirá los datos referentes a ese ejemplar como: Fecha de nacimiento, pruebas de trabajo, pruebas de salud, resultados de exposiciones, etc.

En cuanto a tecnología, usaremos ³Xcode como herramienta de desarrollo, SQLite para la creación de la base de datos y ⁴Pixate y ⁵Sketch para el apartado gráfico; crearemos una app universal para iOS.

Cabe destacar que este proyecto es un reto personal para su autor, debido a que parte sin experiencia en la programación en swift, así como en el desarrollo gráfico, por lo que alguna funcionalidad puede no estar completa.

¹ <http://www.tenset.co.uk/ba/index.html>

² <http://www.zooeasy.com>

³ <https://developer.apple.com/xcode/>

⁴ <http://www.pixate.com>

⁵ <http://bohemiancoding.com/sketch/>

1.2. Entregas

Las Entregas previstas para este Trabajo Final de Carrera son las siguientes:

- Plan de Trabajo (PAC1): Podremos ver la justificación de este trabajo, los objetivos, la estrategia a seguir y la planificación del trabajo.
- Diseño (PAC2): Deberemos realizar la interfaz de usuario, la arquitectura y el diseño de datos.
- Implementación (PAC3): Una vez tengamos la interfaz, deberemos disponer de una implementación de la app, que muestre las soluciones a los objetivos propuestos.
- Entrega Final (Final): En esta última entrega, deberíamos de tener un producto terminado, la memoria final y una presentación de nuestra aplicación.

1.3. Marco Temporal (Productos Obtenidos)

El plazo establecido para la entrega de las partes de este Trabajo Final de Carrera (PAC1, PAC2, PAC3 y Entrega Final) es el siguiente:

- Entrega del Plan de Trabajo (PAC1): 11-03-2015.
- Entrega del Diseño de la App (PAC2): 08-04-2015.
- Entrega de la Implementación preliminar (PAC3): 20-05-2015.
- Entrega de la Memoria Final y Presentación: 21-06-2015.

1.4. Recursos e Infraestructura

En este apartado se detallan los recursos necesarios para poder llevar a cabo el Trabajo Final de Carrera.

1.4.1. Hardware

Para la creación de esta aplicación, utilizaremos los siguientes equipos:

- MacBook Pro: S.O. Mac OS X Yosemite (10.10.3), procesador core i5 con 2,3 GHz, 8 GB de RAM, Tarjeta gráfica integrada Intel HD 3000, 500 GB de disco duro, pantalla de 13 pulgadas con 1280 x 800 píxeles de resolución.
- iPhone 5S: iOS 8.2, procesador A7, 16GB de memoria interna, pantalla de 4 pulgadas con 1134 x 640 píxeles de resolución.
- iPhone 6: iOS 8.2, procesador A8, 16GB de memoria interna, pantalla de 4,7 pulgadas con 1334 x 750 píxeles de resolución.

1.4.2. Software

Para la creación de esta aplicación, utilizaremos las siguientes herramientas:

- Xcode: Herramienta para el desarrollo de aplicaciones para Mac OS X e iOS
- ⁶Sketch: Herramienta con plantillas predefinidas para la interfaz de iOS, usada para hacer las capturas y el diseño de pantalla de la aplicación
- ⁷Pages: Herramienta utilizada para la elaboración de los documentos de este Trabajo Final de Carrera.
- ⁸Flinto: Herramienta descubierta al final del proyecto, muy buena para el prototipado, siendo posible incluso correr el prototipo directamente en el dispositivo.

⁶ <http://bohemiancoding.com/sketch/>

⁷ <https://www.apple.com/mac/pages/>

⁸ <https://www.flinto.com>

2.PLANIFICACIÓN DEL PROYECTO

2.1.Planificación Temporal

Podremos ver la planificación de las diferentes tareas del proyecto dentro de la tabla siguiente:

Tarea	Fecha de Inicio	Días Previstos
PAC 1	25/02/2015	15 Días
Lectura de Materiales facilitados	25/02/2015	4 Días
Análisis del Plan de Trabajo	27/02/2015	3 Días
Elaboración del Plan de Trabajo	29/02/2015	8 Días
Entrega del Plan de Trabajo (PAC 1)	11/03/2015	0 Días
PAC 2	12/03/2015	28 Días
Análisis real de las funcionalidades de la aplicación	12/03/2015	3 Días
Diseño de la Base de Datos	15/03/2015	8 Días
Diseño de la Interfaz de Usuario	22/03/2015	14 Días
Creación de un Prototipo	05/03/2015	3 Días
Entrega de la PAC 2	08/04/2015	0 Días
PAC 3	09/04/2015	38 Días
Implementación Preliminar	09/04/2015	36 Días
Test de la App	18/04/2015	2 Días
Entrega de la PAC 3	20/05/2015	0 Días
Entrega Final	21/05/2015	30 Días
Implementación Final	21/05/2015	20 Días
Revisión Final de la Memoria	09/06/2015	5 Días
Creación de presentación	14/06/2015	5 Días
Entrega Final	21/06/2015	0 Días

Las fechas en negrita, al ser fecha de entrega son fijas, sin embargo, el resto de fechas no son fijas y pueden moverse debido a necesidades del proyecto.

2.2.Descripción de las Actividades

Las principales actividades que compondrán el proyecto son las siguientes:

2.2.1.Planificación del TFC

En esta primera parte, se realiza este documento, donde se muestran las motivaciones, funcionalidad deseadas y se realiza una planificación del proyecto durante todo el semestre.

2.2.2.Análisis y Diseño

En esta segunda parte, procedemos a realizar un análisis real de la funcionalidad de cada característica de la aplicación, de la dificultad de su implementación y de su viabilidad para implementarla dentro del plazo marcado.

Asimismo, esta parte incluye el diseño de la interfaz de usuario preliminar, buscando ante todo la usabilidad. Al ser una aplicación que trata datos relacionados, deberemos realizar una base de datos, donde tengamos toda la estructura que necesitamos para la aplicación final.

2.2.3.Implementación Preliminar y Test

En esta tercera parte, deberemos de implementar una primera versión de lo que será nuestra aplicación final y, mediante pruebas, avanzar hasta lo que será nuestra aplicación final.

2.2.4.Implementación Final y Entrega

La última entrega es quizá la más importante, ya que deberemos hacer video demostrando el funcionamiento de nuestra aplicación, con posibles casos de uso y mostrando la usabilidad real de nuestra aplicación.

Es en esta parte donde deberemos realizar la entrega de la presentación mencionada, de la memoria final y de nuestra aplicación.

3. ANÁLISIS DE REQUERIMIENTOS

Al ser una aplicación centrada en el usuario (Diseño Centrado en el Usuario - DCU), hemos utilizado unas técnicas para obtener información sobre los requisitos que debe cumplir nuestra aplicación.

Nos hemos basado en Entrevistas y hemos utilizado las redes sociales como una encuesta, pero que acabó siendo una lista de funcionalidades deseadas.

Básicamente, con esta técnica, lo que buscamos es obtener información valiosa para nuestro diseño, como deseos, motivaciones, valores y experiencias de nuestros futuros usuarios. Al no tener una aplicación final, el feedback que podríamos obtener viendo como usan la aplicación no lo podemos obtener, por lo que una vez completada la aplicación, deberemos volver a ver a los entrevistados, para evaluar la usabilidad de la aplicación.

3.1. Consulta en Redes Sociales

Lo primero que hice, fue exponer directamente en un grupo del que formo parte de Facebook, que quería hacer una app para criadores y preguntar a estos sobre las características que debería de tener una app para la gestión de sus criaderos. Las respuestas recibidas son bastante interesantes, ya que he descubierto gracias a ellos la posible competencia que podemos tener, así como las principales características que quieren.

A continuación, voy a poner unas capturas de los comentarios recibidos en el grupo de Facebook:

Grím Black Que si cria más de 3 razas no pueda salir en la APP! Lo que sea para hacer un filtro anti-juntaperros y que no acabe como el listado de mundoanimalia
10 de marzo a la(s) 6:42 · Me gusta · 3

Schnauzerin's Paradise Posibilidad de ver pedigrees y galardones de padres ver alimentación y detalle de vacunación y desparasitación así como poder ver las garantías que se dan por la compra del cachorro.
10 de marzo a la(s) 15:39 · Me gusta · 3

Grím Black Y las pruebas genéticas
10 de marzo a la(s) 21:04 · Me gusta · 1

Adrian Civera MichBullykennel Buena idea!!!
11 de marzo a la(s) 23:29 · Me gusta

Adrian Civera MichBullykennel Ver instalaciones y forma de tratar sus perros y crías en cuanto a alimentación, vacunación, seriedad, si cumple las garantías etc
11 de marzo a la(s) 23:30 · Me gusta · 1

Mariam Poveda Gomez Super acertado!! Ojalá puedas llevar a cabo todas estas geniales ideas que te comentan, a mí poco más se me ocurre, pero seríamos muchos los que la usarían, quizás que tuviera la posibilidad de dejar mensajes a través de la aplicación entre criador y dueño... Ver más
11 de marzo a la(s) 23:36 · Editado · Me gusta · 2

Ayeichen Sandra Roldan Gran idea!!!
11 de marzo a la(s) 23:37 · Me gusta · 1

Mariam Poveda Gomez Ah! Y alguna plantilla para introducir pedigrees completos!! Por pedir que no quede, pero el pedigree para los criadores es muy básico y fundamental, así que una plantilla con sus casillas en determinados colores según si es campeón o no, etc.. Estaría genial!
11 de marzo a la(s) 23:40 · Me gusta · 1

Michelle Csuk Un calculador de cuando nacerá la camada, una posibilidad de apuntar a diario el peso de los cachorros y poder ver una estadística del crecimiento de los cachorros. Una posibilidad de apuntar cambios en la madre durante el embarazo y la gestación, tablas orientativas sobre aumento medio de peso en los cachorros, cantidad media de leche que debe tomar en cada toma...
Una posibilidad de apuntar la cantidad de ML bebidos si surge la necesidad de criar uno o todos los cachorros a biberón...
Sin las primeras cosas que me ocurren.
10 de marzo a la(s) 7:19 · Me gusta · 5



Norteynada Kennel Fechas de nacimiento de reproductores . Cuando han tenido el celo promedio de cachorros . Montas y número de cachorros de un semental altura y peso adulto de cada uno
10 de marzo a la(s) 9:37 · Me gusta · 2



Norteynada Kennel Asignar a cada cachorro un dueño con sus datos
10 de marzo a la(s) 9:38 · Me gusta · 3


Blanca Azalea CÁLCULOS DE PORCENTAJES DE CONSANGUINIDAD
10 de marzo a la(s) 10:13 · Me gusta · 6

Manuel Monsell Prado Control de camadas, con registros, fechas, nombres, propietarios, fotos, etc...buena ideaaaaa!!!
10 de marzo a la(s) 10:40 · Me gusta · 3

Santiago Antonio Falcon Sanchez Calendarios de vacunación de los ejemplares o cachorros
10 de marzo a la(s) 10:57 · Me gusta · 3

 **Michelle Csuk** Un calculador de cuando nacerá la camada, una posibilidad de apuntar a diario el peso de los cachorros y poder ver una estadística del crecimiento de los cachorros. Una posibilidad de apuntar cambios en la madre durante el embarazo y la gestación, tablas orientativas sobre aumento medio de peso en los cachorros, cantidad media de leche que debe tomar en cada toma...
Una posibilidad de apuntar la cantidad de ML bebidos si surge la necesidad de criar uno o todos los cachorros a biberón...
Sin las primeras cosas que me ocurren.
10 de marzo a la(s) 7:19 · Me gusta ·  5

 **Concepción Álamo Arcness-Kennel** Sería muy bueno poder llevar en el móvil un programa del criadero... celos, partos, pedigree, propietarios, etc... Así lo tendríamos siempre a mano y no habría que esperar a llegar a casa para consultar nada. Y si es, con posibilidad de poner, en una nube (para cogerlo y añadir luego a tu gestor del criadero en el disco de tu ordenador), sería la repanocha. Ya sé...pido mucho.
12 de marzo a la(s) 8:56 · Editado · Me gusta ·  3

 **Jose Villaferron Amstaff** Un breeding master en APP, que genere un pdf o jpg en plan ficha del ejemplar, posibilidad de compartir esa ficha. Yo lo uso hace tiempo como base de datos y no va mal, todo es mejorable pero metiendole info es una buena herramienta..
12 de marzo a la(s) 10:01 · Me gusta ·  1

Como vemos, la variedad de respuestas no ayuda mucho a la hora de seleccionar las principales características que debe cumplir la app. Debemos de tener en cuenta uno de los principales elementos que nos perjudican: La cantidad de información a mostrar.

No debemos olvidar que la pantalla de nuestro teléfono es limitada y que la cantidad de información a mostrar en la pantalla debe ser la justa y necesaria para desarrollar la tarea que necesitamos. De nada sirve que creamos una app con 100 funcionalidades, si a la hora de mostrarlo por pantalla parece un galimatías.

3.2. Entrevistas

A continuación, vamos a ver las entrevistas que realicé a tres criadores de diferentes razas que conozco. A estos, solo les formulé tres preguntas:

- **¿Qué te parecería tener la información básica de tu criadero en el teléfono?.**
- **¿Qué información consideras como básica, como indispensable, para que esta app se vuelva un indispensable en tu teléfono?.**
- **¿Que teléfono tienes?**

3.2.1. Entrevista 1

¿Nombre?: Manuel SanMartin

¿Raza que crías?: Akita Americano

¿Qué te parecería tener la información básica de tu criadero en el teléfono?.

Me parece muy buena idea, en muchas exposiciones encuentras a gente que te pide información sobre tus perros, así, no solo no tienes que acordarte de todo lo que tienen todos los perros, sino que además, creo que queda bastante profesional

¿Qué información consideras como básica, como indispensable, para que esta app se vuelva un indispensable en tu teléfono?.

Que me de más a conocer... jajajaj...Ahora en serio, pare que se vuelva indispensable, me debe proporcionar la información de cada ejemplar de los que tengo, pedigree, pruebas de salud y resultados de show. Con esa información me quitaría mucho papeleo que llevar en el coche, por si sale alguna monta o me gusta alguna hembra, poder enseñar al propietario de la hembra el palmarés y las pruebas de mi perro.

¿Que teléfono tienes?

Un Samsung Galaxy S3.

3.2.2. Entrevista 2

¿Nombre?: Arturo Mendez

¿Raza que crías?: Boxer

¿Qué te parecería tener la información básica de tu criadero en el teléfono?.

Juanillo, sabes que soy muy malo con las tecnologías, lo mío con la informática y los teléfonos ya no es tratable, aunque con un cursillo practico de uso, le daría una oportunidad ya que me parece buena idea.

¿Qué información consideras como básica, como indispensable, para que esta app se vuelva un indispensable en tu teléfono?.

Pedigree de los perros, pruebas de trabajo, pruebas de salud y posibilidad de tener la información de los propietarios de cada perro. La agenda de mi teléfono es un cúmulo de nombres de perros con nombres de sus dueños, y ya he perdido varios contactos al cambiar de un teléfono a otro.

¿Que teléfono tienes?

Un Nokia 5800.

3.2.3. Entrevista 3

¿Nombre?: Alejandro Freire

¿Raza que crías?: Dobermann

¿Qué te parecería tener la información básica de tu criadero en el teléfono?.

Perfecto!!! Como sabes, aún no he realizado ninguna camada, estoy seleccionando la línea que quiero, moviendo perros por los rings, etc. Me parece perfecto el partir ya con una base informatizada.

¿Qué información consideras como básica, como indispensable, para que esta app se vuelva un indispensable en tu teléfono?.

Líneas de sangre de los perros, pruebas de trabajo, de salud, expos... Posibilidad de tener los datos del chip de cada perro; cuando tenga camadas, me gustaría llevar un seguimiento de los perros, de sus propietarios, de lo que van sacando, para, si llega el momento, poder usarlos como ejemplares en mi criadero.

¿Que teléfono tienes?

Un iPhone 6.

3.3. Conclusiones de las entrevistas

Es curioso ver, como a la hora de realizar las entrevistas y hacer hincapié en la palabra “teléfono”, las funcionalidades básicas no son tantas como las recibidas via redes sociales. Da la impresión de que toma más partida el tamaño de la pantalla y disminuyen los requisitos.

Bajo mi punto de vista, sin ser criador, pero con unos cuantos años de experiencia con perros a mis espaldas, veo que aquél documento que me había hecho en su día para llevar un control de las pruebas que le hacía al perro, de los resultados de las exposiciones, links de interés e información de los padres; no dista mucho de lo que debe de cumplir la app como información básica.

Como ya he dicho en otros puntos, ya hay aplicaciones de escritorio con un sinfín de opciones, que cumplen con el cometido de esta app. Sin embargo, lo que hace única la app de este proyecto, es la sencillez de uso objetivo. Todas las aplicaciones que hemos probado tienen un diseño parecido a un “formulario de access”, por lo que nuestra mayor baza debe estar en un diseño lo más básico e intuitivo posible.

Finalmente, he decidido que las funcionalidades de la app sean las siguientes:

- *Registro de Perros: Alta, Baja y Modificación de Ejemplares Caninos. Añadir pruebas de salud, trabajo y belleza por perro, añadir propietario, añadir padre/madre...*
- *Registro de Propietarios: Alta, Baja y Modificación de Propietarios.*
- *Registro de Pruebas de Salud: Alta, Baja y Modificación de pruebas, con fecha y resultado.*
- *Registro de Pruebas de Trabajo: Alta, Baja y Modificación de pruebas, con fecha y resultado.*
- *Registro de Resultados de Show: Alta, Baja y Modificación de pruebas, con fecha y resultado.*

3.4. Análisis de Competencia

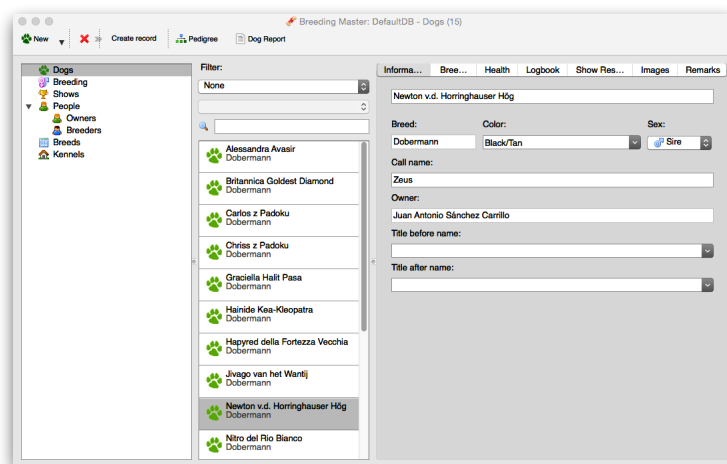
Tras tomar la decisión del tipo de aplicación a desarrollar, debemos de realizar un análisis de la competencia encontrada.

Para iOS, no hemos encontrado aplicaciones que tengan la misma finalidad que la nuestra, pese a que hay en el mercado un gran número de aplicaciones para llevar un diario de: Vacunas administradas, Celos, Deposiciones, Alimentación...etc. Son todas destinadas a particulares y no a criadores.

Para Mac/PC, si hemos encontrado aplicaciones destinadas al mismo propósito que la nuestra; haremos un breve análisis de una de las aplicaciones encontradas, ya que es la única que hemos encontrado para MAC

3.4.1. Breeding Master

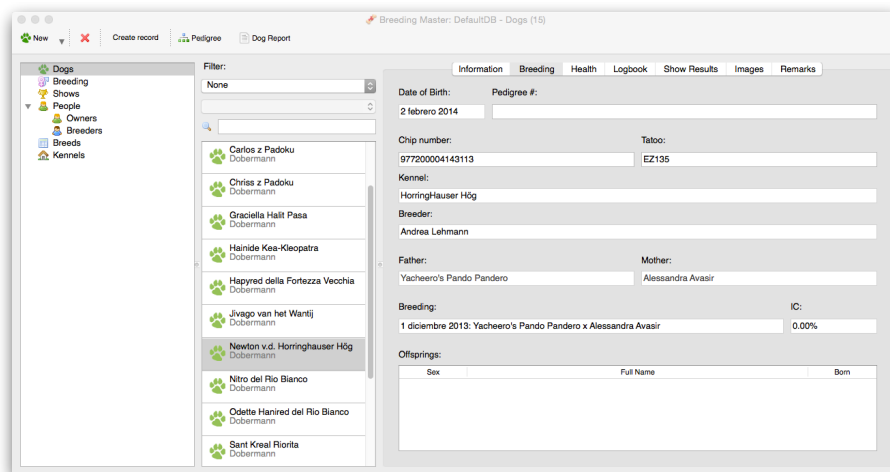
Esta aplicación es la única que hemos encontrado que sea compatible con Mac, por lo que ha sido la única que hemos podido probar en nuestro equipo.



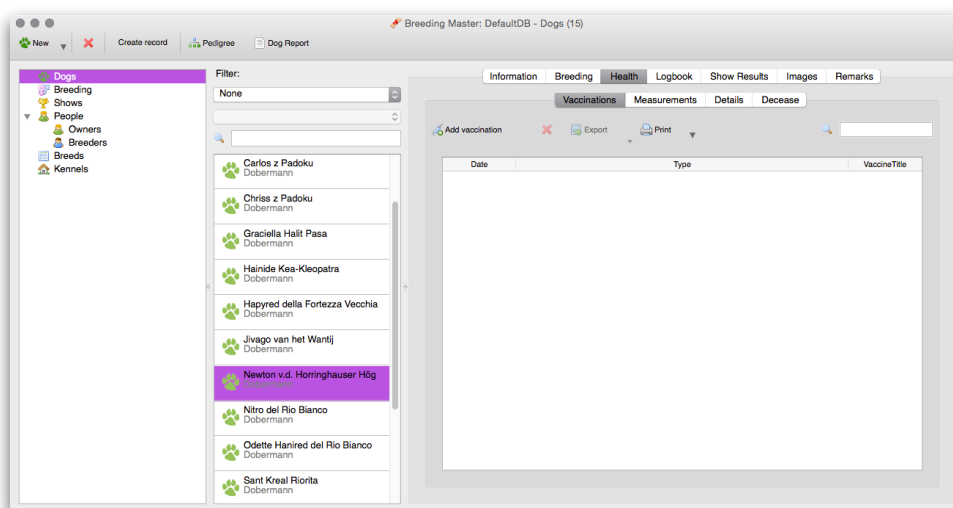
Esta sería la pantalla principal, en ella podemos agregar/eliminar perros, y ver la información de cada uno de ellos.

En el panel del lateral izquierdo, tenemos la navegación de la aplicación, donde encontramos: Perros, Montas, Shows, Gente (Propietarios y Criadores), Razas y Afijos.

El aspecto visual deja mucho que desear, así como el uso en si. Básicamente es un formulario de base de datos donde vas introduciendo datos. No llama la atención a primera vista, pese a que la información que es capaz de almacenar si resulta interesante.

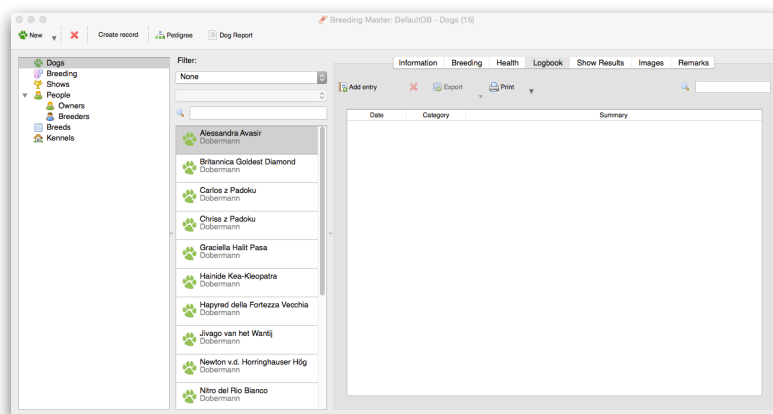


En esta captura, podemos ver más información del perro, como su propietario, afijo, padre y madre, coeficiente de consanguinidad, chip...



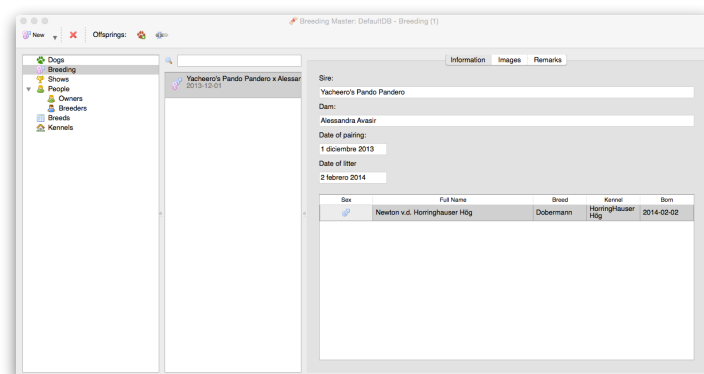
Aquí podemos ver el apartado de salud, tenemos la opción de introducir vacunas, tratamientos, detalles (grupo sanguíneo, intolerancias...etc) y fecha de defunción.

En este caso, consideramos que la información no es suficiente, ya que faltaría una sección donde poder introducir las pruebas de salud médicas como displasia, wobblers, willenbrand... Para mi, es más importante este tipo de pruebas, que las vacunas o tratamientos, ya que influyen mucho más a la hora de decidir una monta, o de seleccionar a un ejemplar para ella.

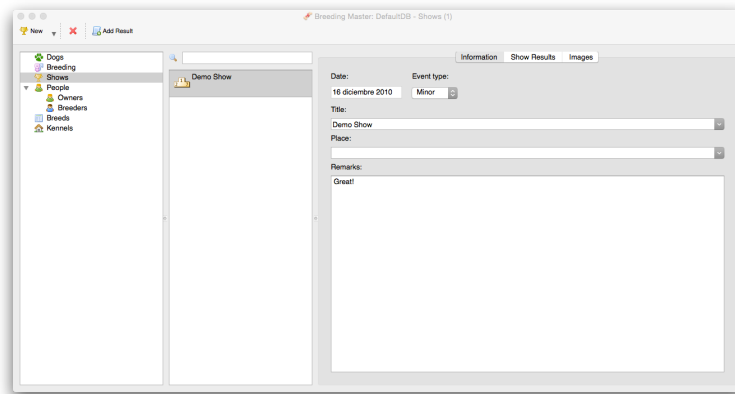


En esta captura podemos ver otra sección interesante, el apartado de shows. La información es la misma que la que quiero usar en la aplicación, como: Juez, Clase, Resultado, Fecha...

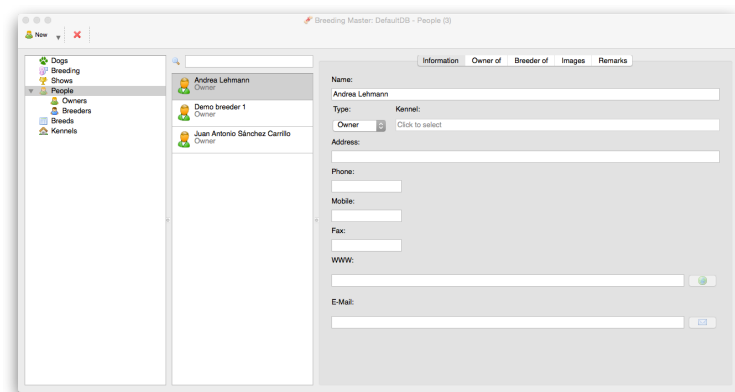
Dentro de cada perro, puedes ver todos los resultados y, además, si nos vamos a la sección de Resultados del panel de navegación, podemos ver todos los resultados de todos los perros.



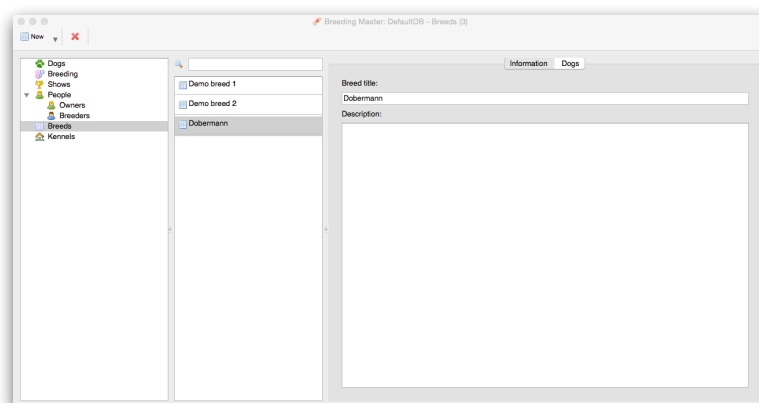
Aquí tenemos la sección de Breeding, que nos da la opción de hacer "montas". En ella podemos meter la fecha de la monta, el padre y la madre de la camada, y la fecha de nacimiento. Dentro de esta sección, podemos insertar imágenes de la camada y notas.



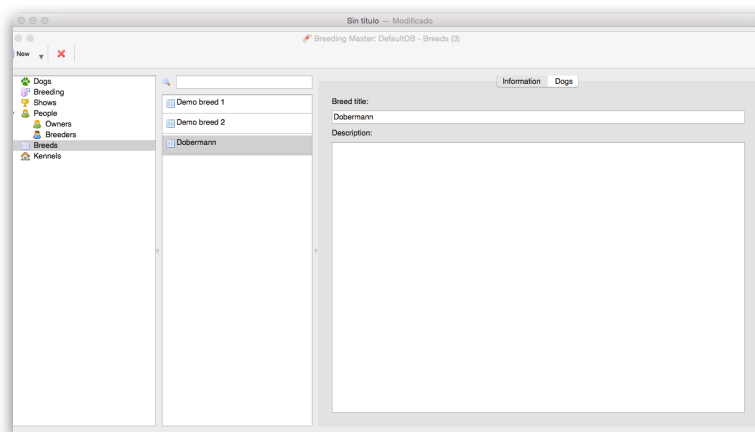
Aquí tenemos la sección de “Shows”, dentro de esta sección, podemos añadir los resultados de las exposiciones y concursos, diferenciándolos por tipo de evento (Mayor y menor). Podemos también insertar la fecha, el juez y el resultado obtenido e imágenes.



En la sección de “Gente”, podemos diferenciar si son criadores o propietarios, y tener los datos de contacto de cada persona. Estos registros, los podemos crear directamente al crear un perro y posteriormente rellenar la información en esta sección.



La sección “Raza” es sencilla, simplemente sirve para insertar raza. Una vez defines la raza en cada perro, puedes ver un listado general en la pestaña “Dogs”



Esta es la última sección de la aplicación, la sección de “Criadores (Kennel). En esta sección podemos llevar un registro de los criadores que forman parte de nuestra línea, con información de contacto, página web, etc. Además, una vez insertado un perro de un criador, nos aparecería listado dentro de esta sección en la pestaña “Dogs”

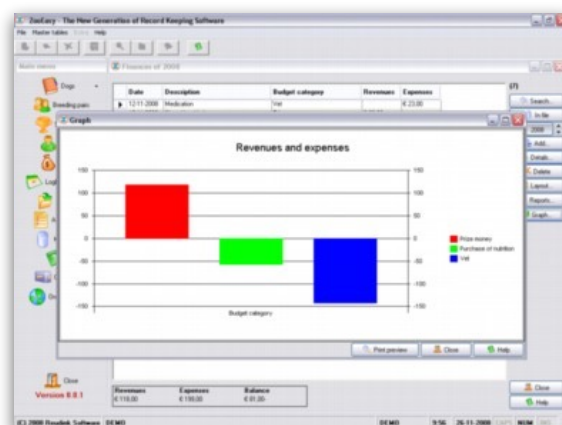
Y esto es todo lo que nos da esta aplicación; como cantidad de información, considero que es capaz de cumplir con la mayoría de las funciones solicitadas por los criadores, sin embargo, también considero que su interfaz gráfica dista mucho de ser intuitiva y atractiva. Tengo la sensación de estar manejando un formulario de una base de datos.

Puntos Fuertes: La información total que puedes manejar con la aplicación; una vez tienes los ejemplares metidos, ya aparecen para seleccionar como padre/madre de un nuevo ejemplar.

Puntos débiles: El diseño; el porcentaje de consanguinidad no está muy claro.

3.4.2.ZooEasy

De esta aplicación no hemos podido hacer pruebas, ya que no está disponible para sistemas Mac, sin embargo, hemos encontrado capturas por internet, que nos dan a entender que tanto la interfaz como la información es la misma en en la anterior aplicación. Nos encontramos el mismo problema gráfico, un formulario de base de datos.



3.5. Requerimientos Funcionales

La aplicación que estamos creando, tiene como finalidad ser un diario o agenda para el criador canino.

Buscamos evitar que el criador lleve toda la documentación de cada perro, como su palmarés, pruebas de salud, de trabajo, de belleza... Mediante nuestra aplicación, toda la información estará a la alcance de la mano, sin tener que llevar un dossier de cada perro.

3.5.1. Descripción Básica del funcionamiento

Intentaremos explicar brevemente cada una de las posibilidades de nuestra aplicación, para así poder tener una idea más clara de sus funcionalidad y de su funcionamiento.

3.5.1.1. Gestión de Perros

En este apartado, el usuario será capaz de añadir perros; para ello, deberá de insertar el nombre del perro, el afito, su número de chip, el propietario y el criador, además, podrá insertar una imagen descriptiva, para hacer la gestión más visual.

Una vez dado de alta el perro, podremos añadir pruebas de salud, de trabajo y de belleza.

También podremos modificar registros y eliminarlos.

3.5.1.2. Gestión de Personas

En este apartado, podremos dar de alta tanto criadores como propietarios; solo deberemos insertar: Nombre, Apellidos, Dirección y Teléfono.

Además, deberemos marcar si la persona es criador o no.

Una vez insertada la persona, podremos ver si posee algún perro, ya que la información tanto de propietario como de criador, estará enlazada con el apartado de Gestión de Perros.

A mayores, podremos modificar los registros así como eliminarlos.

3.5.1.3. Gestión de Pruebas de Salud

En este apartado, daremos de alta las pruebas de Salud, con Nombre de la prueba, Fecha, Resultado y Perro al que hace referencia.

Al igual que en el resto de apartados, una vez tengamos introducidos los registros, podremos acceder a modificar o eliminar registros.

3.5.1.4. Gestión de Pruebas de Trabajo

Este apartado es muy parecido al anterior; a mayores de Nombre, Fecha, Resultado y Perro, necesitaremos el de Juez.

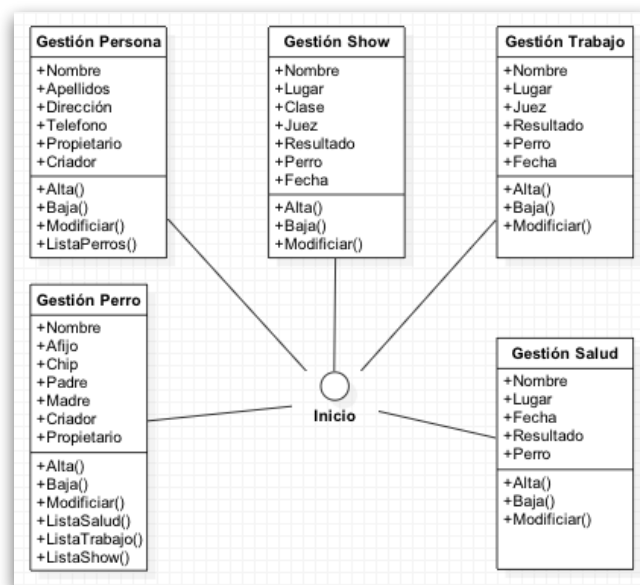
También podremos modificar y eliminar registros.

3.5.1.5. Gestión de Resultados de Show

En nuestro último apartado, tendremos los resultados de las exposiciones de belleza, para dar de alta un registro, necesitaremos la siguiente información: Nombre, Fecha, Clase, Resultado, Perro y Juez.

Con el atributo Clase, no nos referimos a si es Exposición o Concurso, sino a la clase en la que está registrado el perro (Abierta, Intermedia, Cachorros, Trabajo, Campeones..).

Finalmente, una imagen de como sería:



3.6. Requerimientos No Funcionales

Para el correcto funcionamiento de esta app, tendremos los siguientes requerimientos:

- La App estará programada en Objective-C/Swift para iPhone con Xcode, para la última versión de iOS disponible.
- Los gráficos serán lo más sencillos y representativos posible.
- No será necesario registro por parte del usuario final, y toda la información se guardará en su terminal mediante CoreData
- En primera instancia estará desarrollada en castellano, pero no se descarta que en futuras versiones se traduzca a otros idiomas.
- Al estar programado de manera universal, cualquier dispositivo iOS con la última versión debería de poder mover la app sin problemas., sin embargo, es posible que no se vea bien en todos al no utilizar autolayouts ni size classes

4. ANÁLISIS DE RIESGOS

En este apartado, veremos los riesgos a los que nos enfrentamos para realizar este proyecto.

Riesgo	Descripción	Probabilidad de Aparición	Impacto	Acciones
Avería del hardware	Avería de cualquiera de los componentes de hardware del proyecto, como iPhone y Mac.	Media	Critico	Tener otro Mac donde poder realizar pruebas. Disponer de otro iPhone.
Pérdida de Datos	Pérdida de cualquier tipo de datos del proyecto	Media	Critico	Múltiples copias de seguridad, tanto en red (NAS) como con TimeMachine, así como en iCloud.
Ausencia de Conocimientos de Objective-C / Swift	No es solo una tecnología nueva para mi, sino que mi experiencia en programación orientada a Objetos es muy limitada	Alta	Alto	Buscar libros y cursos para aumentar los conocimientos en este área.
Ausencia de Conocimientos de iOS	Como en el caso anterior, nunca he programado para iOS, por lo que no sé a qué me voy a enfrentar. Mi experiencia programando es muy escasa.	Alta	Alto	Buscar libros y cursos para aumentar los conocimientos en este área.

Riesgo	Descripción	Probabilidad de Aparición	Impacto	Acciones
Cuotas de inscripción en el desarrollo de iOS	Actualmente hay APIs como CloudKit que ayudan al desarrollador, pero requieren de una suscripción Anual para poder usarlas.	Media	Alto	La cuota de desarrollador de iOS es de 99€ Anuales que tendré que pagar para poder usar CloudKit. A mayores, debemos sumar la inscripción, que considero necesaria en mi caso, de un curso de desarrollo de IOS de 358€
Fallos en la planificación del proyecto	Durante el desarrollo del proyecto, pueden aparecer errores en la cuantificación del esfuerzo de una tarea y penalizar el resto del proyecto.	Media	Medio	Seguimiento estricto del itinerario del proyecto. Intentar una programación modular que evite que un error afecte a toda la aplicación
Hardware no adecuado o suficientemente potente.	Es posible que durante la realización del proyecto, los requisitos de funcionamiento penalicen una ejecución fluida.	Baja	Bajo	Sistema Operativo Actualizado, Xcode Actualizado, Ampliación de memoria RAM a 8GB.
Enfermedad	Debido al clima de Galicia, las probabilidades de resfriado o gripe son altas.	Media	Media	Seguir las pautas recomendadas por sanidad para evitar enfermarse.

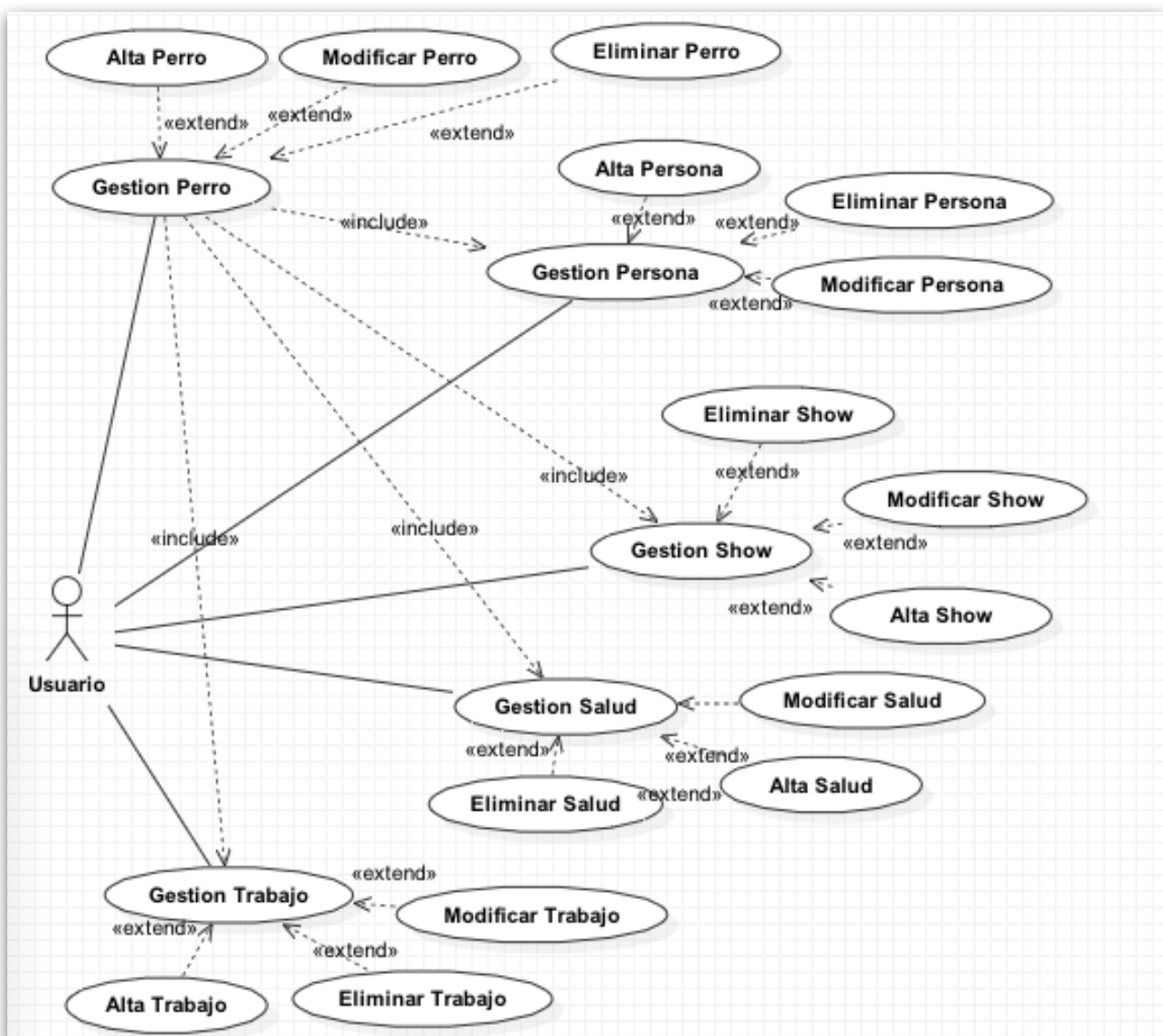
Riesgo	Descripción	Probabilidad de Aparición	Impacto	Acciones
Coordinación entre Familia y Trabajo	Solo dispongo de los fines de semana y alguna noche para poder realizar el proyecto, por lo que se solapa con el tiempo que tengo para la familia	Inevitable	Alto	Si sumamos el proyecto y los cursos, casi la totalidad del tiempo libre estará dedicado a la consecución de este proyecto, por lo que lo único que puedo hacer es prometer unas buenas vacaciones.

5. DISEÑO CONCEPTUAL

En este apartado, veremos los casos de uso de la aplicación así como los escenarios de uso. Al utilizar CloudKit, tomamos los datos de iCloud del usuario para almacenar sus datos, por lo que llamamos "Usuario Registrado" a dicho usuario, aunque realmente no haga falta un registro.

5.1. Casos de Uso

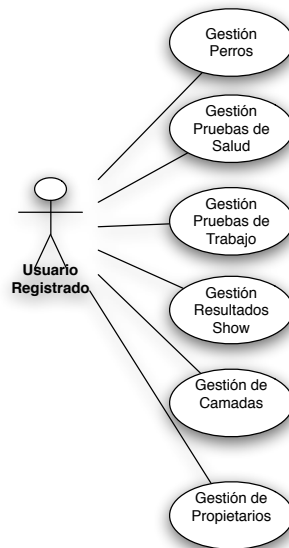
Aquí podemos ver una vista general de los casos de uso de la aplicación. En la siguiente



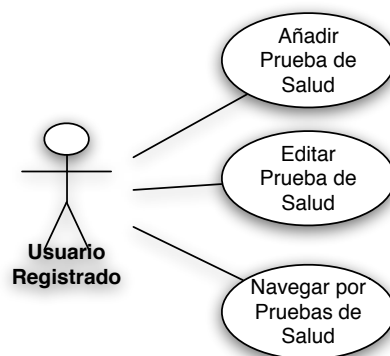
5.2.Escenarios de Uso

Mediante las capturas de este apartado, podremos ver las acciones concretas que podrá hacer el usuario con nuestra aplicación.

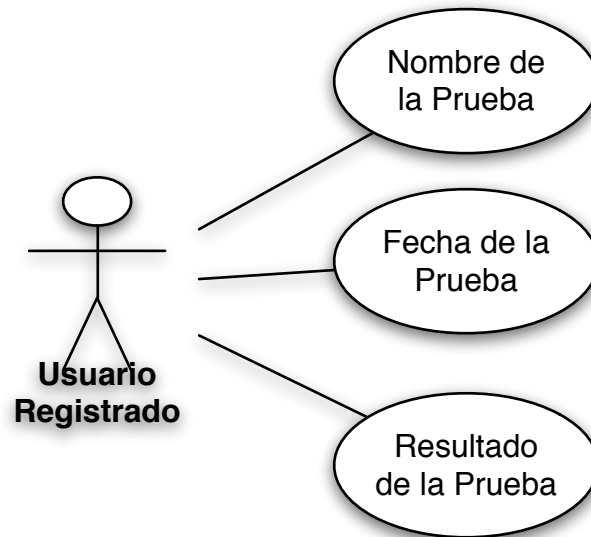
5.2.1.Pantalla Principal



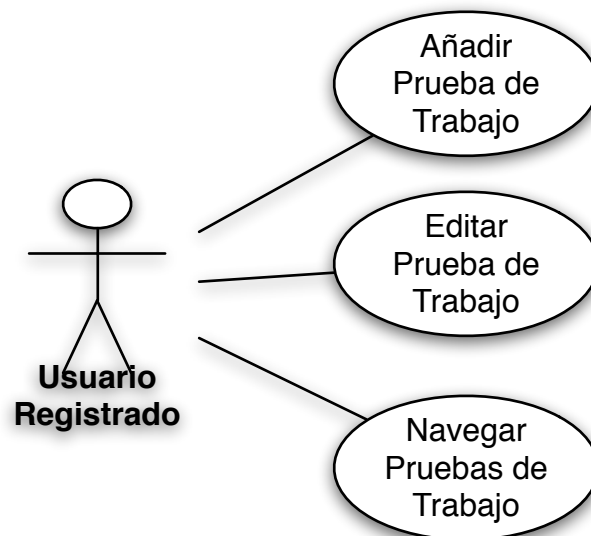
5.2.2.Gestión Pruebas de Salud



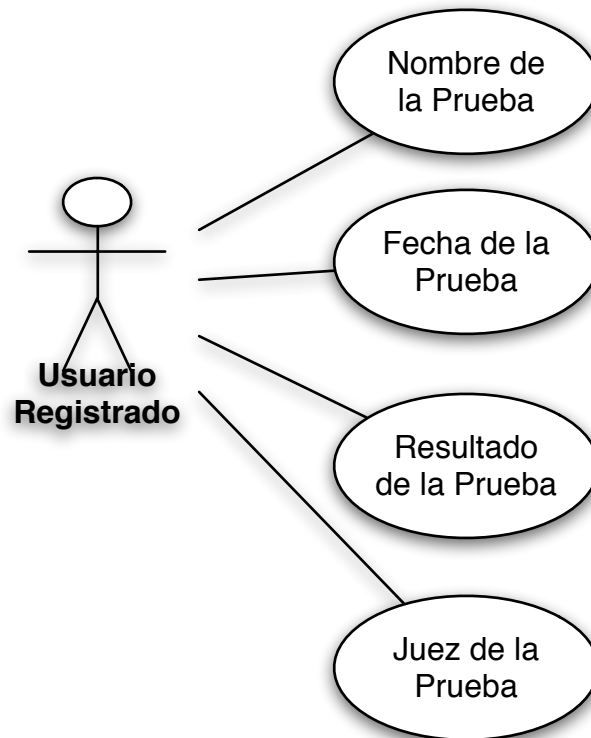
5.2.3. Gestión Prueba de Salud - Añadir/Modificar



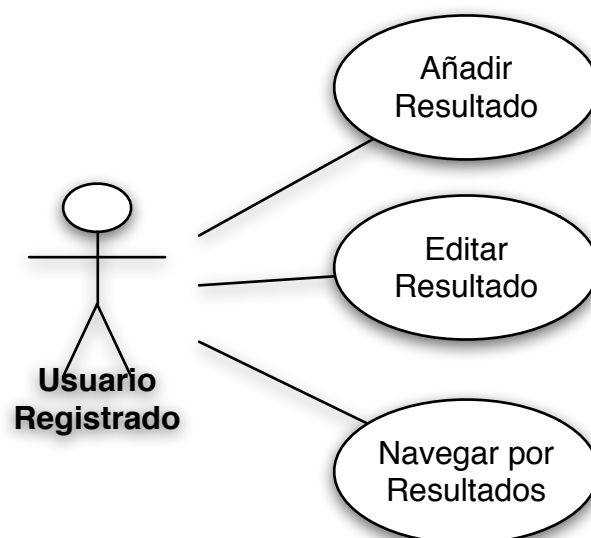
5.2.4. Gestión Pruebas de Trabajo



5.2.5.Gestión Prueba de Trabajo - Añadir/Modificar



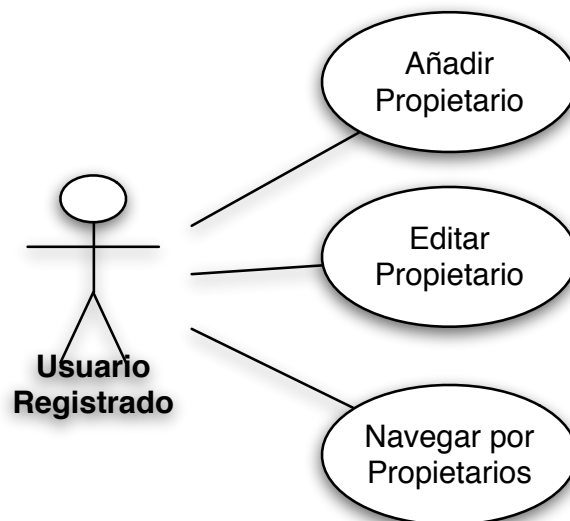
5.2.6.Gestión Resultados de Show



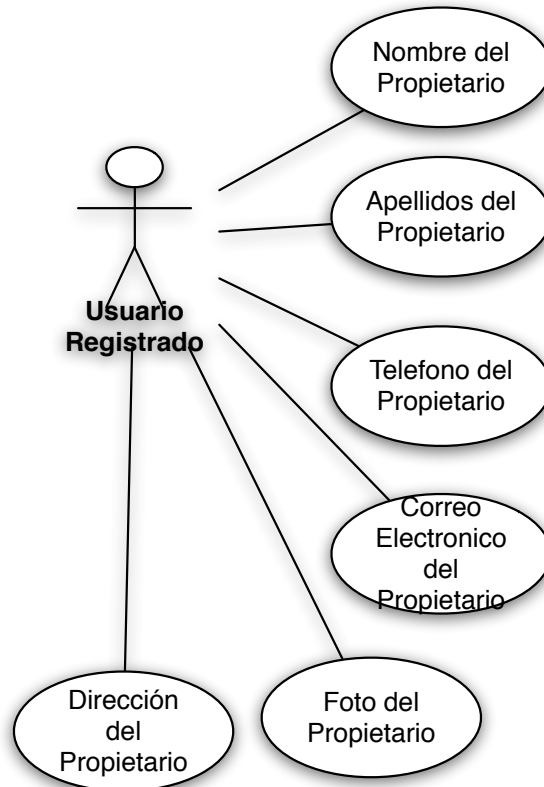
5.2.7. Gestión Resultado de Show - Añadir/Modificar



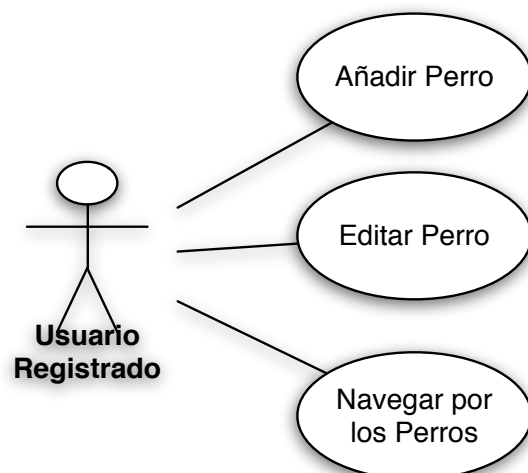
5.2.8. Gestión Propietarios



5.2.9.Gestión Propietarios - Añadir/Modificar



5.2.10.Gestión Perros



5.2.11.Gestión Perros - Añadir/Modificar



5.3. Descripción de los Casos de Uso

Veremos ahora las diferentes descripciones de los casos de uso.

5.3.1. Pantalla Principal (PAP)

Código	Nombre del Caso	Actor
PAP1	Gestionar Propietarios	Usuario Logueado
PAP2	Gestionar Perros	Usuario Logueado
PAP3	Gestionar Pruebas de Salud	Usuario Logueado
PAP4	Gestionar Pruebas de Trabajo	Usuario Logueado
PAP5	Gestionar Resultados de Show	Usuario Logueado

PAP1	Gestionar Propietarios
RESUMEN	Usuario loqueado en la aplicación pulsa sobre Gestión de Propietarios
ACTOR	Usuario loqueado
PRECONDICIONES	El Usuario pulsa sobre el icono "Gestion de Propietarios"
POSTCONDICIONES	Se muestra la pantalla de gestión de Propietarios, con los propietarios existentes mostrados en forma de lista.
FLUJO DE ACCIÓN	En el caso de que ya existan propietarios registrados, estos se listan para su navegación. Si no hay ninguno creado, no se listan. En la zona inferior de la pantalla, se muestran los botones Añadir y Editar.
FLUJO ALTERNATIVO	En el caso de que no exista ningún Propietario, aparece una alerta emergente informando de esto, y da la opción de crear un propietario

PAP2	Gestionar Perros
RESUMEN	Usuario loqueado en la aplicación pulsa sobre Gestión de Perros
ACTOR	Usuario loqueado
PRECONDICIONES	El Usuario pulsa sobre el icono "Gestión de Perros"
POSTCONDICIONES	Se muestra la pantalla de gestión de perros, con los perros existentes mostrados en forma de lista.
FLUJO DE ACCIÓN	En el caso de que ya existan perros registrados, estos se listan para su navegación. Si no hay ninguno creado, no se listan. En la zona inferior de la pantalla, se muestran los botones Añadir y Editar.
FLUJO ALTERNATIVO	En el caso de que no exista ningún perro, aparece una alerta emergente informando de esto, y da la opción de crear un perro
PAP3	Gestionar Prueba de Salud
RESUMEN	Usuario loqueado en la aplicación pulsa sobre Gestión de Prueba de Salud
ACTOR	Usuario loqueado
PRECONDICIONES	El Usuario pulsa sobre el icono "Gestión de Prueba de Salud"
POSTCONDICIONES	Se muestra la pantalla de gestión de pruebas de salud, con las pruebas existentes en forma de lista.
FLUJO DE ACCIÓN	En el caso de que ya existan pruebas registradas, estos se listan para su navegación Si no hay ninguna creada, no se listan. En la zona inferior de la pantalla, se muestran los botones Añadir y Editar.
FLUJO ALTERNATIVO	En el caso de que no exista ninguna prueba, aparece una alerta emergente informando de esto, y da la opción de crear una prueba.

PAP4	Gestionar Prueba de Trabajo
RESUMEN	Usuario loqueado en la aplicación pulsa sobre Gestión de Prueba de Trabajo
ACTOR	Usuario loqueado
PRECONDICIONES	El Usuario pulsa sobre el icono “Gestión de Prueba de Trabajo”
POSTCONDICIONES	Se muestra la pantalla de gestión de pruebas de trabajo, con las pruebas existentes en forma de lista.
FLUJO DE ACCIÓN	En el caso de que ya existan pruebas registradas, estos se listan para su navegación Si no hay ninguna creado, no se listan. En la zona inferior de la pantalla, se muestran los botones Añadir y Editar.
FLUJO ALTERNATIVO	En el caso de que no exista ninguna prueba, aparece una alerta emergente informando de esto, y da la opción de crear una prueba.
PAP5	Gestionar Resultado de Show
RESUMEN	Usuario loqueado en la aplicación pulsa sobre Gestión de Resultado de Show
ACTOR	Usuario loqueado
PRECONDICIONES	El Usuario pulsa sobre el icono “Gestión de Resultados de Show”
POSTCONDICIONES	Se muestra la pantalla de gestión de resultados de Show, con los resultados existentes en forma de lista.
FLUJO DE ACCIÓN	En el caso de que ya existan resultados registrados, estos se listan para su navegación Si no hay ninguna creado, no se listan. En la zona inferior de la pantalla, se muestran los botones Añadir y Editar.
FLUJO ALTERNATIVO	En el caso de que no exista ningún resultado, aparece una alerta emergente informando de esto, y da la opción de crear un resultado.

5.3.2.Gestionar Propietarios (GPR)

GPR	Gestionar Propietarios - Añadir-Editar
RESUMEN	Usuario logueado, dentro de Gestión de Propietarios pulsa Añadir
ACTOR	Usuario logueado
PRECONDICIONES	El usuario está dentro de la sección Gestión de Propietarios
POSTCONDICIONES	Se muestra la pantalla para añadir Propietario
FLUJO DE ACCIÓN	Se pulsa sobre el botón añadir o en su defecto, si no hay propietario creado, se pulsa en la alerta emergente sobre el botón añadir Se rellenan los datos y se guarda el propietario.
FLUJO ALTERNATIVO	En caso de querer editar el propietario, se selecciona el propietario a editar y se pulsa sobre el botón Editar

5.3.3.Gestionar Perros (GPE)

GPR	Gestionar Propietarios - Añadir-Editar
RESUMEN	Usuario logueado, dentro de Gestión de Perros pulsa Añadir
ACTOR	Usuario logueado
PRECONDICIONES	El usuario está dentro de la sección Gestión de Perros
POSTCONDICIONES	Se muestra la pantalla para añadir Perro
FLUJO DE ACCIÓN	Se pulsa sobre el botón añadir o en su defecto, si no hay perro creado, se pulsa en la alerta emergente sobre el botón añadir En este caso, al ser bastante información, se muestra en varias pantallas, con una flecha de "Siguiete" en cada tramo. Al final, aparece el botón para guardar.
FLUJO ALTERNATIVO	En caso de querer editar el perro, se selecciona el perro a editar y se pulsa sobre el botón Editar

5.3.4.Gestionar Pruebas de Salud (GPS)

GPS	Gestionar Pruebas de Salud - Añadir-Editar
RESUMEN	Usuario logueado, dentro de Gestión de Pruebas de Salud pulsa Añadir
ACTOR	Usuario logueado
PRECONDICIONES	El usuario está dentro de la sección Gestión de Pruebas de Salud
POSTCONDICIONES	Se muestra la pantalla para añadir Prueba de Salud
FLUJO DE ACCIÓN	Se pulsa sobre el botón añadir o en su defecto, si no hay prueba creada, se pulsa en la alerta emergente sobre el botón añadir Se rellena la información correspondiente y se guarda.
FLUJO ALTERNATIVO	En caso de querer editar la prueba, se selecciona y se pulsa editar.

5.3.5.Gestionar Pruebas de Trabajo (GPT)

GPT	Gestionar Pruebas de Trabajo - Añadir-Editar
RESUMEN	Usuario logueado, dentro de Gestión de Pruebas de Trabajo pulsa Añadir
ACTOR	Usuario logueado
PRECONDICIONES	El usuario está dentro de la sección Gestión de Pruebas de Trabajo
POSTCONDICIONES	Se muestra la pantalla para añadir Prueba de Trabajo
FLUJO DE ACCIÓN	Se pulsa sobre el botón añadir o en su defecto, si no hay prueba creada, se pulsa en la alerta emergente sobre el botón añadir Se rellena la información correspondiente y se guarda.
FLUJO ALTERNATIVO	En caso de querer editar la prueba, se selecciona y se pulsa editar.

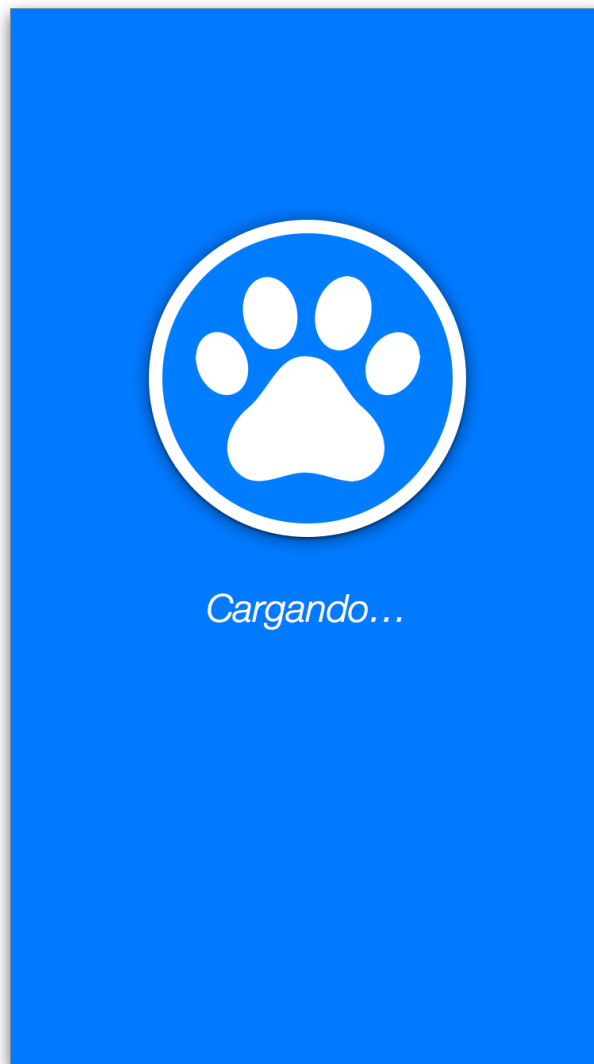
5.3.6.Gestionar Resultados de Show (GRS)

GRS	Gestionar Resultados de Show - Añadir-Editar
RESUMEN	Usuario logueado, dentro de Gestión de Resultados de Show y Pulsa Editar.
ACTOR	Usuario logueado
PRECONDICIONES	El usuario está dentro de la sección Gestión Resultados de Show
POSTCONDICIONES	Se muestra la pantalla para añadir Resultados de Show
FLUJO DE ACCIÓN	Se pulsa sobre el botón añadir o en su defecto, si no hay resultados, se muestra una alerta emergente que avisa, y da la opción de crear una. Si ya hay resultados, se pulsa el botón añadir. Se rellenan los datos necesarios y se guarda.
FLUJO ALTERNATIVO	En caso de querer editar el resultado, se selecciona y se pulsa editar.

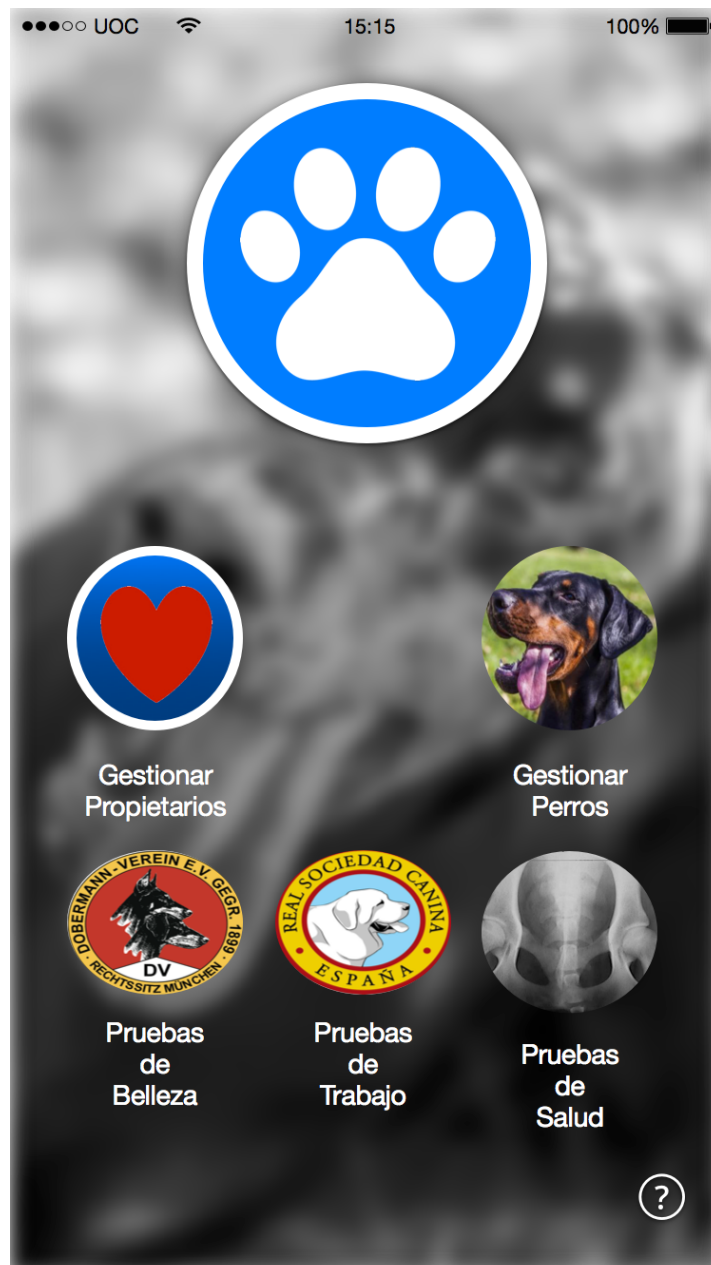
6.PROTOTIPADO

En este apartado, veremos unas capturas de lo que será la aplicación final, con toda la funcionalidad. Al ser prototipos, la versión final puede verse de manera diferente por aspectos técnicos no previstos.

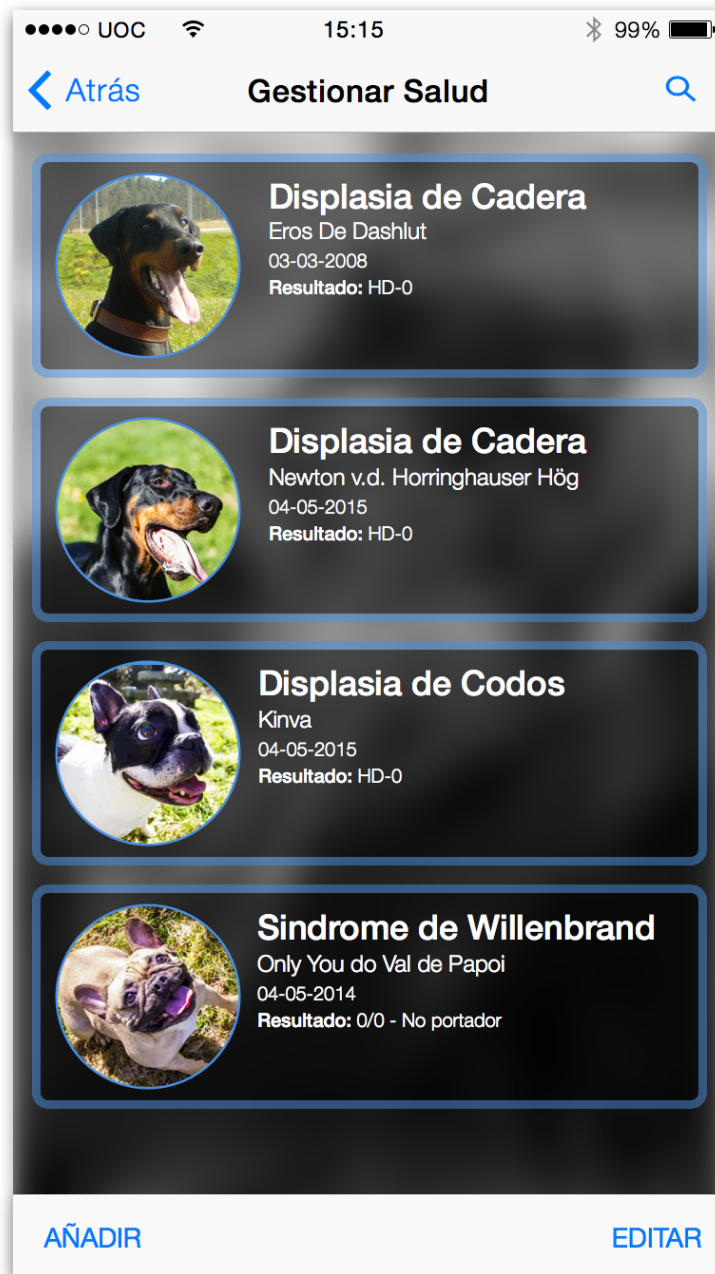
6.1. Pantalla de Carga



6.2. Pantalla Principal



6.3. Pantalla Gestión de Pruebas de Salud



6.4. Pantalla Gestión de Pruebas de Trabajo



6.5. Pantalla Gestión de Resultados de Show



6.6. Pantalla Detalle Perro



6.7. Pantalla Gestión de Perros



The screenshot shows a mobile application interface for managing dogs. At the top, the status bar displays 'UOC', signal strength, Wi-Fi, time '15:15', and battery '99%'. The app header includes a back arrow labeled 'Atrás', the title 'Gestionar Perros', and a search icon. The main content area lists four dogs, each with a circular profile picture and text details:

- Eros De Dashlut**: Armani Dente del Gigante x Nina de Grifeu, 28-01-2006 / 21-08-2012, Propietario: Juan Antonio Sánchez Carrillo, Teléfono: 666666666
- Newton v.d Horrigh...**: Yacheero's Pando Pandro x Alessandra Avasir, 02-02-2014, Propietario: Juan Antonio Sánchez Carrillo, Teléfono: 666666666
- Kinva**: Desconocido x Desconocido, 08-02-2006, Propietario: Cristina Nuñez Martinez, Teléfono: 666666666
- Only You Do Val de Papoi**: Aquiles de la Bouillabaise x Leona Lewis Do Val ..., 09-02-2013, Propietario: Cristina Nuñez Martinez, Teléfono: 666666666

At the bottom of the screen, there are two buttons: 'AÑADIR' on the left and 'EDITAR' on the right.

6.8. Icono de la Aplicación



7. DISEÑO TÉCNICO

En este apartado, veremos como evoluciona nuestra aplicación de manera técnica. Nos centraremos en el modelo de datos y en el diseño en Xcode en base a los prototipos creados con Sketch.

En nuestra aplicación, todo se realiza desde el dispositivo móvil de manera local, por lo que no necesitaremos configurar nada, simplemente instalar y a funcionar.

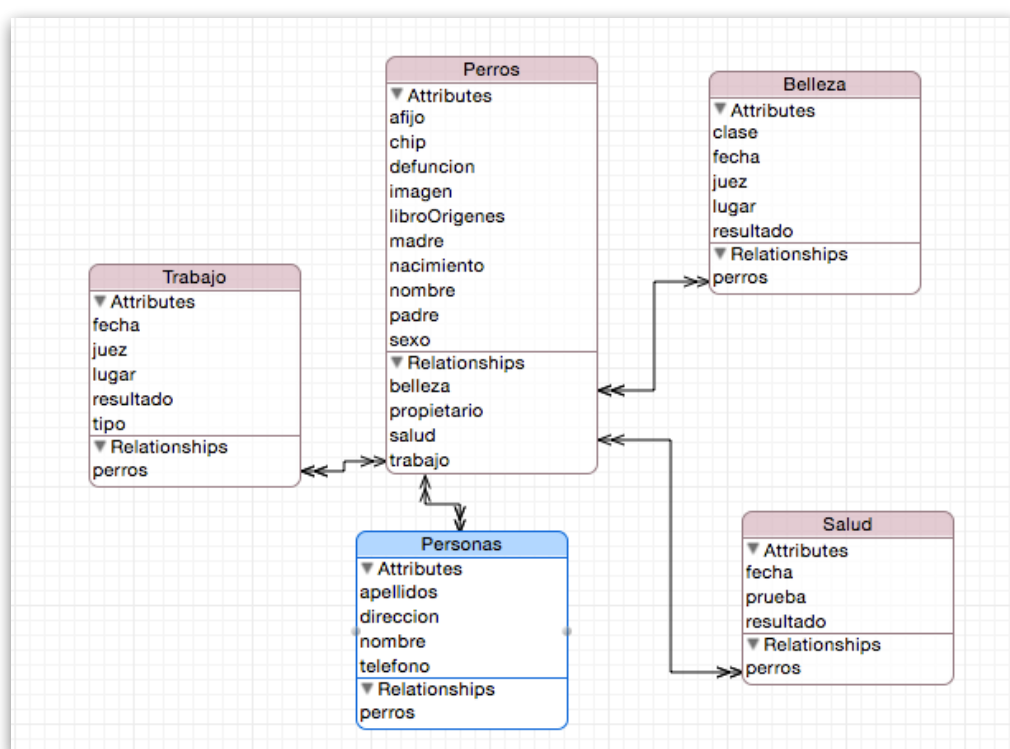
7.1. Modelo de Base de Datos

Una vez que tenemos claro el concepto de la aplicación que queremos crear, es necesario crear una base de datos para el manejo de la información dentro de la aplicación.

Para esta primera versión de la aplicación, se va a utilizar CoreData para el modelado de la base de datos ya que disponemos dentro del propio IDE de programación de las herramientas necesarias para acceder a la información.

Para agilizar el proceso de programación posterior al diseño de la base de datos, se han tomado todos los datos como String.

Seguidamente, mostramos como ha quedado nuestra base de datos:



Como se ve en la imagen, tenemos las siguientes entidades: "Perros", "Personas", "Salud", "Trabajo" y "Belleza".

La Entidad "Perros", es la que lleva el peso de la información, ya que desde ella se puede acceder a todas las demás entidades, editar su información y eliminarla.

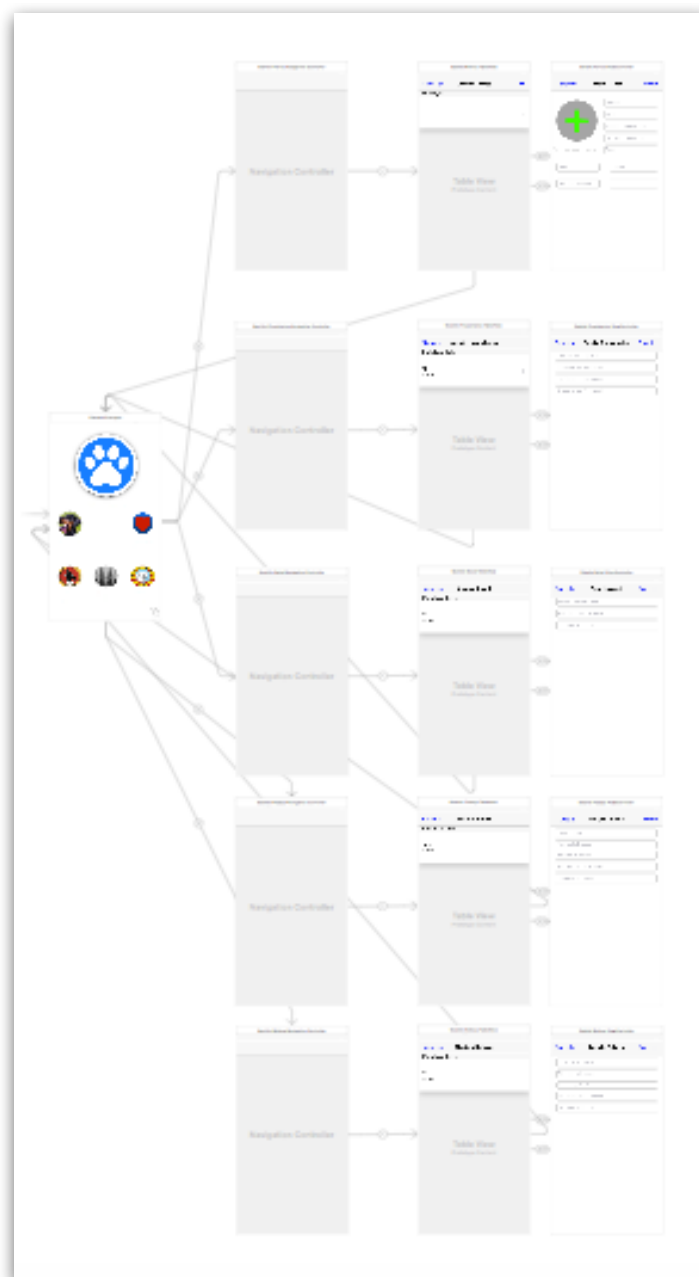
He decidido que todas las relaciones sean del tipo Many-To-Many, ya que considero que un criador puede sacarle la misma prueba el mismo día a varios perros, por lo que descarto la relación de tipo One-To-Many.

7.2. StoryBoard

Algo que me ha sorprendido muy gratamente desde el inicio de este proyecto ha sido la manera de programar en xCode; esta se centra en “lo que ve el usuario” y posteriormente conecta dicha interfaz al código.

Si bien en un principio parece fácil de usar, conseguir una aplicación universal utilizando AutoLayouts y Size Classes puede llegar a ser un verdadero quebradero de cabeza. Es por esto último que nuestra aplicación se ha desarrollado para iPhone 6.

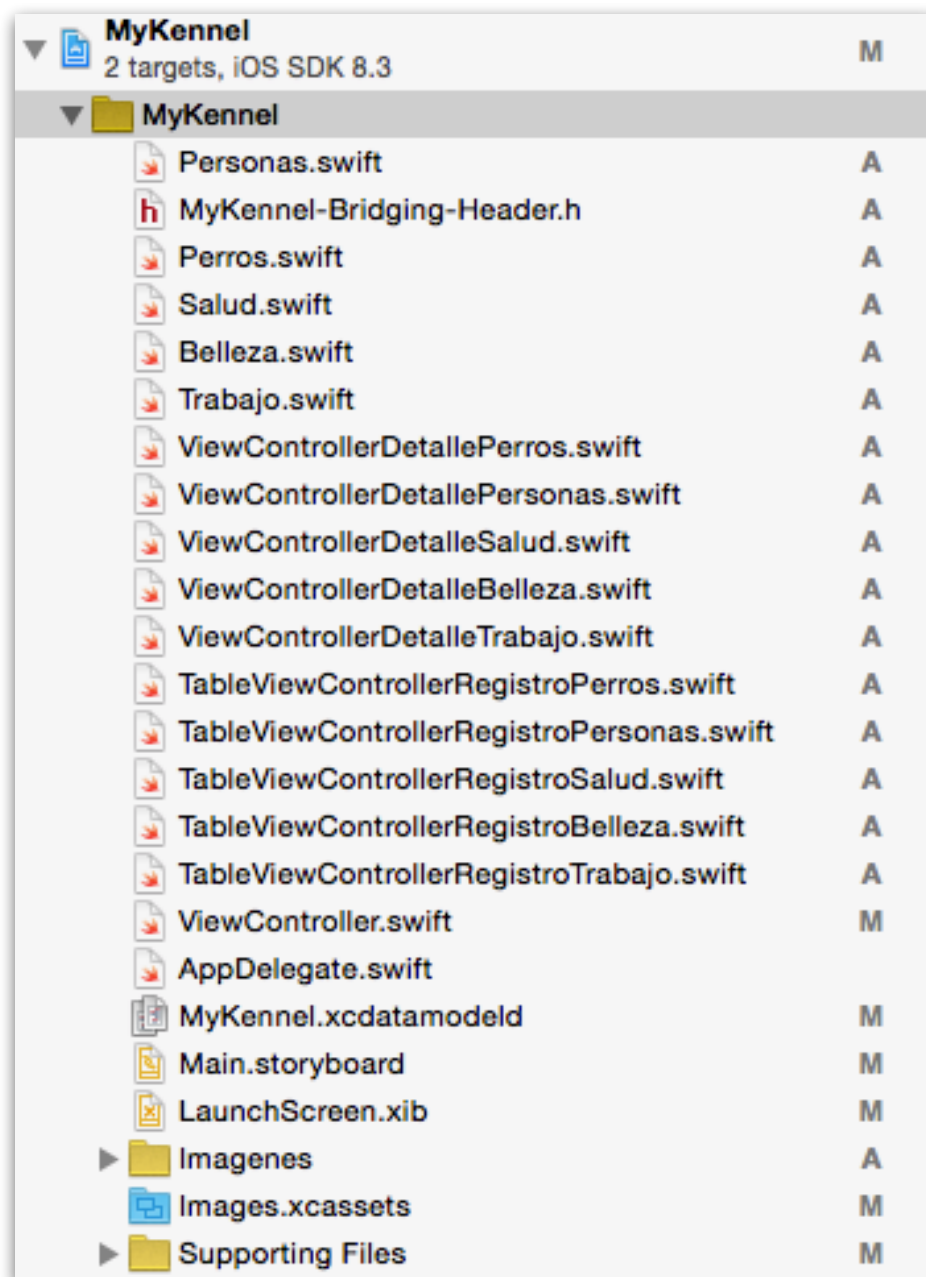
Aquí se puede ver el diseño final de nuestra aplicación:



8.IMPLEMENTACIÓN

En este apartado vamos a ver como está implementada la aplicación, sus clases y métodos propios.

A continuación, se muestra la estructura de la aplicación:



Las clases "Personas.swift", "Perros.swift", "Belleza.swift", "Trabajo.swift" y "Salud.swift", han sido generadas mediante el modelo de datos y simplemente tienen los atributos de nuestras entidades, así como las relaciones entre ellas.

Aquí se muestra una imagen de la clase "Perros.swift"

```
import Foundation
import CoreData
//Hago la clase visible para otras clases
@objc (Perros)

class Perros: NSManagedObject {

    @NSManaged var afijo: String
    @NSManaged var chip: String
    @NSManaged var defuncion: String
    @NSManaged var imagen: NSData
    @NSManaged var libroOrigenes: String
    @NSManaged var madre: String
    @NSManaged var nacimiento: String
    @NSManaged var nombre: String
    @NSManaged var padre: String
    @NSManaged var sexo: String
    @NSManaged var belleza: NSSet
    @NSManaged var propietario: NSSet
    @NSManaged var salud: NSSet
    @NSManaged var trabajo: NSSet

}
```

Las clases `ViewControllerDetalleXXX` y `TableViewCellRegistroXXX` son las encargadas de gestionar la información proporcionada por el usuario final, pudiendo guardar, modificar y eliminar registros.

Todo el código se ha comentado para que se entienda lo que se hace en cada momento, sin embargo, cabe destacar los métodos más utilizados:

Dentro de la función `viewDidLoad()`, ponemos la imagen de fondo de nuestra aplicación y comprobamos si existe el registro seleccionado y si existe, lo sacamos por pantalla. (Ejemplo para Perros)

```
override func viewDidLoad() {
    super.viewDidLoad()

    //Pongo la imagen de Fondo
    UIGraphicsBeginImageContext(self.view.frame.size)
    UIImage(named: "Background.png")?.drawInRect(self.view.bounds)
    var image: UIImage = UIGraphicsGetImageFromCurrentImageContext()
    UIGraphicsEndImageContext()
    self.view.backgroundColor = UIColor(patternImage: image)

    // Si existe el perro, muestro los datos por pantalla
    if(existePerro != nil)
    {
        nombrePerroEditable.text = nombrePerro
        afijoPerroEditable.text = afijoPerro
        padrePerroEditable.text = padrePerro
        madrePerroEditable.text = madrePerro
        fechaNacimientoEditable.text = nacimiento
        fechaDefuncionEditable.text = defuncion
        sexoEditable.text = sexo
        loeEditable.text = loe
        chipEditable.text = chip
        imagenPerroEditable.image = UIImage(named:"imagen")
    }
}
```


En la siguiente captura, se pueden ver los métodos para guardar la información proporcionada por el usuario y para cancelar. (Ejemplo para Perros)

```
@IBAction func guardarDetallePerros(sender: AnyObject) {

    //Creo instancia de la clase appDelegate para gestionar la pila de CoreData
    let miDelegate : AppDelegate = UIApplication.sharedApplication().delegate as! AppDelegate
    //Creo instancia al objeto Managedeg Objetc Context
    let managedObjectContexto : NSManagedObjectContext = miDelegate.managedObjectContext!
    //Creo instancia a la entidad Perro del Core Data
    let entityDescription = NSEntityDescription.entityForName("Perros", inManagedObjectContext: managedObjectContexto)
    if (existePerro != nil){

        existePerro.setValue(nombrePerroEditable.text as String, forKey: "nombre")
        existePerro.setValue(afijoPerroEditable.text as String, forKey: "afijo")
        existePerro.setValue(padrePerroEditable.text as String, forKey: "padre")
        existePerro.setValue(madrePerroEditable.text as String, forKey: "madre")
        existePerro.setValue(fechaNacimientoEditable.text as String, forKey: "nacimiento")
        existePerro.setValue(fechaDefuncionEditable.text as String, forKey: "defuncion")
        existePerro.setValue(chipEditable.text as String, forKey: "chip")
        existePerro.setValue(sexoEditable.text as String, forKey: "sexo")
        existePerro.setValue(loeEditable.text as String, forKey: "libroOrigenes")
        existePerro.setValue(NSData (data: UIImagePNGRepresentation(imagenPerroEditable.image)), forKey: "imagen")

    }else{
        //Mapeamos nuestras propiedades con los valores del TextField
        //Creamos una instancia de la clase Perros y le asignamos una entidad y un contexto de objeto gestionado
        var newPerros : Perros = NSEntityDescription.insertNewObjectForEntityForName("Perros", inManagedObjectContext:
            managedObjectContexto) as! Perros
        newPerros.nombre = nombrePerroEditable.text
        newPerros.afijo = afijoPerroEditable.text
        newPerros.padre = padrePerroEditable.text
        newPerros.madre = madrePerroEditable.text
        newPerros.nacimiento = fechaNacimientoEditable.text
        newPerros.defuncion = fechaDefuncionEditable.text
        newPerros.imagen = NSData (data: UIImagePNGRepresentation(imagenPerroEditable.image))
        println(newPerros)
    }
    //Guardamos los valores
    managedObjectContexto.save(nil)
    let alertView = UIAlertView (title: "Guardado Correctamente", message: "Se han guardado correctamente los datos!!!",
        delegate: nil, cancelButtonTitle: "OK")
    alertView.show()

    navigationController?.popToRootViewControllerAnimated(true)
}

@IBAction func cancelarDetallePerros(sender: AnyObject) {
    navigationController?.popToRootViewControllerAnimated(true)
}
}
```

Funciones para mostrar la información en los TextFields: (Ejemplo Perros)

```
//Método que hace que se presenten en los textField los datos al pasar a la vista de detalle
override func prepareForSegue(segue: UIStoryboardSegue, sender: AnyObject!) {
    //Condiciona para ver si el identificador es el mismo
    if segue.identifier == "actualizaPerros" {
        //Variable para almacenar el índice de la tabla
        var indexPath : NSIndexPath = self.tableView.indexPathForSelectedRow()!
        //Creo Objeto para la clase NSManagedObject y le paso el array y la fila seleccionada
        var selectedItem : NSManagedObject = datosConsultaPerros[indexPath.row] as! NSManagedObject
        //Presento la vista de Detalle
        let vistaDetallePerros : ViewControllerDetallePerros = segue.destinationViewController as! ViewControllerDetallePerros
        //Paso el valor seleccionado en la propiedad SelectItem
        vistaDetallePerros.nombrePerro = selectedItem.valueForKey("nombre") as? String
        vistaDetallePerros.afijoPerro = selectedItem.valueForKey("afijo") as? String
        vistaDetallePerros.padrePerro = selectedItem.valueForKey("padre") as? String
        vistaDetallePerros.madrePerro = selectedItem.valueForKey("madre") as? String
        vistaDetallePerros.chip = selectedItem.valueForKey("chip") as? String
        vistaDetallePerros.loe = selectedItem.valueForKey("libroOrigenes") as? String
        vistaDetallePerros.sexo = selectedItem.valueForKey("sexo") as? String
        vistaDetallePerros.nacimiento = selectedItem.valueForKey("nacimiento") as? String
        vistaDetallePerros.defuncion = selectedItem.valueForKey("defuncion") as? String
        vistaDetallePerros.imagen = selectedItem.valueForKey("imagen") as? UIImage

        vistaDetallePerros.existePerro = selectedItem
    }
}
}
```

Función que gestiona la información de la Celda:

```

override func tableView(tableView: UITableView, cellForRowAtIndexPath indexPath: NSIndexPath) -> UITableViewCell {
    //Identificador de Celda
    let CellID: NSString = "Cell"

    //Paso Identificador de celda
    var cell: UITableViewCell = tableView.dequeueReusableCellWithIdentifier(CellID as String) as! UITableViewCell

    //Creo objeto NSManagedObject
    var datos: NSManagedObject = datosConsultaPerros[indexPath.row] as! NSManagedObject

    //String que almacena Nombre del Perro
    var nombre = datos.valueForKeyPath("nombre") as! String

    //String que almacena el Afijo del Perro
    var afijo = datos.valueForKeyPath("afijo") as! String

    //Guardo nombre+afijo
    var nombreAfijo = nombre.stringByAppendingString(" \ \(afijo)")

    //Imagen del perro
    var imagen : UIImage = UIImage(data: datos.valueForKeyPath("imagen") as! NSData)!

    //Creo Constante para establecer un tipo de fuente y su tamaño
    let font: UIFont = UIFont(name: "AvenirNext-Regular",size:16.0)!

    //Creo Constante para establecer altura de celda
    let alturaCeldaX: CGFloat = alturaCelda - 15

    //Creo Constante a la clase Objective-C (Pod)
    let placeholder = BMInitialsPlaceholderView(diameter: alturaCeldaX)

    //Modificamos las propiedades del placeholder
    placeholder.circleColor = circleColorForIndexPath (indexPath)

    placeholder.initials = initialsStringForPerrosString (nombreAfijo) as String

    //Le Paso al circulo el tipo de fuente
    placeholder.font = font

    //Paso el Placeholder como accessoryView de mi celda
    cell.accessoryView = placeholder

    //Paso la fuente y el tamaño a las celdas
    cell.textLabel?.font = font

    //Presento la información Nombre - Afijo en Pantalla con la imagen del perro
    cell.textLabel?.text = "\ \(nombre) - \ \(afijo)"

```

```

//Presento la información Nombre - Afijo en Pantalla con la imagen del perro
cell.textLabel?.text = "\ \(nombre) - \ \(afijo)"

cell.detailTextLabel?.text = datos.valueForKeyPath("sexo") as? String

cell.imageView?.image = imagen

return cell
}

```

8.1. Aplicación Desarrollada

Los ficheros que contienen el código fuente de la aplicación desarrollada se adjuntan a este documento, junto con las imágenes empleadas, prototipos, etc.

9. CONCLUSION

Una vez finalizado este proyecto, teniendo en cuenta que he partido de 0 en programación para iOS, he de decir que si bien estoy satisfecho con el resultado final, no he podido implementar ciertos aspectos que considero necesarios (uso de las relaciones entre entidades a nivel usuario) y otros que considero secundarios (Interfaz gráfica) debido a que aún no manejo todo lo que me gustaría el xCode.

Asimismo, me he encontrado con varias actualizaciones, tanto de Xcode como de iOS, lo que ha dificultado las pruebas en el iPhone y la aparición de errores debido a la actualización del lenguaje de programación.

Por último, la realización de este proyecto me ha supuesto una experiencia enriquecedora que me permitirá desempeñar mejor mi trabajo en los roles de gestión y desarrollo que asuma mi futura actividad profesional.

10.BIBLIOGRAFÍA

- Iconos varios para el prototipo:
 - <https://ui8.net/search?term=icon>

- Plantillas para Sketch:
 - <http://www.sketchappsources.com/all-free-sources.html>

- Flinto:
 - <https://www.flinto.com>

- Sketch:
 - <http://bohemiancoding.com/sketch/>

- Formación Complementaria:
 - <https://www.udemy.com/courses/>
 - <https://designcode.io>

- iOS Apprentice (Third Edition) - Beginning with iOS Development with Swift
 - http://www.amazon.es/iOS-Apprentice-Third-Beginning-Development/dp/1942878028/ref=sr_1_1?ie=UTF8&qid=1434916984&sr=8-1&keywords=iOS+apprentice