|  |  |
| --- | --- |
| ClinicVet  Tutor: David Escrich | Descripción breve  ClinicVet se basa en una APP motivada para facilitar las cosas a los usuarios de una Clínica Veterinaria.  Francisco Nicolau Ausejo  2º GS Desarrollo de Aplicaciones Multiplataforma |

INDICE

[1. INTRODUCCIÓN Y JUSTIFICACIÓN 2](#_Toc74041365)

[2. OBJETIVOS Y METODOLOGÍA. 3](#_Toc74041366)

[Objetivos principales. 3](#_Toc74041367)

[3. ANALISIS DEL CONTEXTO. 4](#_Toc74041368)

[Dispositivo: Android 4](#_Toc74041369)

[Lenguaje de programación: Kotlin 4](#_Toc74041370)

[Base de datos: SQLite 4](#_Toc74041371)

[Consultas 5](#_Toc74041372)

[Diagramas: Draw.io y Visio 5](#_Toc74041373)

[Metodología 6](#_Toc74041374)

[4. DESARROLLO DEL CONTENIDO 7](#_Toc74041375)

[Análisis de la Base de Datos 7](#_Toc74041376)

[Funcionalidad de la aplicación 9](#_Toc74041377)

[Diagrama de flujo. 9](#_Toc74041378)

[Diagrama de Secuencia 10](#_Toc74041379)

[Funcionamiento APP (Registro, acceso, registro mascota, acceso historial) 11](#_Toc74041380)

[Funcionamiento APP (Admin registro publicación/Producto). 12](#_Toc74041381)

[Funcionamiento APP (Pedir cita). 13](#_Toc74041382)

[Funcionamiento APP (Tienda). 13](#_Toc74041383)

[Clases principales de la aplicación. 14](#_Toc74041384)

[Adapters en un RecyclerView 14](#_Toc74041385)

[Creación de la BD. 14](#_Toc74041386)

[Problemas encontrados 15](#_Toc74041387)

[Ampliación 16](#_Toc74041388)

[5. CONCLUSIÓN Y VALORACION PERSONAL 20](#_Toc74041389)

[Conclusiones del proyecto 20](#_Toc74041390)

[Valoración personal 21](#_Toc74041391)

[6. BIBLIOGRAFÍA 22](#_Toc74041392)

# INTRODUCCIÓN Y JUSTIFICACIÓN

ClinicVet se basa en un trabajo de investigación, estudio y desarrollo de un prototipo de app creada para gestionar las tareas llevadas a cabo en una Clínica Veterinaria por un usuario.

Se basa en una aplicación creada para que el usuario pueda crear sus citas, verlas, y borrarlas.

Así como llevar un registro de sus propios animales, su historial de visitas, sus citas, sus características, etc.

También he llevado a cabo una sección de Curiosidades donde un usuario Admin, en este caso yo como administrador, poseo la libertad de crear curiosidades sobre los animales.

Así mismo, también he incluido una sección Tienda donde el usuario Admin también puede crear productos para la tienda.

Así mismo, el usuario, puede visualizar estas curiosidades y la tienda de la app.

# OBJETIVOS Y METODOLOGÍA.

## Objetivos principales.

El objetivo principal es crear una facilidad al cliente de cara a una clínica veterinaria.

El sistema incorpora las siguientes funciones principales:

* Creación de usuario. El usuario puede realizar el registro en una base de datos de su usuario, de tal manera que el usuario, solo pueda gestionar sus animales.
* Borrado de usuario. El usuario puede realizar un borrado de su usuario, con una confirmación de Usuario/Contraseña.
* Acceso de usuario.
  + Acceso Admin. Un usuario admin, puede controlar las publicaciones de productos y curiosidades.
  + Acceso usuario. Da paso al menú principal.
* Pedir cita. El usuario puede registrar sus citas en la base de datos, para luego visualizarlas en sus animales.
* Borrar cita. El usuario puede borrar sus citas de la base de datos.
* Mis animales.
  + Registro de animales.
  + Visualización de mascotas registradas. El usuario puede visualizar una foto de su mascota, su nombre y su número de chip de un solo vistazo.
  + Borrado de mascota previa confirmación mediante un botón.
* Historial Animal. El usuario pinchando en cada mascota, puede visualizar una foto de su mascota, sus datos personales tales como (nombre, chip, nacimiento, sexo, especie, raza y propietario). También puede visualizar las citas previamente registradas con el detalle de estar ordenadas por más recientes a más antigua.
* Curiosidades. El usuario puede visualizar una serie de curiosidades de animales, así como una foto relacionada con este, previamente publicadas por el admin.
* Tienda. El usuario puede ver los productos que puede comprar.
* Cuidados e higiene. El usuario realiza una búsqueda de su mascota para sus cuidados básicos.

# ANALISIS DEL CONTEXTO.

A continuación, se detallan las herramientas utilizadas para el desarrollo del proyecto y de la aplicación. Así mismo se justifica la elección de dichas herramientas.

## Dispositivo: Android

Se ha decidido utilizar Android, de manera que la aplicación pueda ser probada en cualquier dispositivo móvil, ya que hoy en día es lo más utilizado.

## Lenguaje de programación: Kotlin

Kotlin destaca por varias características que no solo simplifican la lectura del código sino el propio desarrollo de este. Es por ello que las ventajas de Kotlin respecto a Java a la hora de desarrollar aplicaciones móviles ha hecho que este lenguaje se haga cada vez más popular.

Además, Kotlin implementa librerías que facilitan las tareas mucho más fácilmente que Java en mi opinión.

Entre ellas destaco la posibilidad de crear una base de datos SQLite con tan solo unas clases y utilizar la misma sintaxis que se utiliza en la propia base de datos MySQL.  
Por el contrario, la base de datos es local, aspecto a mejorar que mencionaremos más adelante en los problemas encontrados.

## Base de datos: SQLite

SQLite es una herramienta bastante fácil de usar como he mencionado hace unos párrafos, fácil de visualizar, facilidad para la creación de tablas e insertas datos, y facilidad en la realización de consultas.

## Consultas

Para las consultas se ha empleado todo tipo de lugares. Google como principal motor de búsqueda. Así mismo, la propia web de Android Studio para ciertas consultas.  
StackOverFlow también ha sido un buen aliado a la hora de la resolución de ciertos problemas, búsqueda de ejemplos, etc.

Como contrapartida, destaco, que al ser un lenguaje “reciente” no hay tanta información como en Java, además del añadido de que Android Studio se actualiza con mucha asiduidad y ciertas clases que antes funcionaban, luego no funcionan. Para ello la solución fue analizar el código, entenderlo, y realizar los pertinentes cambios.

## Diagramas: Draw.io y Visio

Para la realización de los diagramas he utilizado estas dos herramientas sencillamente porque han sido las utilizadas durante los 2 años de duración del curso y ya me resultaban familiares.







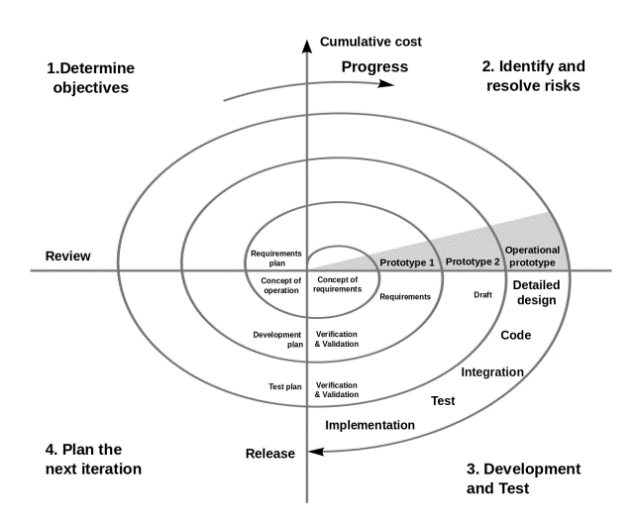




## Metodología

La metodología utilizada para este proyecto, ha sido en espiral, ya que hemos ido implantando funciones y secciones, hasta llegar al proyecto final dependiendo de su complicación.

De tal manera que se ha empezado por los objetivos principales, cumpliendo con cada una de las secciones.



# DESARROLLO DEL CONTENIDO

En este apartado es donde explicamos los detalles más relevantes de la aplicación siguiendo su curso normal. Es decir, las funciones que se ejecutan cuando el usuario interactúa con la aplicación.

Previamente mostramos como está desarrollada la base de datos y su justificación.

## Análisis de la Base de Datos

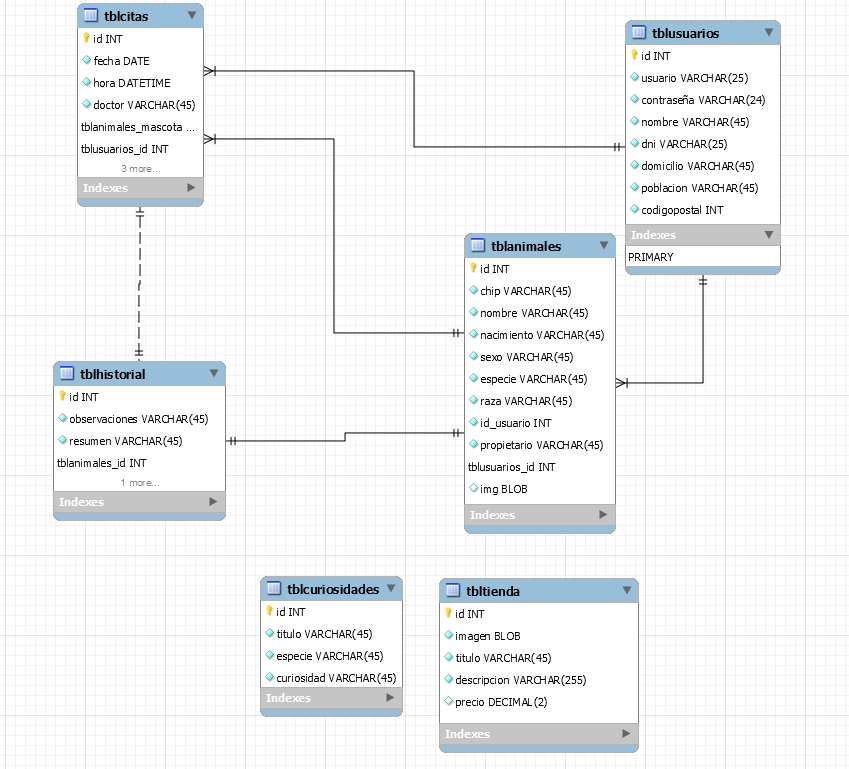
La base de datos empieza por el usuario, desde donde el cual va a partir todo, exceptuando dos tablas (curiosidades y tienda), ya que son secciones apartes que el usuario simplemente tiene acceso a ellas, pero no se relacionan.

El usuario, recalco que es desde donde va a empezar todo, ya que al hacer “login”, tendremos que recoger el usuario y la contraseña del usuario que ha hecho acceso. De tal manera que la tabla “tblanimales” interactúa con el usuario para así registrar sus animales y mostrarlas según el ID del usuario que ha hecho “login”.

A su vez, la tabla “tblanimales” interactúa con la tabla “tblcitas”, y “tblhistorial”, ya que, al pulsar en una mascota en concreto, mostraremos el historial de citas y los datos de dicha mascota.

Y la tabla “tblcitas” lo que hace es recoger el usuario que ha hecho “login”, para mostrar en un desplegable los animales de este usuario para poder registrar la cita.

Y por último, contamos con las tablas “tblcuriosidades” y “tbltienda”, las cuales hemos registrado en la base de datos, porque el usuario Administrador mencionado anteriormente es el encargado de realizar las publicaciones de productos y curiosidades, así posteriormente, el usuario puede visualizarlo.



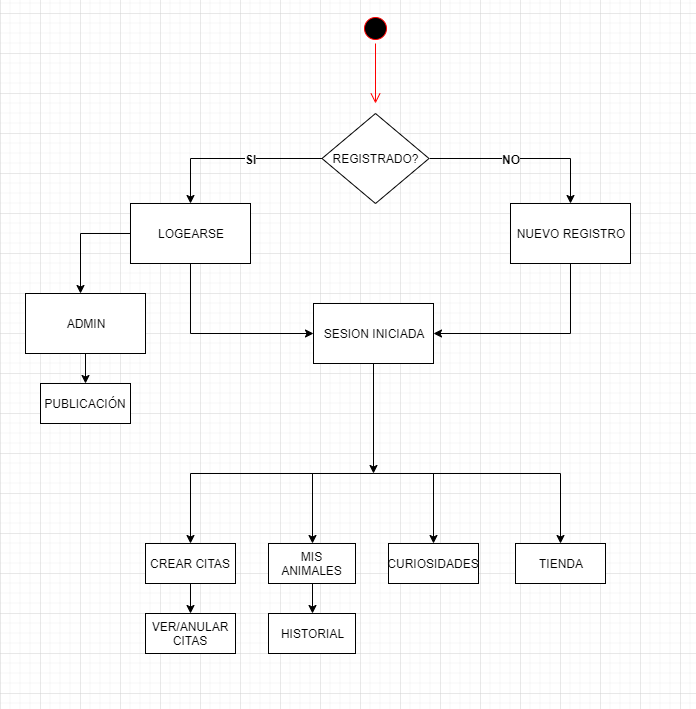
## Funcionalidad de la aplicación

A continuación, vamos a mostrar varios ejemplos del curso normal de la aplicación cuando el usuario interactúa.

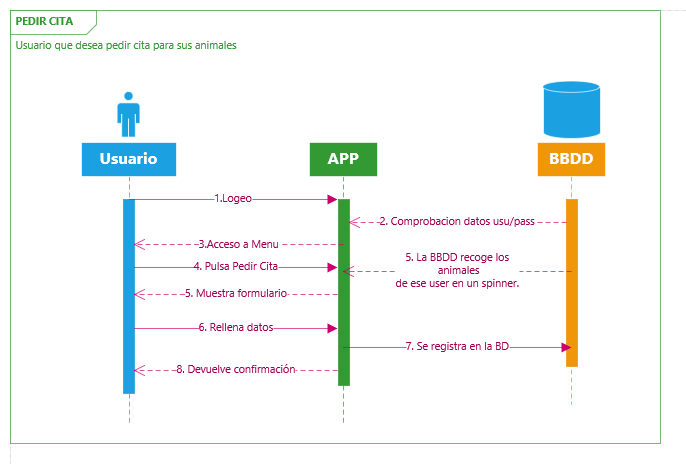
### Diagrama de flujo.

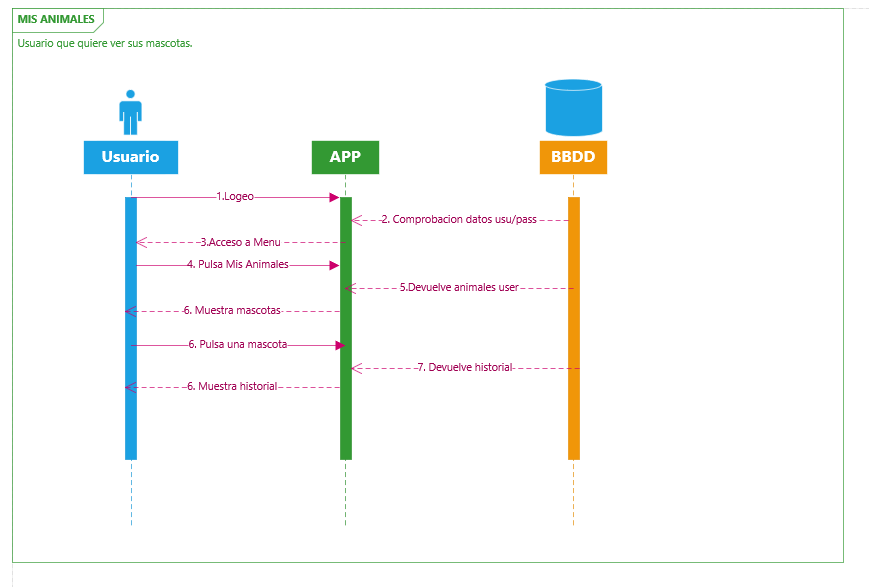
Al iniciarse la APP se le da la oportunidad al usuario de registrarse o acceder. El acceso también puede realizarse como administrador, un usuario ya creado automáticamente por el sistema donde realizar las publicaciones.

Como usuario al iniciar sesión se podrán realizar todas las tareas mostradas en la captura inferior.



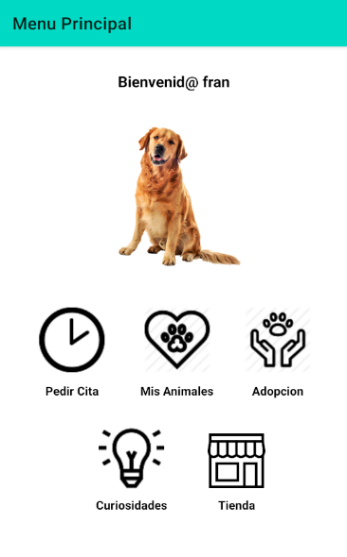
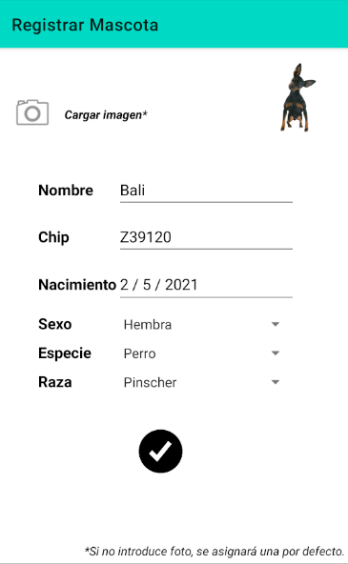
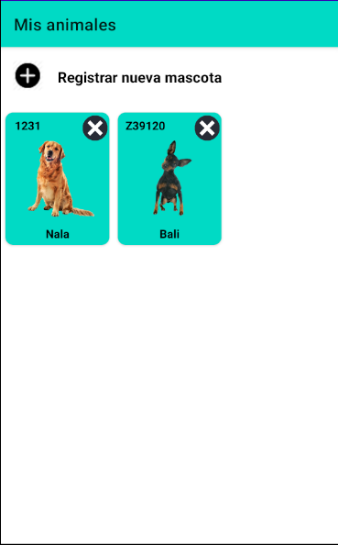
### Diagrama de Secuencia

Un usuario quiere pedir una cita.

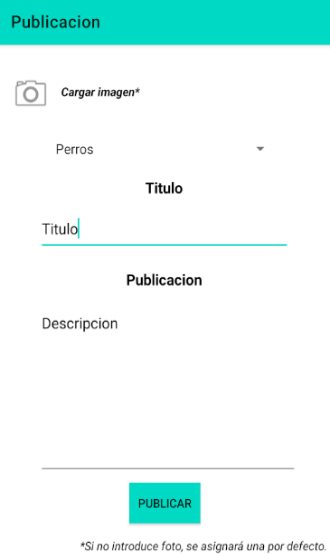
Un usuario quiere acceder a sus animales (suponiendo que ya los haya registrado).

### Funcionamiento APP (Registro, acceso, registro mascota, acceso historial)

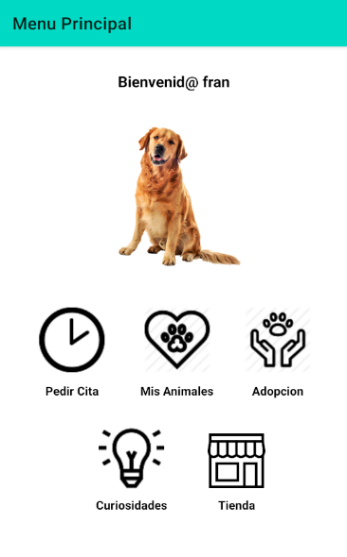
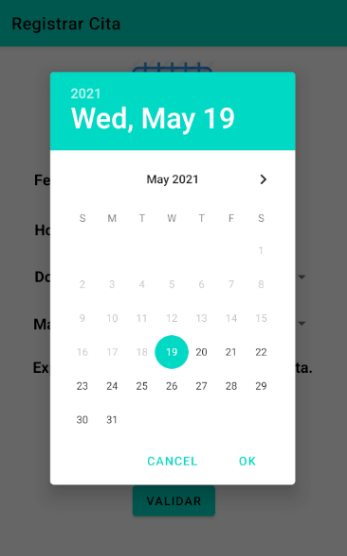
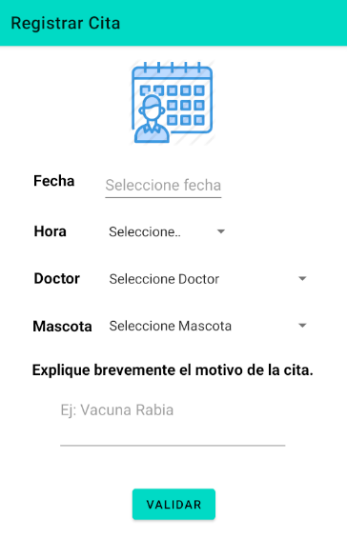




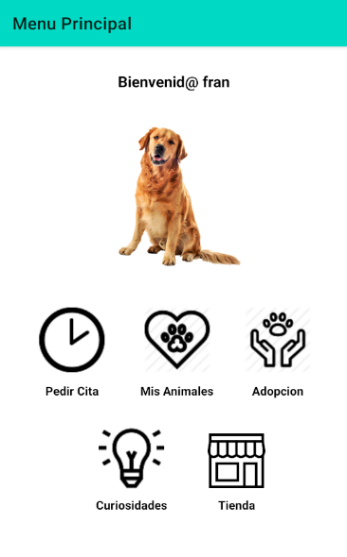
### Funcionamiento APP (Admin registro publicación/Producto).

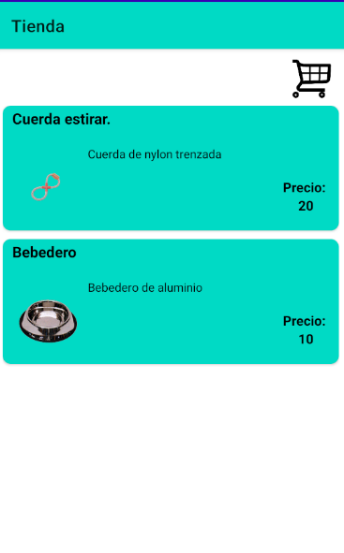
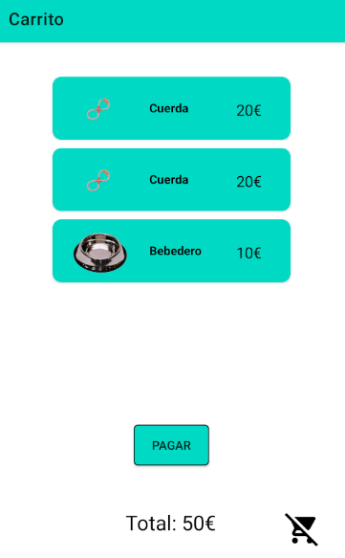


### Funcionamiento APP (Pedir cita).



### Funcionamiento APP (Tienda).



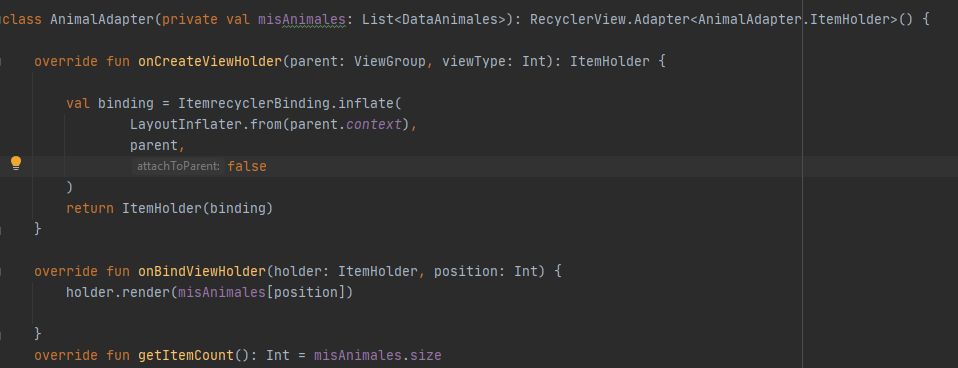
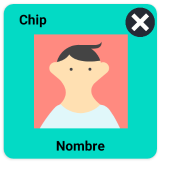


## Clases principales de la aplicación.

### Adapters en un RecyclerView

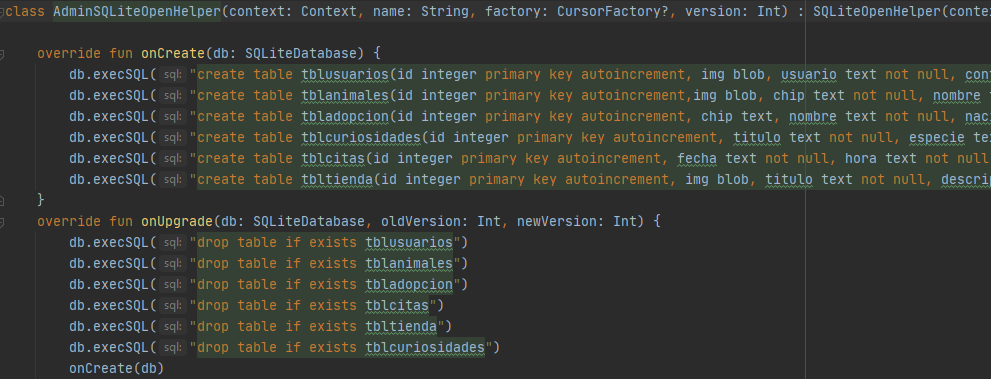
Estos Adapter realizan la función de implementar los datos de mi BD en un data Class y a su vez se implementan en el adapter. Cada animal, cita, curiosidad o producto se adapta al .xml con el diseño que le hemos proporcionado uno seguido de otro.





### Creación de la BD.

De esta clase, hay poco que añadir. Android Studio dispone de librerías para implementar la BD (AdminSQLiteOpenHelper) en la cual, con sentencias SQL podemos realizar consultas, updates, creación de tablas, etc.



## Problemas encontrados

Me he encontrado un sinfín de problemas y errores, incluso aquellos que “ocurren sin haber tocado nada” dado que Android Studio es un entorno maravilloso pero que requiere de constantes actualizaciones. También tiene problemas de caché que puede llevar a cabo errores.

Problemas/errores entre otros:

* Problemas de caché (almacena en memoria algo que ya no existe y por lo tanto da error).
* Conexión BD (No conecta, error de conexión, no responde).
* Consultas BD, borrar registros, etc...
* Desconocimiento sobre algo, o como resolver un error.
* Creación BD. Al ser local, cuando he borrado la APP de Android Studio automáticamente la BD se borra del dispositivo móvil a no ser que la guardes previamente en otro lugar.
* Cargar imagen en la base de datos y posterior lectura en el dispositivo por incompatibilidad de formatos (resuelto en sección ampliación).
* Cargar un RecyclerView en la tienda al pulsar un producto y cargarlo en la Cesta.
* Problemas de Activity con los adapter ya que funcionan de manera diferente. Hay que sacar el context de la activity que lo ejecuta en ocasiones.

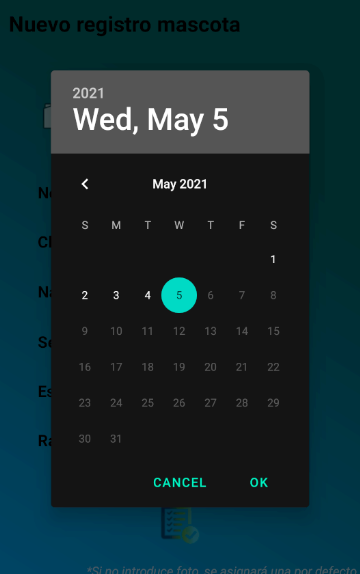
Aun a pesar de haber encontrado muchos problemas, he ido resolviendo la mayoría que se me han presentado, algunos simplemente los he omitido y lo he resuelto de una manera más sencilla

## Ampliación

En esta sección voy a detallar una serie de cosas que me han resultado interesantes o he aprendido a usarlas y he implementado a raíz de la búsqueda y aprendizaje.

#### Implementación Calendario.

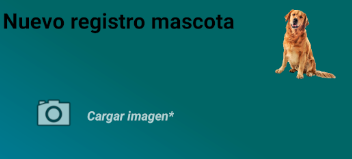
He añadido un Calendario, para 2 cosas en concreto. La creación de citas, y el registro de nacimiento del animal.

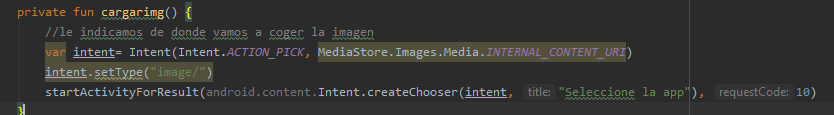


Como se puede ver en esta imagen, algo interesante, es la creación de un calendario, en el que no puedes seleccionar fechas hacia delante, por ejemplo, para el nacimiento, ya que si registras un animal es porque ya ha nacido. Así mismo, solo puedes seleccionar 20 años hacia atrás como fecha límite.

En cambio, al pedir cita para un animal solo se puede hacia delante.

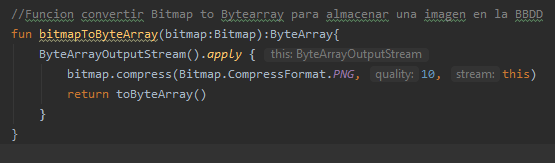
#### Cargar una imagen de la galería y conversión.

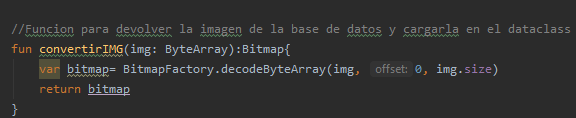
También he implementado un método que te permite coger fotos de tu galería. Lo he usado principalmente para las fotos de las mascotas y las publicaciones.

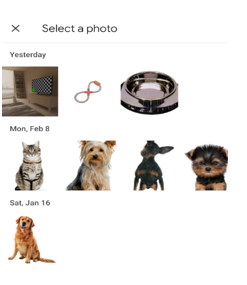
Al pulsar en el botón “se activa” este fragmento de código. 

Como anotación en base al problema antes descrito, me resultaba imposible guardar las imágenes en formato BLOB para la base de datos y luego leerla en el dispositivo. Para ello investigando encontré con una manera compatible de hacerlo.

Declararla como Bitmap y extraerla como ByteArray.



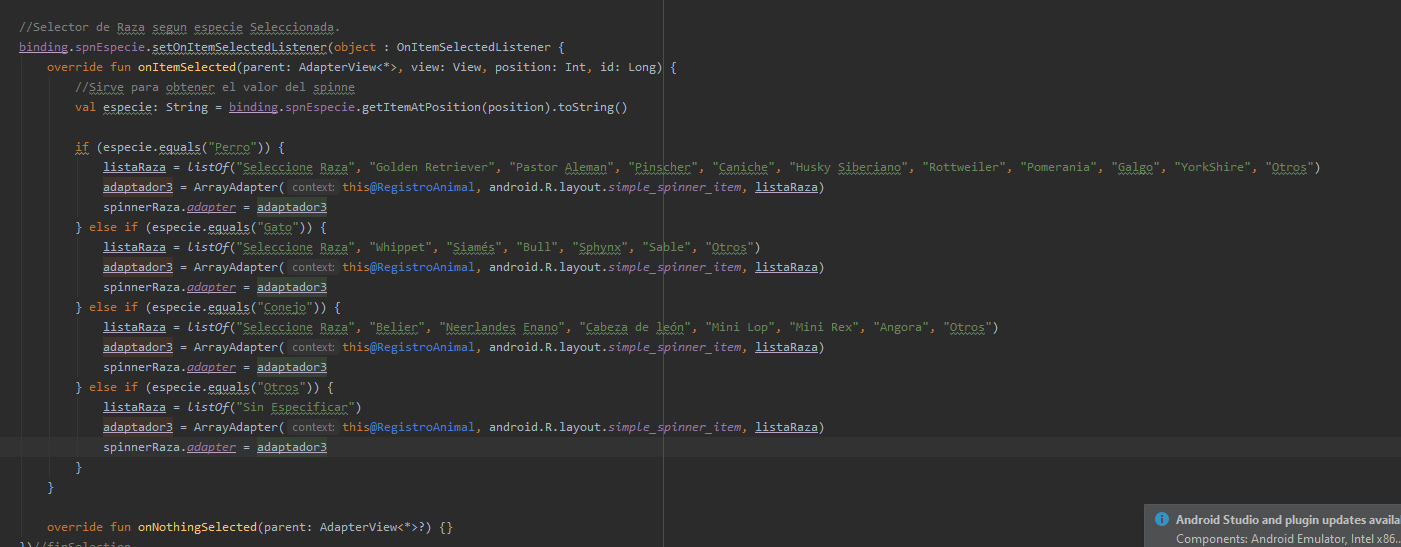
Convertir la imagen de ByteArray a Bitmap para incluirla en el DataClass.



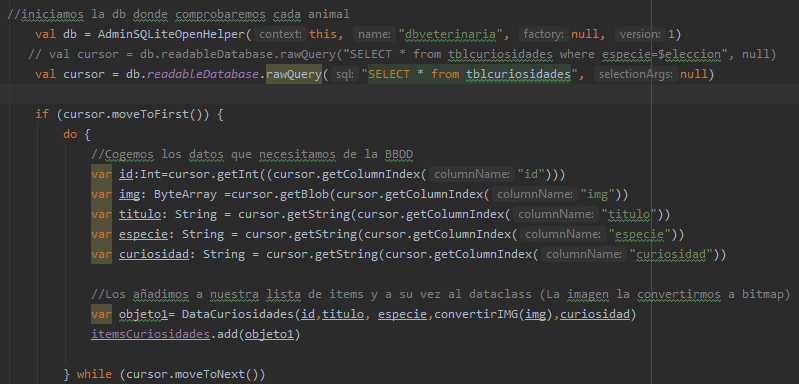
Como se puede observar, puedo acceder a la galería al pulsar en el icono de la cámara, y cargar la foto que posteriormente aparece arriba a la derecha. Si el usuario no coloca ninguna foto, el sistema pone una por defecto.

#### Modificación de Spinner sobre otro Spinner seleccionado.

El principal problema que tuve, es la modificación de un spinner en tiempo real. Encontré la solución con la función onItemSelected() que coge la posición del spinner a verificar y según su posición modifiqué el contenido del spinner de la raza.



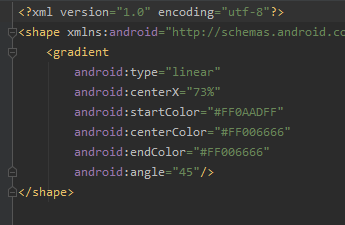
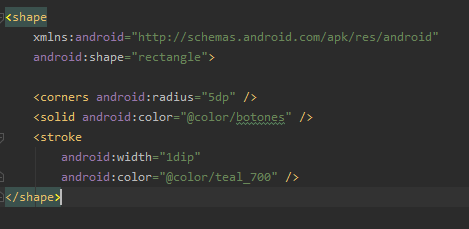
#### Consultas y creación base de datos.

Se inicia la BD en la cual agregamos un cursor de lectura. Si el cursor está en primera posición moveToFirst () va leyendo mientras haya siguiente moveToNext ()

Cogemos los datos cogiendo la columna de la tabla que hemos leído poniendo el columnName y los metemos en una variable.

Una vez leído, lo añadimos en el DataClass, que a su vez lo coloca en el RecyclerView.

#### Guardar drawables para su posterior uso.

Dos ejemplos de creación de drawables que luego podemos implementar en por ejemplo un botón colocando en background @drawable/nombrexml

# CONCLUSIÓN Y VALORACION PERSONAL

Para finalizar el desarrollo del proyecto, se exponen las conclusiones del proyecto.  
Además se hace una valoración personal del proyecto y se exponen las posibles futuras líneas de trabajo en caso de continuar con el proyecto.

## Conclusiones del proyecto

Una vez finalizado el proyecto, podemos concluir que se han cumplido los objetivos marcados con creces en la mayoría de las secciones, tanto a nivel personal como de desarrollo.

Las funcionalidades básicas que se querían incorporar desde el principio y la esencia del proyecto se han podido llevar a cabo, otras directamente las he descartado, porque me parecían poco útiles.

Como conclusión, sobre el proyecto, y bajo mi opinión exclusivamente personal. Me parece una buena opción, con un diseño de interfaz sencillo pero intuitivo, donde con tan solo unos minutos de prueba de la app, te puedes hacer con todas las funciones de la aplicación.

Cabe destacar, la falta de aprendizaje sobre diseño, pero aun así, me siento satisfecho en cierta medida dado que al final muchas veces, se aprende a base de darle uso a las cosas.

Como valoración negativa sobre el proyecto, exclusivamente destacaría el uso de una base de datos local, ya que, realmente, en una aplicación de estas características, a no ser, que fuera meramente informativa para que el usuario pudiera guardar sus citas, pero no pedirlas a una clínica, la base de datos nunca podría ser local. Pero dada la complicación al principio para la creación y conexión de la base de datos, siendo además lo primero que tenía que crear, me resultó imposible este objetivo y cuando el proyecto ya estaba suficientemente avanzado, me resultaba imposible intentarlo de nuevo y cambiar todo el código.

También destacar sobre este último aspecto, el uso de Firebase para la creación de la base de datos, que fue sencillo, pero no relacionaba tablas, y fue descartado.

Y para finalizar, me hubiera gustado o me gustaría haber podido implementar más funciones.

## Valoración personal

A nivel personal y como se comenta al principio, me parecía bastante interesante la realización de un proyecto para dispositivo móvil. Además de ser un sector en auge, Rafa supo transmitirnos su entusiasmo, su enseñanza, sus consejos, etc.

Así mismo, gracias a esto último, también ha sido un factor a la hora de realizarlo en Android Studio y con el lenguaje Kotlin.

Dicho esto, también agradecer al profesorado por su ayuda y disposición tanto en este proyecto, como durante el curso.

Me siento satisfecho también, de empezar pensando que no podría hacerlo y durante el proyecto no haber precisado casi ayuda y haber podido solucionarlos gracias a la búsqueda, aprendizaje, ensayo y error.

Todo este proyecto me ha servido para coger cierta experiencia en la lógica de la programación, ya que me he dado cuenta que es imposible realizar todas las cosas de memoria, sino que vas “cogiendo de aquí y de allí” y tienes que tratar de comprender la lógica y en este aspecto, el objetivo se ha cumplido con creces.

De esta forma, si observo a lo lejos la curva de aprendizaje, y como empezó el proyecto que empecé en diciembre de 2020, y en el cual le he ido dedicando tiempo poco a poco y sin prisas, estando varios días seguidos durante horas, y después, estando hasta 2 semanas sin tocar el proyecto, me siento satisfecho con el resultado.

Para finalizar, destaco, que, aunque me sienta satisfecho con el resultado final, no quiere decir, que no me gustaría mejorar bastantes aspectos, sobre todo, más que en las funciones, empezar de 0 y realizar un código más eficiente, más limpio y mejor ordenado.

# BIBLIOGRAFÍA

Aquí, indico algunas de las webs visitadas para el aprendizaje del proyecto:

**Para algunos iconos**

<https://icon-icons.com/es/>

**Para dudas generales**

https://www.google.com/

<https://es.stackoverflow.com/>

<https://www.w3schools.com>

<https://developer.android.com/>

**Para la inserción de un calendario**

<https://programacionymas.com/>