



CICLO FORMATIVO DE GRADO SUPERIOR  
DESARROLLO DE APLICACIONES MULTIPLATAFORMA

PROYECTO FIN DE CICLO

---

# **Animal Quiz**

**Autores**

**CURSO 2019-20**

**TÍTULO :** Animal Quiz

**AUTORES:** Marco González López

Nicolas Martínez Cabrera

Juan Carlos Molina Pedraza

**TUTOR DEL PROYECTO:** Ernesto Ramiro Córdoba

**FECHA DE LECTURA:** 08/12/2020

En Madrid

Ernesto Ramiro Córdoba  
Tutor del PFC

## **RESUMEN:**

Esta Aplicación nace en la cabeza de tres jóvenes estudiantes con ambiciones similares y en busca de algo distinto a lo habitual.

Animal Quiz es una aplicación didáctica y llamativa para todo tipo de público, pero sobre todo orientada para niños en un ámbito escolar.

¿Que buscamos con esta aplicación? Basándonos en las carencias que hay en la enseñanza, debido que, se sigue enseñando con la ley Moyano 1857, hemos visto una oportunidad en el mercado para desarrollar una aplicación de uso sencillo y útil, pero muy didáctica, en la cual los alumnos podrán aprender de forma sencilla y divertida, las diferentes especies de animales que existen y su ubicación actual en el planeta.

En la actualidad los niños pequeños desconocen muchas de las especies, con esta app lo que queremos hacer es que vean la gran variedad de fauna que hay en la tierra.

Ahora los niños no tienen la libertad que había hace unos veinte años, que estaban más en el campo y podían explorar por ellos mismos dicha fauna.

Esta aplicación como hemos dicho anteriormente va dirigida sobre todo para niños pequeños, se puede utilizar tanto en el ámbito educativo, como extraescolarmente.

Nuestra app, aparte de ser llamativa, tiene una parte bastante interactiva con el usuario, ya que dispone de un quiz de preguntas 66 para poder reforzar los conocimientos que se han adquirido de una forma fácil y divertida. Esto hará que el quiz sea más ameno y divertido y no sea tomado como si fuese un examen.

El objetivo de Animal Quiz es ser una aplicación de entretenimiento y aprendizaje al mismo tiempo.

## **ABSTRACT:**

This Application was born in the head of three young students with similar ambitions and in search of something different from the usual.

Animal Quiz is a didactic and eye-catching application for all types of public, but above all oriented to children in a school environment.

What are we looking for with this application? Based on the shortcomings in teaching, because it continues to be taught with the Moyano 1857 law, we have seen an opportunity in the market to develop a simple and useful, but very educational application, in which students can learn in a simple and fun way, the different species of animals that exist and their current location on the planet.

At present, young children are unaware of many of the species. With this app we want them to see the great variety of fauna that exists on earth.

Now children do not have the freedom that there was some twenty years ago, they were more in the countryside and could explore this fauna for themselves. This application, as we have said before, is aimed above all at young children, and can be used both in the educational field and outside of school.

Our app, apart from being eye-catching, has a quite interactive part with the user, as it has a quiz of questions to be able to reinforce the knowledge that has been acquired in an easy and fun way. This will make the quiz more enjoyable and fun and not be taken as if it were an exam.

The aim of Animal Quiz is to be an application for entertainment and learning at the same time.

## **AGRADECIMIENTOS**

En primer lugar, queríamos agradecer a todos los profesores del grado por la implicación que han tenido en nuestro desarrollo a lo largo de estos dos años. Poner un énfasis en una persona que ha sido nuestro tutor y guía del proyecto (Ernesto Ramiro) quien nos ha ido orientando a lo largo de este proceso, ayudándonos en todos los aspectos del proyecto, tanto en la parte profesional como en la parte personal.

En estos momentos difíciles que hemos pasado de covid, en este duro confinamiento que nos ha tocado vivir a todos, no ha sido fácil adaptarse a la "nueva normalidad". Por esto también agradecer a nuestros seres queridos que nos han apoyado en estos duros momentos, de bajadas y subidas, etc.

El último agradecimiento que queríamos mencionar en este apartado, nos lo queríamos dedicar a nosotros. Como mencionábamos antes en esta dura situación en la que nos encontramos y en la que se encuentran las empresas, hemos estado mucho tiempo con la incertidumbre de nuestro inicio en las FCT, y aun así invirtiendo todo nuestro tiempo en el desarrollo de este proyecto, agradecimiento al EQUIPO ANIMAL QUIZ.

Por lo tanto, este agradecimiento a los profesores implicados y a los compañeros que nos han acompañado durante estos dos años.



Esta obra se distribuye bajo una licencia Creative Commons.

Se permite la copia, distribución, uso y comunicación de la obra si se respetan las siguientes condiciones:

- Se debe reconocer explícitamente la autoría de la obra incluyendo esta nota y su
- enlace.
- La copia será literal y completa
- No se podrá hacer uso de los derechos permitidos con fines comerciales, salvo permiso expreso de los autores.

El texto precedente no es la licencia completa sino una nota orientativa de la licencia original completa(jurídicamente válida) que puede encontrarse en:

<http://creativecommons.org/licenses/by-nc-nd/3.0/deed.es>

## INDICE

### 1. INTRODUCCIÓN

1.1. Objetivos	1
1.2. Motivación	1
1.3. Antecedentes	2

### 2. DESARROLLO DEL PROYECTO

2.1. Herramientas tecnológicas	3
2.2. Planificación	3
2.3. Descripción del trabajo realizado	3
2.4. Resultados y validación	4

### 3. CONCLUSIONES

3.1. Innovación	5
3.2. Trabajo futuro	5

### 4. BIBLIOGRAFÍA Y WEBGRAFÍA

### 5. ANEXOS

5.1. Presentación general de un informe	I
5.2. Directrices particulares para el proyecto	II
5.3. Directrices particulares para los listados de programas	II
5.4. Algunas reglas mecanográficas	IV

## 1. INTRODUCCIÓN

El propósito de PFC es desarrollar una aplicación en Android studio.

En la cual hemos un meticuloso estudio de mercado, para comprobar las diferentes competencias a las que nos podíamos enfrentar de cara a este proyecto que decidimos en una reunión conjunta.

Para poder realizar este estudio. La primera idea que se nos ocurrió fue, buscar diferentes aplicaciones que fuesen similares a la nuestra y que estuviesen actualmente operativas.

Encontramos las siguientes apps:

- Feas apps
- Mama papa
- Andrey Solovyev
- Papumba
- CLEVERBIT
- Abuzz
- micky appz

Tras encontrar todas estas apps y ver su contenido, vimos una oportunidad de crear algo distinto. Algo lo cual saliese de lo habitual.

Una de las primeras diferencias importantes con respecto a las otras aplicaciones de animales son las siguientes es que nuestra aplicación lleva implementada un mapa, en el cual sale la ubicación del animal, dando sus coordenadas de exactas de donde se ubica.

El quiz es otro elemento que nos diferencia de la competencia, dado que las otras aplicaciones, no disponen de un test tan exhaustivo como el nuestro, donde pueden ver, si han adquirido sus conocimientos.

La sencillez de nuestra app hace que los usuarios puedan disfrutar de ella, esto hará que no sea una aplicación que se descarguen y la desinstalen al momento.



## 1.1. Objetivos

### **Funcionalidad de la aplicación:**

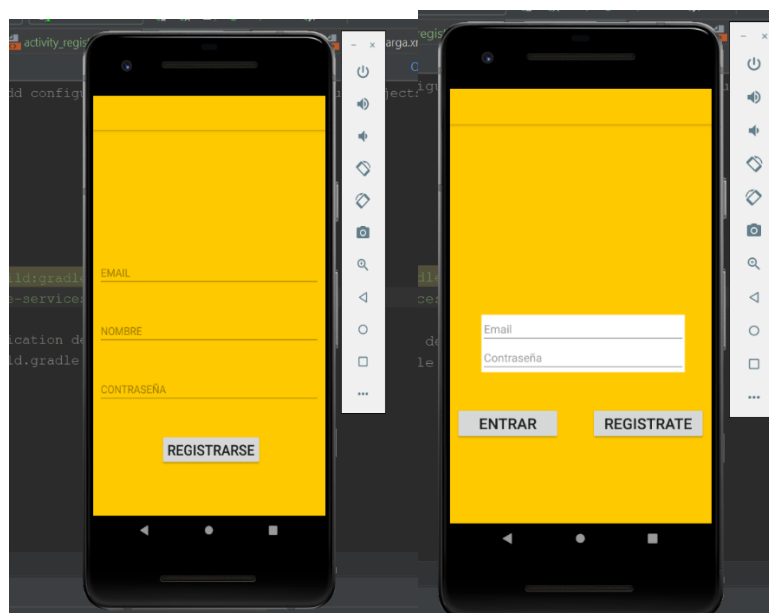
- Login/registro usuario mediante Firebase
- Pantalla de carga
- Interfaz de selección de especies
- Listado de animales
- Menu drawer
- Desarrolladores
- Google Maps
- Quiz

### **LOGIN Y REGISTRO:**

En el login accedemos a través de un usuario y una contraseña, si el usuario está registrado entrará directamente en la aplicación.

En el caso de que el usuario no esté registrado, automáticamente le mandará a otro activity en el cual le solicitará una serie de datos para poder registrarse en la aplicación.

Una vez registrado, podrá acceder sin ningún problema a la y poder disfrutar de ella.

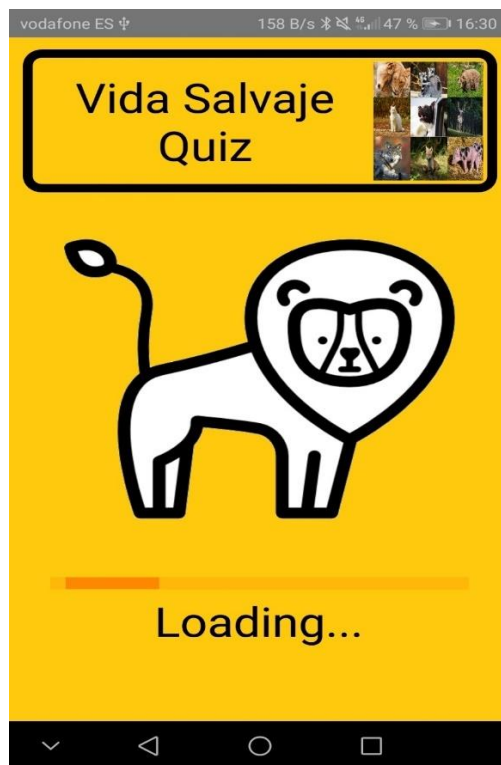


## PANTALLA DE CARGA:

La pantalla de carga es el primer activity que se inicia al arrancar la aplicación, en el cual muestra el logo inicial de la app y una pantalla de carga mediante progressBar.

Una vez cargado esta pantalla inicial se irá al login de registro.

En esta pantalla podremos ver el logo de la aplicación y un pequeño eslogan anticipando el contenido de la aplicación

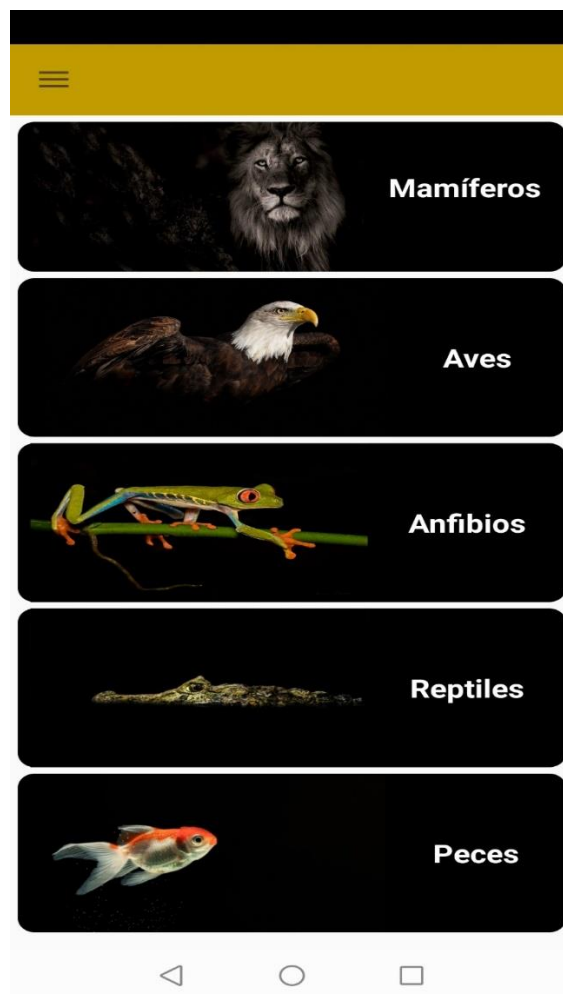


## INTERFAZ PRINCIPAL:

En la interfaz principal, mostramos las diferentes categorías de animales (mamíferos, aves, anfibios, reptiles, peces)

Pulsando la categoría de la siguiente especie mencionadas, te dará acceso a otro activity, el cual te mostrará más información.

En la parte superior del activity mostramos un menú en el que incorporamos las siguientes funciones (quiz, desarrolladores, Google maps, cerrar aplicación).



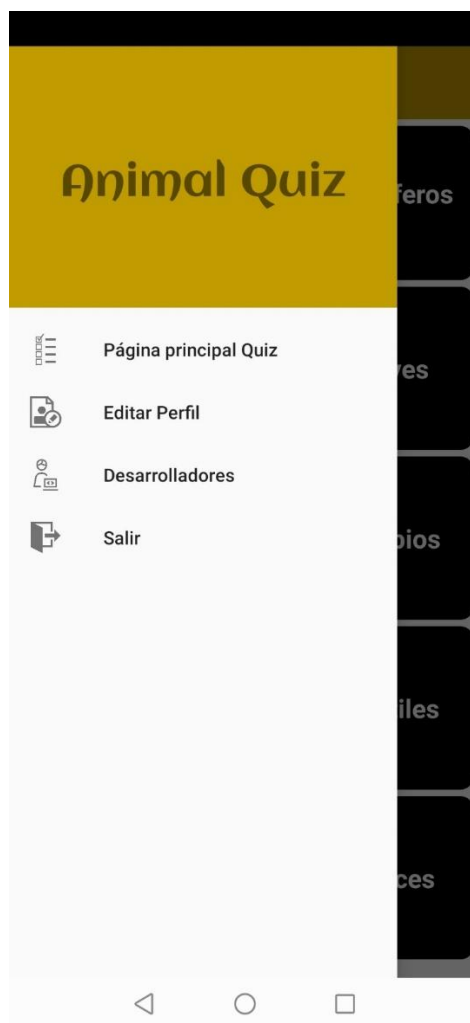
## LISTADO DE ANIMALES:

En este activity, nos va llevar a la información detallada de cada especie. Si clicamos en mamíferos saldrán todos los animales implementados en ella, al clicar en uno de ellos nos llevará al activity de cada animal, detallando los siguientes ítems.



## MENU DRAWER:

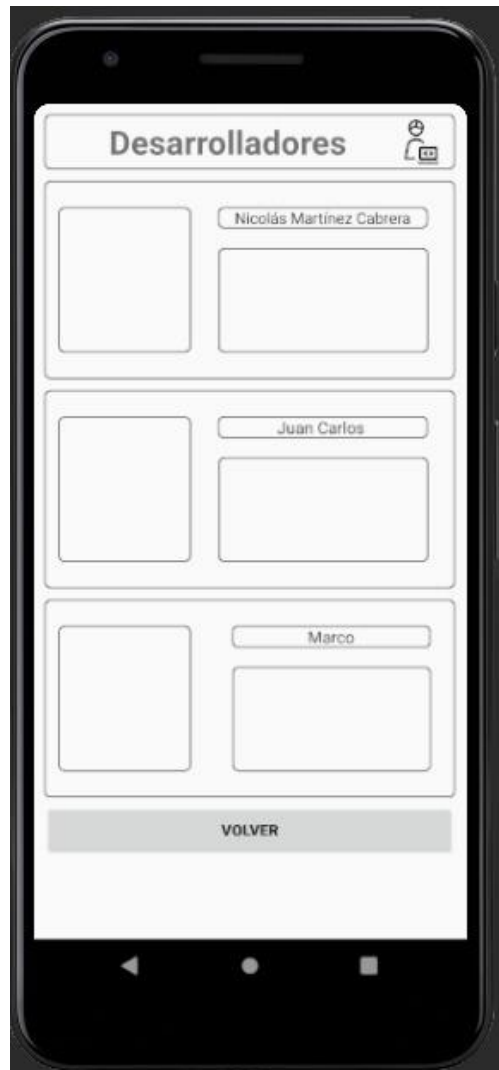
Habr  un men  drawer implementado en la parte arriba izquierda de la aplicaci n, en todas las pantallas, en el cual podremos acceder a las siguientes activities.



## DESARROLLADORES:

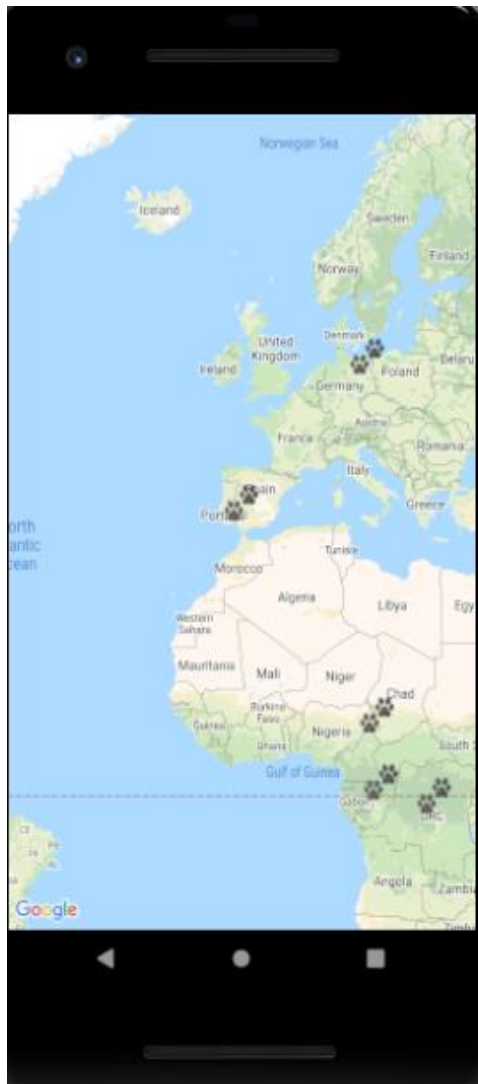
En el siguiente activity mostramos quien han sido los creadores de la aplicación, una foto de cada uno de nosotros y una breves descripción de quien somos, lenguajes adquiridos durante el curso.

Según vaya avanzando en estos meses el desarrollo de esta app, iremos metiendo más contenido en ella.



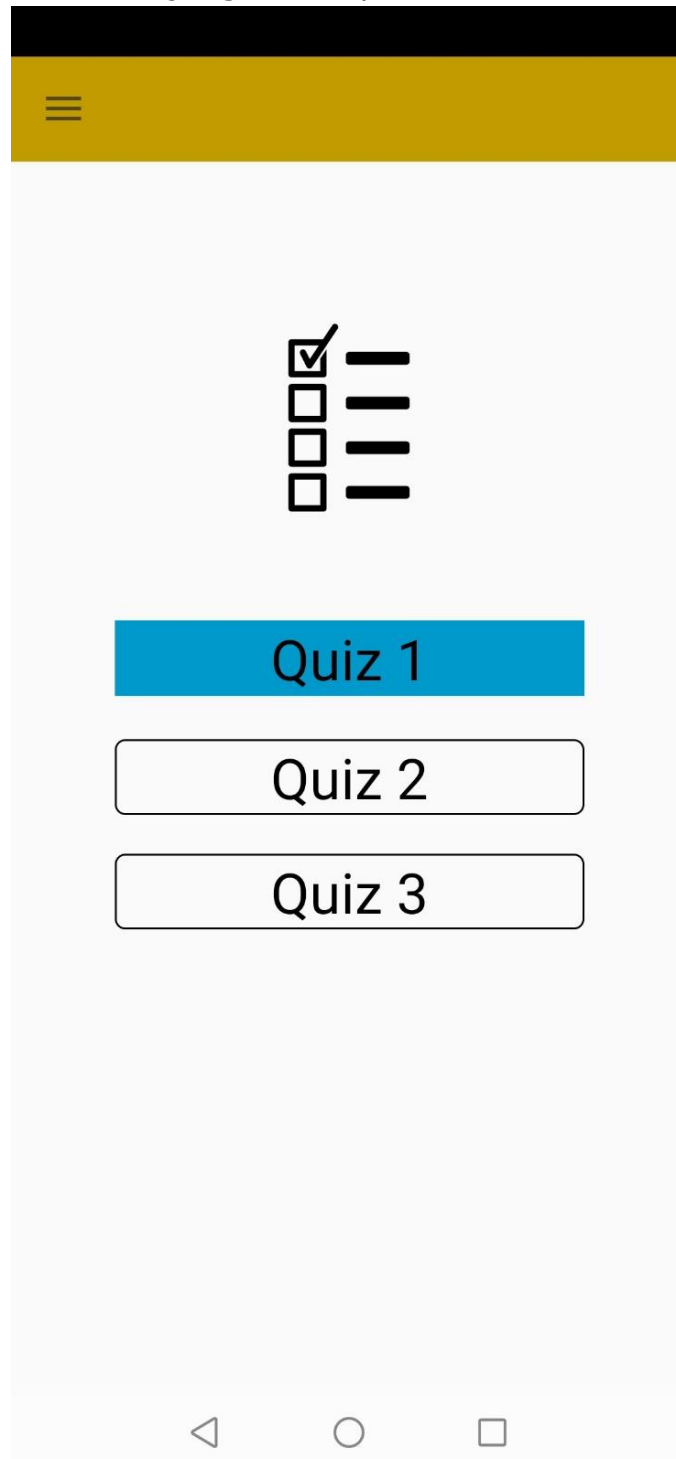
## GOOGLE MAPS:

En esta pantalla vemos la ubicación de los siguientes animales, que hemos metido en la aplicación.



## QUIZ:

En esta pantalla tendremos la opción de que los usuarios, hagan un quiz de preguntas, en el cual ellos puedan ver los conocimientos que han adquirido durante la instancia en la aplicación, creemos que este es uno de los puntos fuertes, ya que le da mucho juego a la aplicación.







## 1.2. Motivación

Nuestra primera razón por la que empezamos este proyecto fue debido a las PFC.

Para poder sacar una idea de proyecto, pensamos que lo mejor sería hacer una lluvia de ideas brainstorming.

En la cual cada uno expuso unas cuantas ideas e hicimos un debate para seleccionar las tres mejores ideas.

Una vez seleccionadas las tres mejores ideas, hicimos una valoración de impacto mercantil, para concretar nuestra decisión y así poder seleccionar la idea adecuada.

Gracias a esto nació lo que hoy presentamos aquí ¡ANIMAL QUIZ!

Nuestro principal objetivo actualmente es que nuestro proyecto agrade al jurado que lo va a valorar y a su vez poder sacarle partido en un futuro.

Los objetivos que nos planteamos a corto plazo, ya que hemos tenido unos 3 meses más o menos han sido muy claro.

La aplicación tenía que funcionar, hemos puesto mucho énfasis en la funcionalidad de la aplicación, ya que es una parte importante para que una aplicación sea eficaz.

Otro de los grandes detalles importantes es el aspecto de la aplicación, como muy bien no ha ido insistiendo nuestro tutor.

Ya que una aplicación visualmente llamativa siempre es más vistosa y en la cual te van a dar más ganas de seguir interactuando con ella.

Siguiendo con los objetivos de la aplicación, queremos enumerar paso a paso los más importantes.

En primer lugar, una interfaz de carga que fuera llamativa, con el slogan de la Aplicación y el logotipo.

Una pantalla de login, la cual puedas, logearse o registrarse.

Iniciado este proceso llegaremos a la pantalla principal, con sus siguientes apartados para poder elegir el tipo de especie del cual quieres aprender.

Una vez seleccionado el tipo de especie, se abrirá otra pantalla la cual podrá seleccionar el animal que quiera y encontrar toda la información de dicho animal, al igual que tendrá acceso a escuchar el sonido del animal.

Uno de los objetivos más ambiciosos y costosos han sido los siguientes.

Implementar un Quiz, en el cual el usuario tendrá test aleatorios de preguntas, sobre la información que viene de los animales. Esto le dará un toque a nuestra aplicación diferencial con el resto de las del mercado.

Seguido de otro, que es nuevo en este tipo de aplicaciones. Que es, mostrar la ubicación de dichos animales, que tenemos implementados en nuestra app.

Una de nuestras mayores motivaciones hacer esta app, ha sido que los tres somos amantes de la naturaleza y de los animales, esto hace que.

El trabajo de investigación por las especies y el desarrollo de la app, haya sido mucho más eficiente y conciso.

A quien de nosotros no nos hubiera gustado tener de pequeños, algo así.

Una aplicación en la cual se pueda aprender y a la vez disfrutar del mundo animal.

Todas estas cosas nos han hecho, investigar más sobre el mundo animal y empaparnos de todo lo relacionado.

Tanto donde viven, que comen, su hábitat, sus depredadores.

### 1.3. Antecedentes

En este apartado vamos a hablar un poco de la historia de cómo surgió Animal Quiz. Para comenzar a desarrollar la aplicación primero tuvimos que exponer nuestras ideas para poder tener un denominador común.

A la hora de desarrollar la aplicación salían muchas ideas, pero luego surgía la pregunta de, como se podrá plasmar eso en la Aplicación. Que lenguajes utilizar, que librerías implementar para poder utilizar todo, etc.

Para el proyecto decidimos utilizar Android studio y realizar una aplicación móvil funcional, una vez desarrollada la idea vimos que a la hora de hacer la pantalla del login, queríamos registrar con Firebase los usuarios, para que luego poder almacenar sus datos, sus resultados en los quiz etc. Pero esa parte final la vamos a dejar para más adelante, ya que la seguiremos dando funcionalidad poco a poco, una vez se hayan finalizado las practicas y se pueda tener más tiempo para hacer el proyecto.

Otra de las cosas que hemos incorporada es la implementación de un mapa, Google maps, en el cual vamos a mostrar mediante una ubicación, metida por coordenadas, el lugar donde se encuentra esa especie.

Una vez explicado un poco, lo utilizado, vemos que el contexto de esta aplicación es mostrar algo diferente a lo que hemos encontrado ya en el mercado, como hemos puesto en uno de los puntos anteriores.

Apartado número 1 Introducción, observamos que estas aplicaciones son la mayoría visualmente muy llamativas, pero creemos que la funcionalidad de la aplicación es muy básica, unas solo muestran información de animales, otras tienen el sonido con la foto de los animales.

Por eso nosotros quisimos enfocarnos en hacer una aplicación que fuese más completa en ese sentido, sabiendo que teníamos menos de 3 meses en el desarrollo de ella, y esas aplicaciones han tenido detrás bastantes más meses de desarrollo.

Pero creo que hemos conseguido nuestro objetivo, principalmente que funcionase, ya que era algo que teníamos muy claro desde el principio, lo segundo y por ello no menos importante, darle a la aplicación un toque personal, que fuera bonita y llamativa, que una vez la hayas descargado, te llame para que puedas interactuar y seguir navegando por ella.

## 2. DESARROLLO DEL PROYECTO

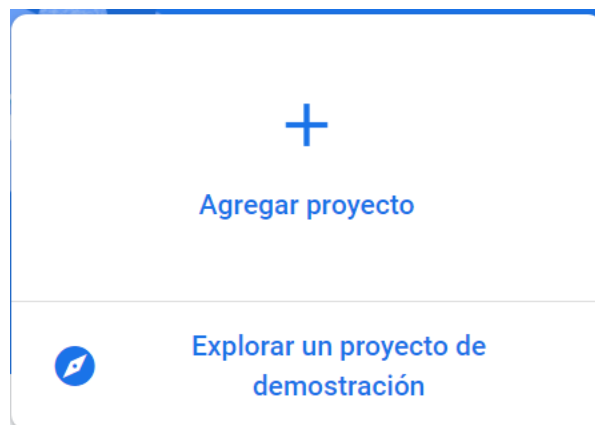
### 2.1. Herramientas tecnológicas

#### **Firebase:**

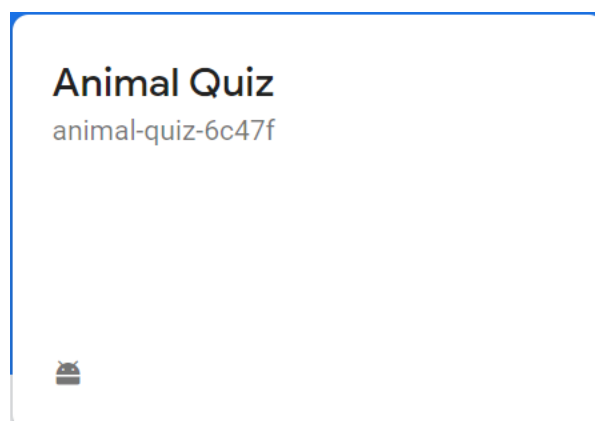
Esta herramienta es una de las principales que hemos utilizado, ya que estamos creando una base de datos en la cual almacenaremos usuarios a través de un mail y una contraseña.

A continuación, le mostraremos y explicaremos los siguientes pasos realizados hasta concluir la tarea con Firebase.

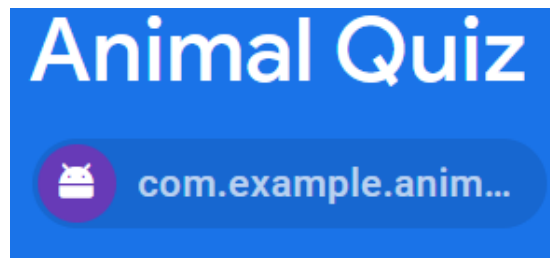
En primer lugar, se accederá pulsando esta pestaña, la cual te solicitará una serie de datos para poder crear correctamente el proyecto en Firebase.



En la siguiente imagen mostramos el proyecto ya creado y preparado para configurar con Android studio. Pulsando el icono que viene en la parte inferior, podrá acceder a la configuración de la base de datos



En esta imagen podremos apreciar el proyecto configurado, con el paquete ya asignado en Android studio



Una vez configurada la base de datos y realizada la creación del proyecto, pasaremos a la autenticación de los emails, para gestionar correctamente el almacenamiento de nuestros usuarios. Este paso es uno de los pasos más importantes a la hora de la creación porque sin él, no te reconocerá que los usuarios se están registrando.

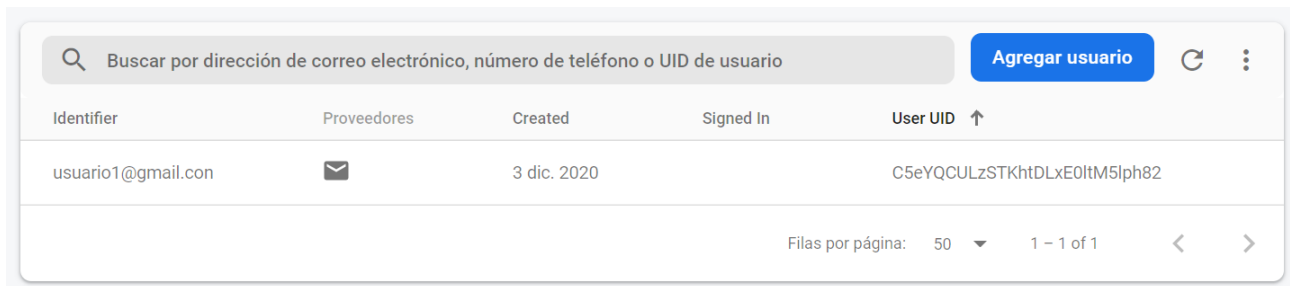
## Authentication ?

Users Sign-in method Templates Usage


Proveedores de acceso

Proveedor	Estado
 Correo electrónico/contraseña	Habilitada
 Teléfono	Inhabilitado

En esta última imagen de la utilización de la herramienta Firebase, mostramos como se registra uno de nuestros usuarios de testeo de la aplicación, para asegurarnos de que la base de datos, registra y almacena usuarios correctamente.



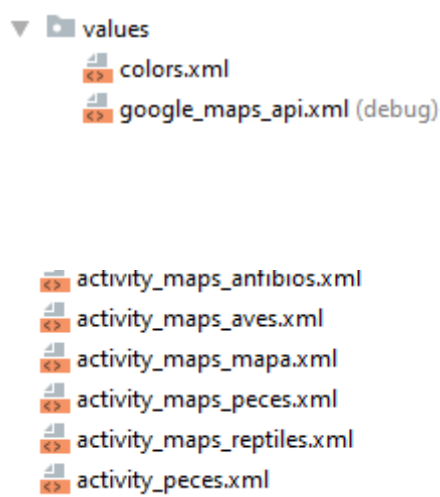
The screenshot shows the Firebase console interface. At the top, there is a search bar with the placeholder text "Buscar por dirección de correo electrónico, número de teléfono o UID de usuario". To the right of the search bar is a blue button labeled "Agregar usuario". Below the search bar is a table with the following columns: "Identifier", "Proveedores", "Created", "Signed In", and "User UID". The table contains one row of data: "usuario1@gmail.com" under Identifier, an email icon under Proveedores, "3 dic. 2020" under Created, an empty cell under Signed In, and "C5eYQCULzSTKhtDLxE0ltM5lph82" under User UID. At the bottom right of the table, there is a pagination control showing "Filas por página: 50" and "1 - 1 of 1".

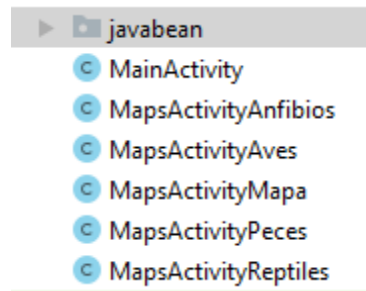
Identier	Proveedores	Created	Signed In	User UID ↑
usuario1@gmail.com		3 dic. 2020		C5eYQCULzSTKhtDLxE0ltM5lph82

Abrir una sección por cada tecnología usada o implicada en la realización del trabajo (ej.: Firebase, Java, Android, etc.). En este capítulo se pretende ofrecer un resumen destinado a lectores no familiarizados con la tecnología utilizada. Destinar más páginas según la importancia en el trabajo, de dos a cuatro páginas cada sección con todas las referencias bibliográficas que se consideren oportunas.

## Implementación de Google Maps a la aplicación:

Lo primero que tenemos que hacer es crear un nuevo activityMapaGoogle, se crean varias clases, tanto la principal, como en la carpeta values, como un xml. Que mostraremos en las siguientes fotos





Una vez creada la estructura empezamos a configurar. Nos vamos a una pagina de internet, [console.developers.google.com](https://console.developers.google.com) donde vamos a crear las apis, para que la implementación con la app sea adecuada.

Una vez creado el proyecto, nos vamos a la pestaña credenciales, que va ser la indicada de generarte la clave Api, que posteriormente vamos a pasar a nuestro xml.

#### Claves de API

<input type="checkbox"/>	Nombre	Fecha de creación ↓	Restricciones	Clave			
<input type="checkbox"/>	▲ Clave de API 1	9 nov 2020	Ninguna	AIzaSyB4jT...-jK3Mw2nk			

Nombre \*

Clave de API 1

API Key

AIzaSyB4jTI8XmXUSbJCaUJQFbdxw-jK3Mw2nk

#### Restricciones de clave



Esta clave no tiene restricciones. Las restricciones ayudan a evitar el uso sin autorización y el robo de cuotas. [Más información](#)

Para usar esta clave en tu aplicación, transfírela con el parámetro `key=CLAVE_API`.

Fecha de creación 9 de noviembre de 2020, 14:55:52 GMT+1

Creada por juancarlospedraza.1990@gmail.com (tú)

Para generar la huella digital y el paquete, nos tenemos que ir al Xml de Android y buscar su línea de código que habrá que pasársela para poder acceder a nuestra Api Key y su posterior, línea de código que tendremos que pasarle, para poder ver nuestro mapa en el móvil.



Package name:

B6:0C:C0:87:9D:45:97:23:52:66:67:1E:F2:D0:77:94:CD:0C:8B:C9

SHA-1 certificate fingerprint:

B6:0C:C0:87:9D:45:97:23:52:66:67:1E:F2:D0:77:94:CD:0C:8B:C9

Alternatively, follow the directions here:

<https://developers.google.com/maps/documentation/android/start#get-key>

Once you have your key (it starts with "AIza"), replace the "google\_maps\_key" string in this file.

-->

```
<string name="google_maps_key" templateMergeStrategy="preserve" translatable="false">
```

```
AIzaSyB4jTI8XmXUSbJCaUJQFbdxw-jJk3Mw2nk</string>
```

```
</resources>
```









Una vez hecho esto, volvemos a la pagina de internet y vamos al apartado de habilitar las apis y los servicios

APIs y servicios

+ HABILITAR APIS Y SERVICIOS

Dentro de mapas buscamos una pestaña que pone mapas y acceder a una pantalla donde para finalizar le tendríamos que habilitar una serie de permisos para que funcione nuestro Google Maps.

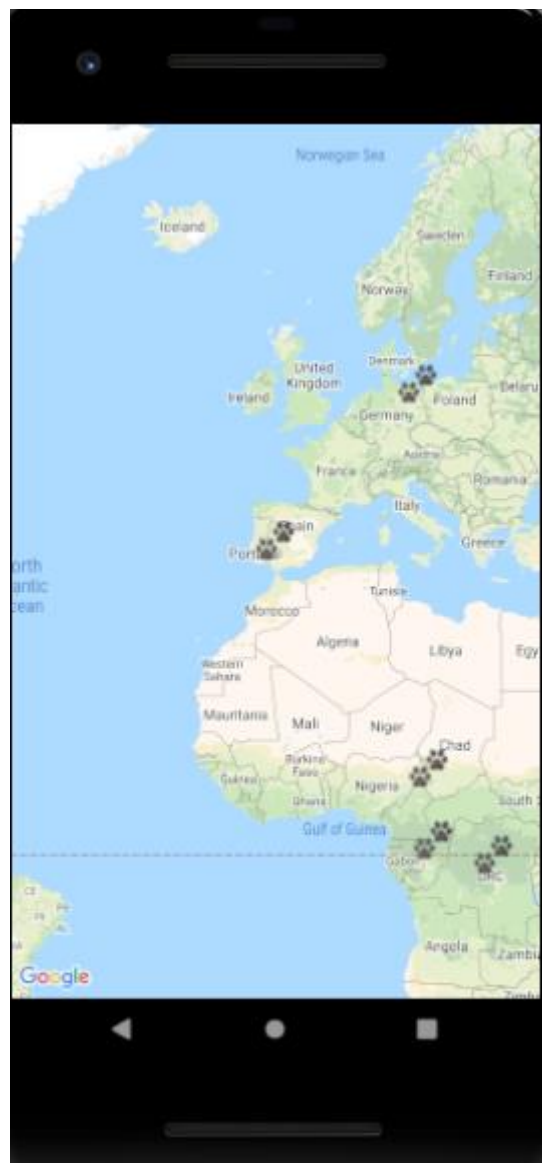
17 resultados

 <b>Directions API</b> Google Directions between multiple locations.	 <b>Distance Matrix API</b> Google Travel time and distance for multiple destinations.	 <b>Geocoding API</b> Google Convert between addresses and geographic coordinates.	 <b>Geolocation API</b> Google Location data from cell towers and WiFi nodes.
 <b>Maps Elevation API</b> Google Elevation data for any point in the world.	 <b>Maps Embed API</b> Google Make places easily discoverable with interactive Google Maps.	 <b>Maps JavaScript API</b> Google Maps for your website	 <b>Maps SDK for Android</b> Google Maps for your native Android app.

Una vez generadas todo vamos implementar nuestra clave y ya con esto es suficiente para cargar nuestro mapa, solo haría falta programar los botones para poder acceder a las siguientes especies.

```
-->  
<string name="google_maps_key" templateMergeStrategy="preserve" translatable="false">  
    AIzaSyB4jTI8XmXUSbJCaUJQFbdxw-jJk3Mw2nk</string>  
</resources>
```

Aquí vemos el resultado.



The screenshot displays the Android Studio environment. On the left, the XML code for an adapter is visible, defining views for different animal categories. The code includes:

- Toolbar**: A toolbar at the top with a principal button.
- ConstraintLayout**: The main container for the items, oriented vertically.
- ScrollView**: Wraps the entire list of items.
- LinearLayout**: Used for each item's header and footer text.
- ImageView**: Displays the animal image.
- TextView**: Displays the animal name.

On the right, the visual preview shows a yellow header bar followed by five cards, each representing an animal category with an image and a label:

- Mamíferos**: Image of a lion.
- Aves**: Image of a bald eagle.
- Anfibios**: Image of a green tree frog.
- Reptiles**: Image of a lizard.
- Peces**: Image of a goldfish.

```

public void entrarAnfibios (View v) {
    Intent anfibios = new Intent( packageContext this, AnfibiosActivity.class);
    startActivity(anfibios);
}

public void entrarReptiles (View v) {
    Intent reptiles = new Intent( packageContext this, ReptilesActivity.class);
    startActivity(reptiles);
}

public void entrarPeces (View v) {
    Intent peces = new Intent( packageContext this, PecesActivity.class);
    startActivity(peces);
}

@Override public boolean onKeyDown(int keyCode, KeyEvent event) {
    if (keyCode == KeyEvent.KEYCODE_BACK) {
        return true;
    }
    return super.onKeyDown(keyCode, event);
}

@Override
public boolean onNavigationItemSelected(@NonNull MenuItem menuItem) {

    switch (menuItem.getItemId()){
        case R.id.itemQuiz:
            Intent i = new Intent( packageContext this, PrincipalQuizActivity.class);
            startActivity(i);
            drawerLayout.closeDrawers();
            break;
        case R.id.itemDesarrolladores:
            Intent desarrolladores = new Intent( packageContext this, DesarrolladoresActivity.class);
            startActivity(desarrolladores);
            drawerLayout.closeDrawers();
            break;
        case R.id.itemSalir:
            AlertDialog.Builder dialogol = new AlertDialog.Builder( context this);
            dialogol.setTitle("Confirmacion salida");
            dialogol.setMessage("¿Desea salir de la aplicación?");
            dialogol.setCancelable(false);
            dialogol.setPositiveButton( text: "Cancelar", (dialogol, id) -> {
                dialogol.dismiss();
            });
            dialogol.setNegativeButton( text: "Confirmar", (dialogol, id) -> {
                finish();
                onDestroy();
            });
            dialogol.show();
            drawerLayout.closeDrawers();
            break;
    }
}

```

## 2.2. Planificación

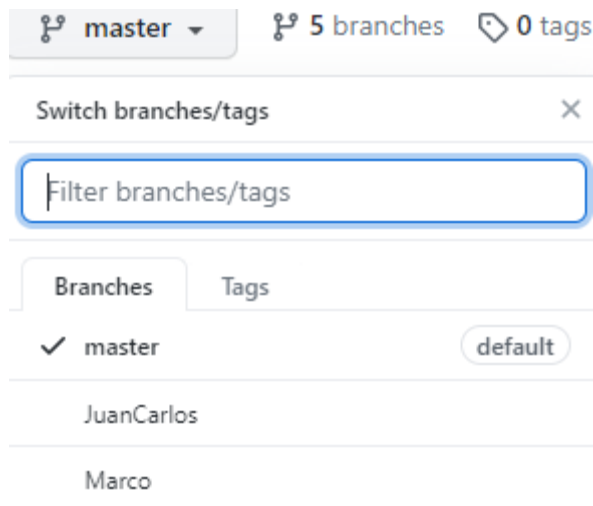
Este proyecto comenzó con Nico, siendo el ejecutor de la idea.

Al informarnos de que los tres íbamos a ser los componentes de realizar el proyecto, tuvimos una reunión por Skype.

En la cual concretamos nuevas ideas para implementar al proyecto, que como dije antes tenía Nico en la cabeza.

A parte de hacer reuniones cada semana para ver cómo íbamos en Skype, creamos un grupo de wasap, que para el día a día nos vino bastante bien. Para compartir el código utilizamos GitHub, una herramienta que nos ha facilitado mucho las cosas.

Nico tenía la rama Master, sobre esa rama nos creamos, las de Marcos y Juan Carlos.



A la hora de asignarnos las tareas, lo dejamos claro en la primera reunión, si que es cierto que, durante la semana, nos íbamos ayudando mutuamente, ya que, si había alguna duda sobre alguna parte del código, siempre teníamos el apoyo del resto de compañeros. Durante las semanas siempre íbamos teniendo una o dos reuniones a la semana en la cual nos íbamos enseñando el código y nuestros progresos.

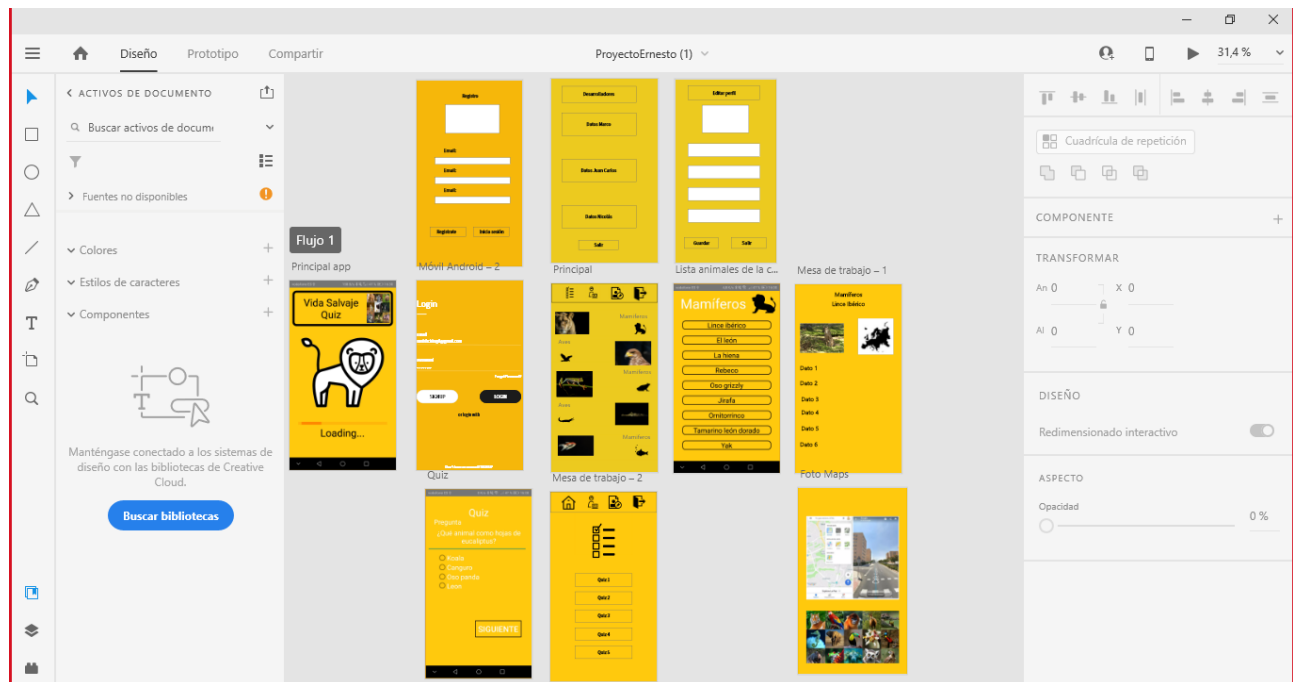
En general, creo que se ha formado un buen grupo, no hubo malos momentos y siempre hemos sabido luchar contra lo que teníamos delante.

En este apartado también decir que los profesores en las tutorías, siempre te iban guiando, sobre cosas que implementar, que quitar, que parte modificar, si en vez de utilizar una manera de ejecutar, se podría hacer de otra para que fuese más vistosa. Sobre todo en la parte visual de la aplicación, ya que ninguno de los tres somos buenos en la parte visual, pero creo que hemos hecho un buen trabajo.

### 2.3. Descripción del trabajo realizado

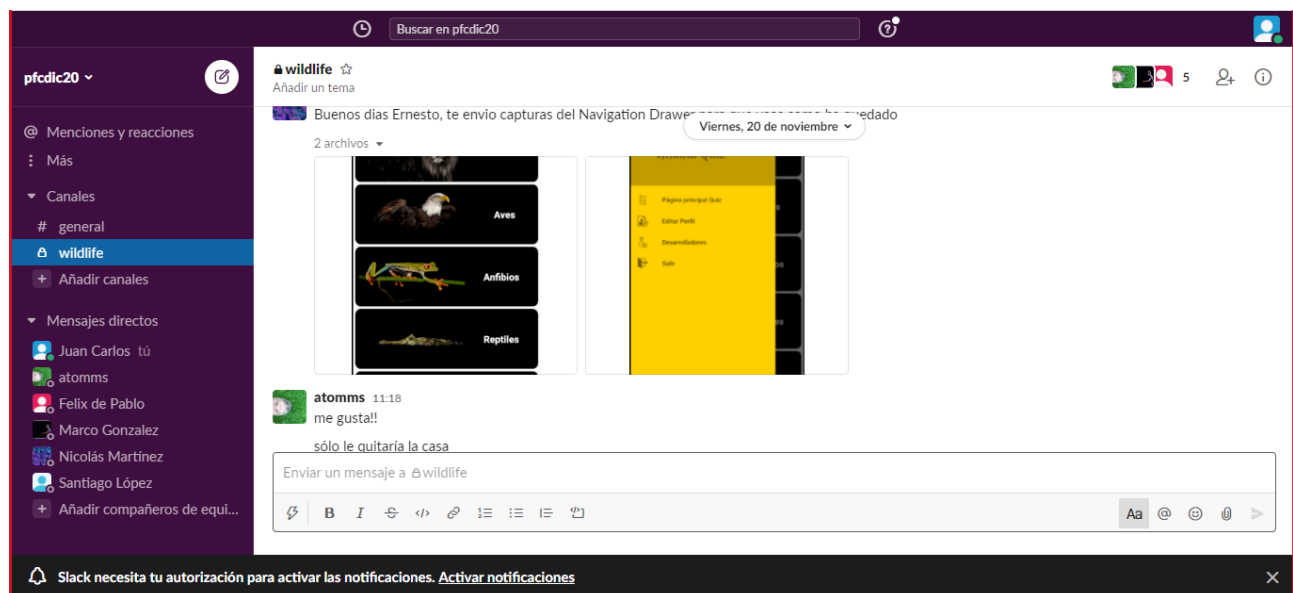
En primer lugar tuvimos una reunión para ver cómo íbamos a realizar las pantallas y el diseño, para esto hicimos un mockups, a continuación

mostraremos el diseño de las pantallas iniciales, siendo una base de lo que queríamos hacer, aquí mostramos el resultado.



**Figura 1: Vista Herramienta mockup adobe xd**

Con la herramienta Slack, teníamos acceso directo con los miembros del grupo y los profesores, para poner dudas, intercambiar ideas. En general, para poder comunicarnos.



**Figura 2: Vista Herramienta Slack**

A continuación, vamos a poner las partes del código que creemos, que son importantes de cara a la creación de la aplicación.

Añadimos este código porque es muy importante a la hora de listar los animales y acceder a los datos del animal seleccionado.

```
final DatosAnfibios datosAnfibios = new DatosAnfibios();

rv = findViewById(R.id.rvAnfibios);

rv.setHasFixedSize(true);

anA = new AnfibiosAdapter(datosAnfibios.getListaAnfibios());

llm = new LinearLayoutManager(context, this);
rv.setLayoutManager(llm);

anA.asignacionOnClickListener((v) -> {
    int i = rv.indexOfChild(v);

    //Obtenemos los datos del ave seleccionado.
    Anfibios an = datosAnfibios.getListaAnfibios().get(i);

    Intent intent = new Intent(getApplicationContext(), DatosAnfibiosCompletoActivity.class);
    intent.putExtra(CLAVE_ANFIBIOS, an);
    startActivity(intent);
});

rv.setAdapter(anA);
```

**Figura 3: Configuración del Recycler**

En este código que vamos a ver a continuación muestra el menú donde van aparecer las siguientes funciones.

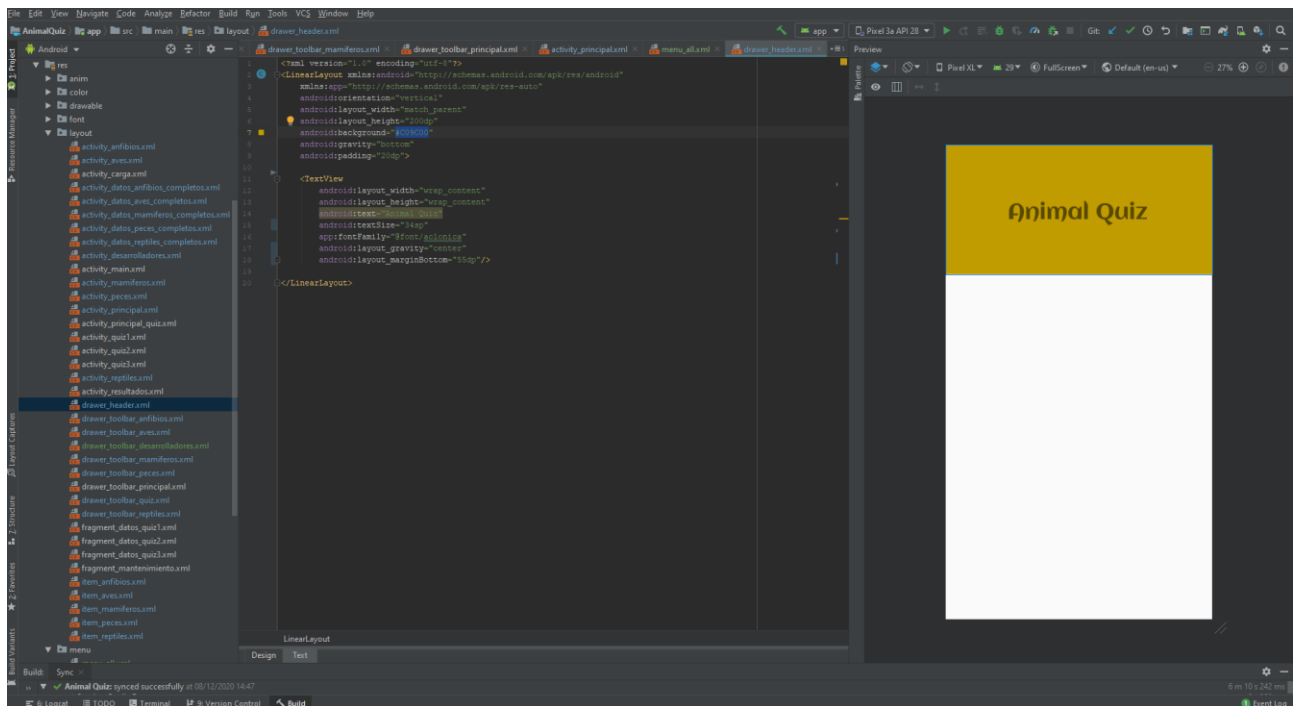


Figura 4: configuración cabecera del drawer layout

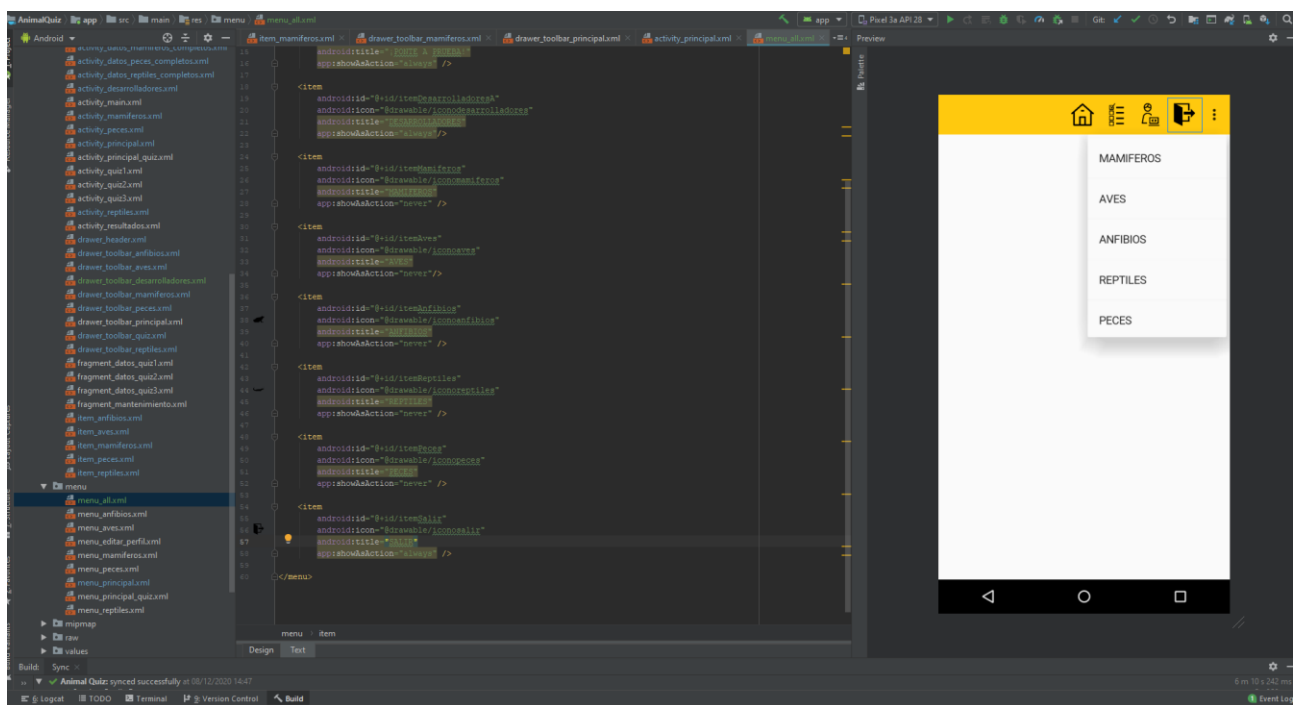


Figura 5: configuración del menú



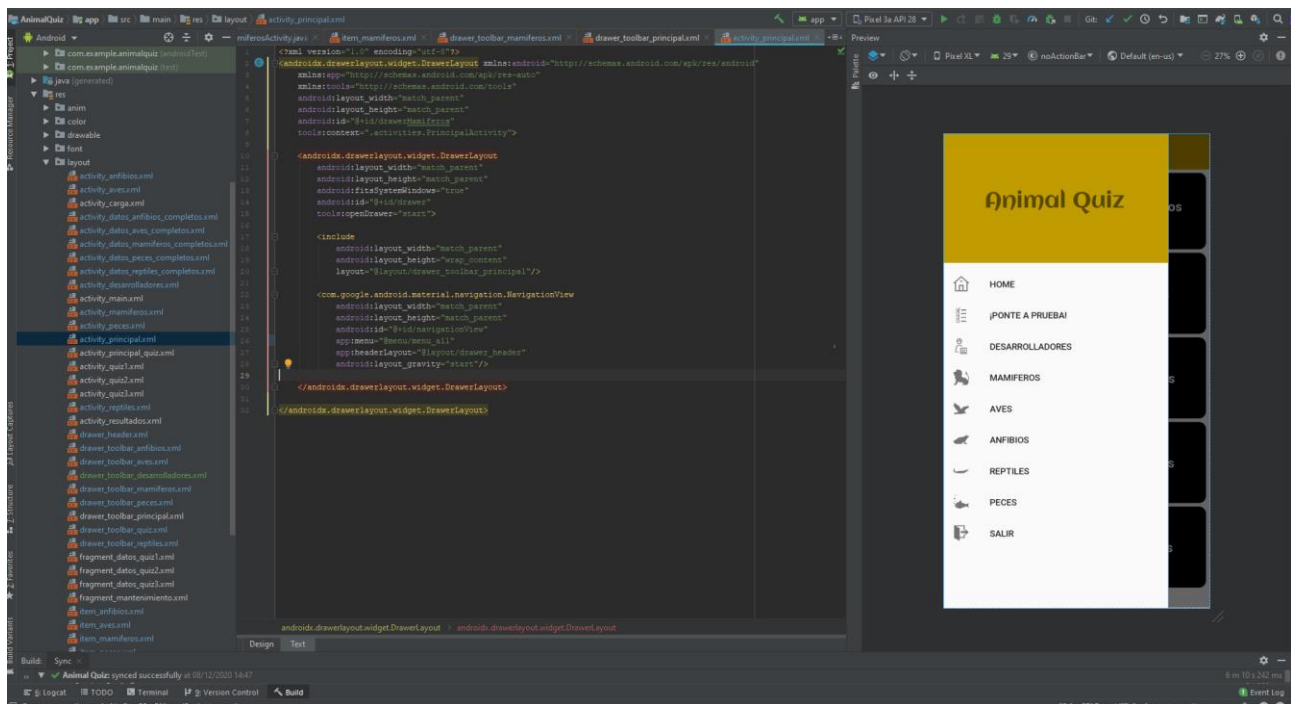


Figura 6: drawer layout completo con el menú dentro

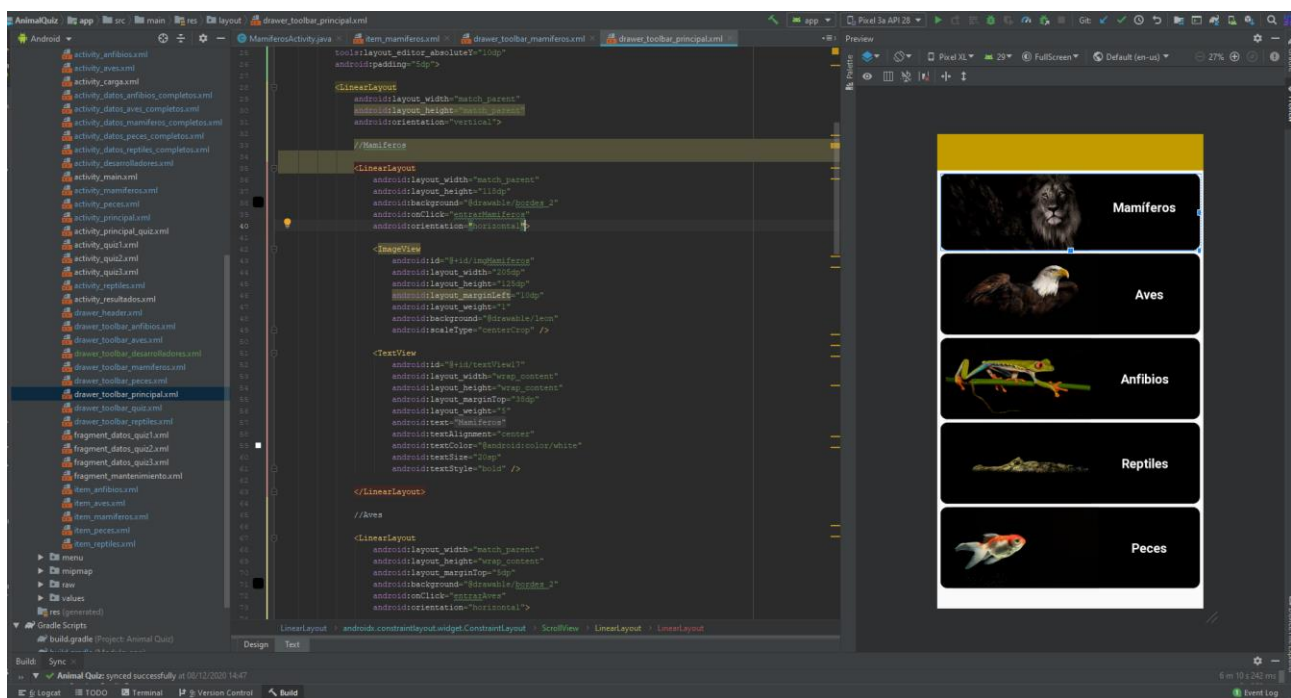


Figura 6: cerrado el drawer layout

The screenshot displays the Android Studio IDE with the 'drawer\_toolbar\_quiz.xml' file open. The XML code defines the layout of the toolbar, including a logo, three quiz buttons, and a title. The visual preview on the right shows the resulting UI design.

**XML Code (drawer\_toolbar\_quiz.xml):**

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:orientation="vertical" android:layout_width="match_parent"
    android:layout_height="match_parent">

    <androidx.appcompat.widget.Toolbar
        android:id="@+id/toolbar"
        android:layout_width="match_parent"
        android:layout_height="48dp"
        android:background="@color/white"
        android:background="?attr/actionBarColor"
        android:background="?attr/colorPrimary" />

    <androidx.constraintlayout.widget.ConstraintLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <ImageView
            android:id="@+id/imageView"
            android:layout_width="100dp"
            android:layout_height="100dp"
            android:layout_gravity="center"
            android:background="@drawable/logo_quiz"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintLeft_toLeftOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent"
            app:layout_constraintVertical_bias="0.15" />

        <TextView
            android:id="@+id/textView"
            android:layout_width="200dp"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:background="@drawable/state_button"
            android:onClick="onClick"
            android:text="Quiz 1"
            android:textAlignment="center"
            android:textColor="@android:color/black"
            android:textSize="18sp"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintLeft_toLeftOf="parent"
            app:layout_constraintRight_toRightOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent"
            app:layout_constraintVertical_bias="0.34" />

        <TextView
            android:id="@+id/textView2"
            android:layout_width="200dp
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:background="@drawable/state_button"
            android:onClick="onClick"
            android:text="Quiz 2"
            android:textAlignment="center"
            android:textColor="@android:color/black"
            android:textSize="18sp"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintLeft_toLeftOf="parent"
            app:layout_constraintRight_toRightOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent"
            app:layout_constraintVertical_bias="0.53" />

        <TextView
            android:id="@+id/textView3"
            android:layout_width="200dp
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:background="@drawable/state_button"
            android:onClick="onClick"
            android:text="Quiz 3"
            android:textAlignment="center"
            android:textColor="@android:color/black"
            android:textSize="18sp"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintLeft_toLeftOf="parent"
            app:layout_constraintRight_toRightOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent"
            app:layout_constraintVertical_bias="0.72" />

    </ConstraintLayout>

</LinearLayout>
```

**Visual Preview:** The preview shows a yellow toolbar with a logo on the left and three yellow buttons labeled 'Quiz 1', 'Quiz 2', and 'Quiz 3' on the right.

### Figura 7: Principal del Quiz

[illegible]

**Figura 8: Código del quiz uno, configuración**

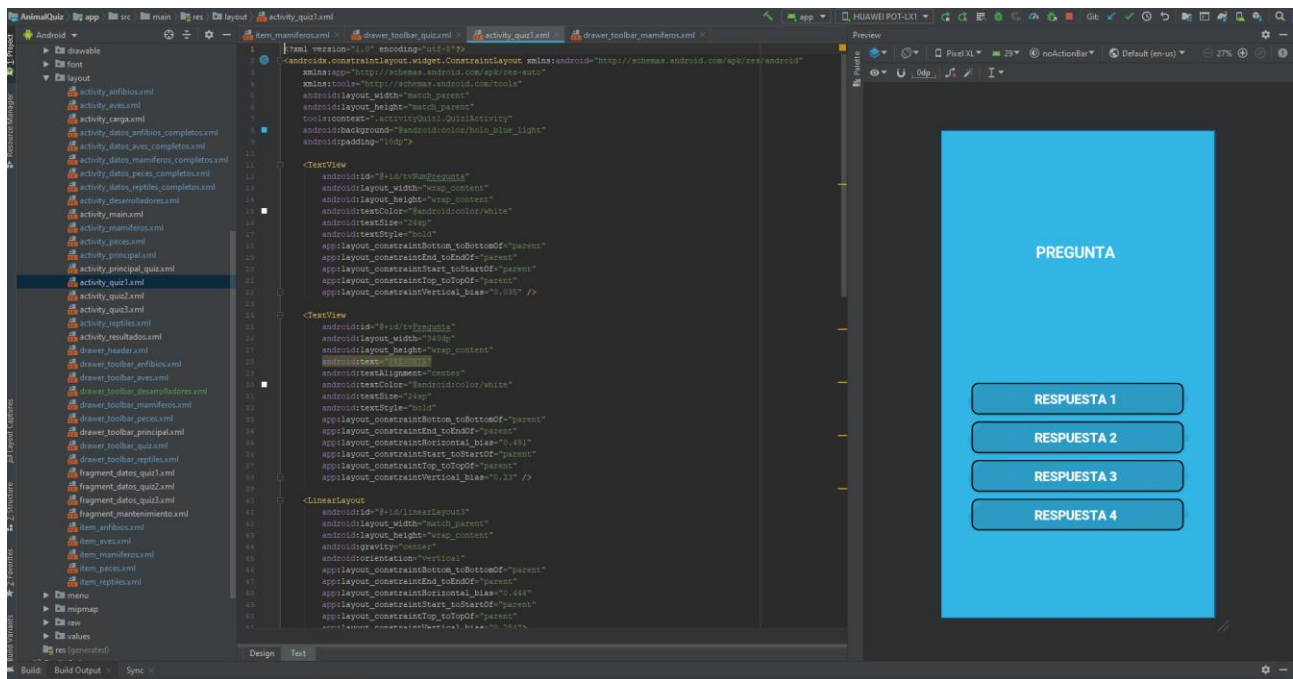


Figura 9: diseño gráfico del quiz

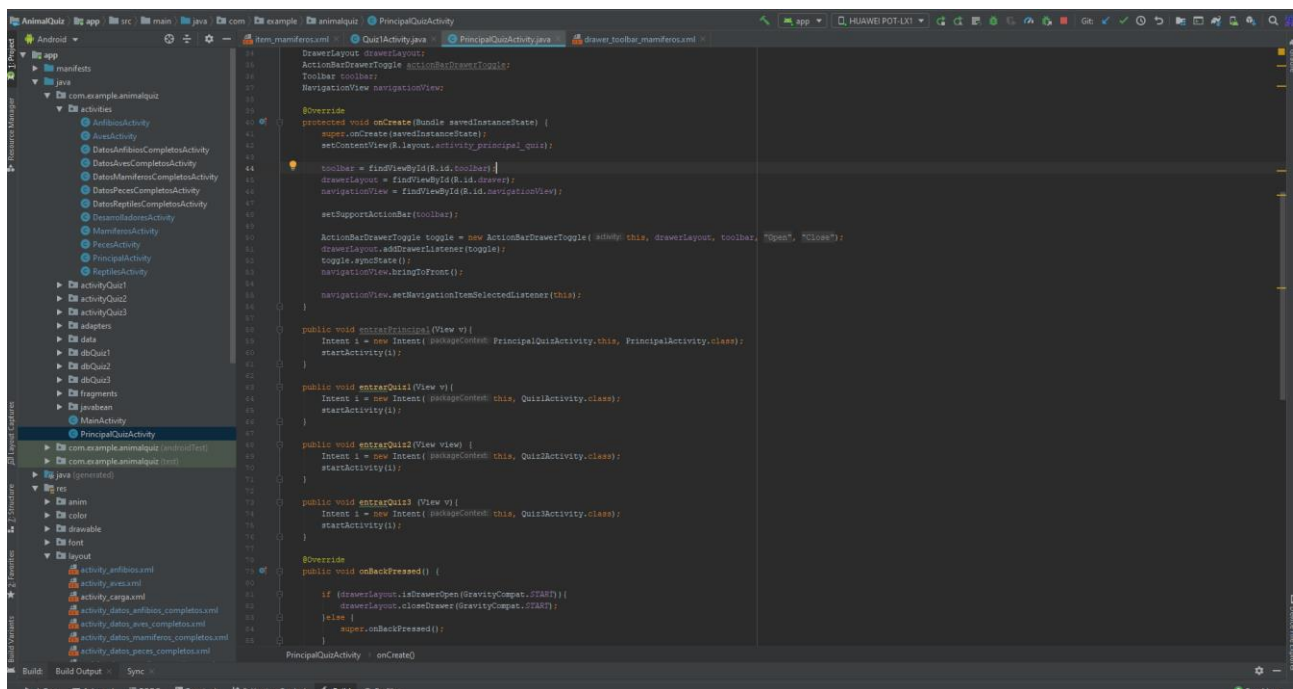


Figura 10: Código para cada uno de los quizzes

## 2.4. Resultados y validación

Dependiendo del procesador del móvil, tardará más o menos en ejecutar la aplicación.

Al iniciar la aplicación en el splash le pusimos que tardase 2 segundos.

En el caso de que el procesador no sea muy potente, la aplicación tarda (3-4 segundos)

```
V/ActivityThread: callActivityOnCreate
I/OverScrollerOptimization: start init SmartSlideOverScroller and get the overscroller config
    get the overscroller config
W/mpl.animalquiz: Accessing hidden method Landroid/graphics/FontFamily:-><init>()V (greylist, reflection, allowed)
    Accessing hidden method Landroid/graphics/FontFamily:~addFontFromAssetManager(Landroid/content/res/AssetManager;Ljava/lang/String;I)Z (greylist, reflection, allowed)
    Accessing hidden method Landroid/graphics/FontFamily:~addFontFromBuffer(Ljava/nio/ByteBuffer;I)Z (greylist, reflection, allowed)
    Accessing hidden method Landroid/graphics/FontFamily:~freeze()Z (greylist, reflection, allowed)
    Accessing hidden method Landroid/graphics/FontFamily:~abortCreation()V (greylist, reflection, allowed)
    Accessing hidden method Landroid/graphics/Typeface:~createFromFamiliesWithDefault(Landroid/graphics/FontFamily;Ljava/lang/String;I)Z (greylist, reflection, allowed)
I/AwareAppSchedulerManager: post cache drawable res id to aware, resId = 2131230816, packageName = com.example.animalquiz, cost time = 10159333
D/ActivityThread: add activity client record, r= ActivityRecord{38e5527 token=android.os.BinderProxy@1afdc [com.example.animalquiz/com.example.animalquiz.activities.PrincipalActivity]} token= android.os.BinderProxy@1afdc
D/HiTouch_PressGestureDetector: onAttached, package=com.example.animalquiz, windowType=1, mHiTouchRestricted=false
D/mali_winsys: EGLint new_window_surface(egl_winsys_display *, void *, EGLSurface, EGLConfig, egl_winsys_surface **, EGLBoolean) returns 0x3000
W/libEGL: EGLNativeWindowType 0x7d2305a450 disconnect failed
D/ActivityThread: Remove activity client record, r= ActivityRecord{d7843d9 token=android.os.BinderProxy@3802c78 [com.example.animalquiz/com.example.animalquiz.MainActivity]} token= android.os.BinderProxy@3802c78
D/AwareBitmapCacher: handleInit switch not opened pid=18657
W/Settings: Setting device_provisioned has moved from android.provider.Settings.Secure to android.provider.Settings.Global.
V/HiTouch_HiTouchSensor: User setup is finished.
V/AudioManager: querySoundEffectsEnabled...
W/ActivityThread: handleWindowVisibility: no activity for token android.os.BinderProxy@2ee71e
V/ActivityThread: callActivityOnCreate
D/ActivityThread: add activity client record, r= ActivityRecord{e4df494 token=android.os.BinderProxy@2ee71e [com.example.animalquiz/com.example.animalquiz.activities.MamiferosActivity]} token= android.os.BinderProxy@2ee71e
D/HiTouch_PressGestureDetector: onAttached, package=com.example.animalquiz, windowType=1, mHiTouchRestricted=false
D/mali_winsys: EGLint new_window_surface(egl_winsys_display *, void *, EGLSurface, EGLConfig, egl_winsys_surface **, EGLBoolean) returns 0x3000
I/RwViewRootImpl: removeInvalidNode all the node in jank list is out of time
W/libEGL: EGLNativeWindowType 0x7d2305c250 disconnect failed
W/Settings: Setting device_provisioned has moved from android.provider.Settings.Secure to android.provider.Settings.Global.
V/HiTouch_HiTouchSensor: User setup is finished.
V/AudioManager: querySoundEffectsEnabled...
W/ActivityThread: handleWindowVisibility: no activity for token android.os.BinderProxy@7a756af
V/ActivityThread: callActivityOnCreate
D/ActivityThread: add activity client record, r= ActivityRecord{ba4694e token=android.os.BinderProxy@7a756af [com.example.animalquiz/com.example.animalquiz.activities.DatosMamiferosCompletoActivity]} token= android.os.BinderProxy@7a756af
D/HiTouch_PressGestureDetector: onAttached, package=com.example.animalquiz, windowType=1, mHiTouchRestricted=false
D/mali_winsys: EGLint new_window_surface(egl_winsys_display *, void *, EGLSurface, EGLConfig, egl_winsys_surface **, EGLBoolean) returns 0x3000
I/RwViewRootImpl: removeInvalidNode all the node in jank list is out of time
W/libEGL: EGLNativeWindowType 0x7d2305a450 disconnect failed
W/Settings: Setting device_provisioned has moved from android.provider.Settings.Secure to android.provider.Settings.Global.
V/HiTouch_HiTouchSensor: User setup is finished.
W/HiTouch_PressGestureDetector: Touch pointer move a lot. The moving distance of X is:46.94429, limit is:60The moving distance of Y is:94.49451, limit is:60
W/HiTouch_PressGestureDetector: Touch pointer move a lot. The moving distance of X is:37.83696, limit is:60The moving distance of Y is:99.82532, limit is:60
```

### 3. CONCLUSIONES

Nuestra aplicación comienza con una pantalla de carga en la cual presenta el logo de nuestra aplicación y nuestro slogan en la parte superior, acompañado de una foto de animales, anticipando el contenido de la aplicación.

```
<ProgressBar
    android:id="@+id/progressBar"
    style="@style/Widget.AppCompat.ProgressBar.Horizontal"
    android:layout_width="300dp"
    android:layout_height="50dp"
    android:indeterminate="true"
    android:indeterminateTint="@android:color/holo_orange_dark"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.495"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/imageView4"
    app:layout_constraintVertical_bias="0.135" />
```

Posteriormente se accederá a otra pantalla la cual es un login y un registro, implementado con Firebase

```
android {
    compileSdkVersion 28
    buildToolsVersion "29.0.3"
    defaultConfig {
        applicationId "com.example.animalquiz"
        minSdkVersion 16
        targetSdkVersion 28
        versionCode 1
        versionName "1.0"
        testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
    }
    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'
        }
    }
}
```

En primer lugar y uno de los más importantes es tener muy en cuenta, que la versión que estás utilizando sea la adecuada para el funcionamiento de la aplicación.

```
classpath 'com.android.tools.build:gradle:3.5.2'
classpath 'com.google.gms:google-services:4.3.4'
```

En esta imagen apreciamos la versión que podemos encontrar y que tiene que ser acorde con la de la imagen anterior.

```
apply plugin: 'com.android.application'
apply plugin: 'com.google.gms.google-services'
```

Pasando a la siguiente pantalla de carga, entramos en el activity principal donde salen todas las categorías de mamíferos.

```
<ScrollView
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_marginTop="5dp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">
```

A continuación pondremos una parte del código de una de las especies.

```
//Mamiferos

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="118dp"
    android:orientation="horizontal">

    <ImageView
        android:id="@+id/imgMamiferos"
        android:layout_width="match_parent"
        android:layout_height="125dp"
        android:adjustViewBounds="true"
        android:background="@drawable/leon"
        android:keepScreenOn="true"
        android:onClick="entrarMamiferos"/>

    <TextView
        android:id="@+id/textView17"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/mamiferos"
        android:textColor="@android:color/white"
        android:textSize="20sp"
        android:textStyle="bold"
        android:layout_marginLeft="-125dp"
        android:layout_marginTop="20dp"/>

</LinearLayout>
```

Una vez accedemos a una de ellas aparecerá una amplia lista, en la cual se podrá seleccionar el animal que mas le guste al usuario.

Cuando selecciona unos de los animales disponibles, saldrá una breve descripción en la que se explica todo sobre el (su alimentación, sus mayores enemigos, etc...), aparte de eso también tendrá un apartado en el que podrá escuchar los sonidos de lo animales y para obtener aún más información sobre el animal, tendrás acceso en el mismo activity a un mapa que te devolverá su ubicación exacta.

Para que el usuario finalice su estancia en nuestra aplicación con la mayor satisfacción posible, en nuestro navigation drawer, tendrá total acceso a un test en el cual podrá plasmar todo lo aprendido con nuestra aplicación.

### 3.1. Innovación

En este apartado, nos gustaría hablar de uno de nuestros aspectos más novedosos y por los cuales creemos que nuestra aplicación puede llegar a generar un buen impacto en la sociedad y, sobre todo, en nuestro sector más joven.

Como principal novedad hemos implementado un apartado el cual, a través de un radar geolocalizador, podemos ubicar a la especie seleccionada en un habitat natural exacto y no conformarnos con una simple imagen o sonido de dicho animal, lo que hace que coja un encanto especial a la hora de atraer a ese público tan deseado y ansioso por aprender más sobre la fauna de nuestro planeta.

El equipo de "ANIMAL QUIZ" quiso hacer algo que saliese un poco de lo habitual ya que todas las aplicaciones competentes, disponen únicamente de un solo apartado.

**Con esto me refiero, a que las otras aplicaciones, solo presentan imágenes, otras únicamente te muestran el sonido que hacen los animales, y la gran mayoría no se preocupan en los que sus usuarios han aprendido utilizando esas aplicaciones.**

Gracias a todo esto, decidimos realizar una aplicación que complementen a todas ellas, en este caso es donde presentamos: "ANIMAL QUIZ", una aplicación la cual dispone de todo lo mencionado anteriormente en nuestras competencias (sin mencionar de nuevo la geolocalización que es lo que más nos diferencia del resto), pero aparte disponemos de un inicio de cuenta con un registro para darte el trato más personalizado posible, y a su vez nosotros poder entender los gustos y reclamos que necesitan nuestros usuarios para mejorar el rendimiento de nuestra aplicación.



### 3.2. Trabajo futuro

Uno de nuestros principales puntos y el cual nos refuerza mucho como aplicación es:

En primer lugar y como hemos comentado en varios de los puntos analizados anteriormente, es nuestra implementación de un mapa con geolocalización para ubicar a las especies que más nos interesen en su habitat natural.

Otro de los apartados que más queremos destaca, es la realización de nuestro Quiz, el cual tendrá una gran variedad de preguntas sencillas y divertidas para poder contestarlas de la mejor forma posible a la vez que pasa un buen rato.

Creemos que nuestra aplicación, dentro del ámbito educativo y natural, está bastante bien estructurada y completa, lo cual nos hace destacar un poco por encima de las demás.

Aparte también hicimos una previsualización de posibles actualizaciones e implementaciones para que nuestra aplicación vaya creciendo poco a poco. Una de nuestra idea para que esto suceda (Pensando de cara al futuro), seria implementar, por ejemplo, una serie de videos documentales en los cuales los usuarios puedan ver la vida total de los animales gratuitamente desde su perfil personalizado, en el cual podrá:

- Decorar su perfil con más le gusta (modificar colores, cambiar imagen de fondo de perfil, etc...)
- Almacenar los resultados de los test de una forma estructurada y organizada para que pueda encontrar el que más le gusto con un solo golpe de ojo.
- Guardas los videos documentales de los animales que más llamen la atención al usuario para poder visualizarlo las veces que dese.



#### 4. BIBLIOGRAFÍA Y WEBGRAFÍA

- <https://www.sgoliver.net/blog/firebase-android-base-datos-tiempo-real-1/>
- Para Firebase nos hemos ayudado en apuntes de clase y un proyecto que teníamos.
- <http://www.androidcurso.com/index.php/223>
- <https://www.sgoliver.net/blog/mapas-en-android-google-maps-android-api-1/>
- <https://www.ticarte.com/contenido/android-guion-para-crear-tu-propia-aplicacion-appquiz>
- <https://www.desarrollolibre.net/blog/android/usando-el-progressbar-en-android-para-indicar-procesos-de-carga#.X89RT9hKjIU>
- La mayoría de cosas que hemos utilizado son de otras Aplicaciones de otros proyectos realizados.
- Apuntes de Android studio durante el curso e interfaces gráficas.

También añadir que una gran parte de los conocimientos empleados en este proyecto, han sido los que hemos ido aprendiendo durante nuestro recorrido en el grado superior, de desarrollo de aplicaciones multiplataforma, los cuales, nos han servido, para resolver la mayoría de dudas, que se nos han generado a los tres miembros del equipo.

Ahora, vamos a indicar las asignaturas en las que más nos hemos apoyado para la realización de esta aplicación:

- Dispositivos móviles (Android studio)
- Base de datos
- Desarrollo de interfaces
- Programación

## 5. ANEXOS

En esta parte implementamos el API con la clave de acceso que genera Google Maps para poder recoger información acerca de la ubicación.

```
<resources>
  <!--
    TODO: Before you run your application, you need a Google Maps API key.

    To get one, follow this link, follow the directions and press "Create" at
    the end:

    https://console.developers.google.com/flows/enableapi?apiid=maps\_android\_backend
    &keyType=CLIENT\_SIDE\_ANDROID&r=B6:0C:C0:87:9D:45:97:23:52:66:67:1E:F2:D0:77:94:CD:0C:8B:C9%3Bcom.example.animalquiz

    You can also add your credentials to an existing key, using these values:

    Package name:
    B6:0C:C0:87:9D:45:97:23:52:66:67:1E:F2:D0:77:94:CD:0C:8B:C9

    SHA-1 certificate fingerprint:
    B6:0C:C0:87:9D:45:97:23:52:66:67:1E:F2:D0:77:94:CD:0C:8B:C9

    Alternatively, follow the directions here:
    https://developers.google.com/maps/documentation/android/start#get-key

    Once you have your key (it starts with "AIza"), replace the
    "google_maps_key"
    string in this file.
  -->
  <string name="google_maps_key" templateMergeStrategy="preserve"
  translatable="false">

      AIzaSyB4jTI8XmXUSbJCaUJQFbdxw-jJk3Mw2nk</string>
</resources>
```

En el Android Manifest implementamos algunos permisos que son importantes a la hora de usar el API, como por ejemplo el permiso de Internet para poder acceder al sitio web de donde recogerá la información del mapa para luego implementarlo en la app.

```
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.INTERNET" />

<meta-data
    android:name="com.google.android.geo.API_KEY"
    android:value="@string/google_maps_key" />
```



## Consideraciones importantes para la elaboración de la MEMORIA

Lo más importante de un trabajo es sin duda el trabajo en sí y sus resultados, sin embargo, un buen trabajo mal presentado arrojará dudas sobre lo allí expuesto y sobre la/s persona/s que han realizado el trabajo y la memoria. Por lo tanto resulta conveniente cuidar la presentación del mismo:

- Hay que evitar radicalmente las faltas de ortografía y gramaticales. Así como cuidar la formalidad del texto (no escribir como se habla y no escribir en primera persona "He considerado..." mejor "Se considera..." o "Hay o habría que considerar..."). Un trabajo presentado con faltas de ortografía, aunque sea bueno, dará una impresión lamentable.
- Es mejor utilizar un tipo de letra estándar y un tamaño también estándar. El tamaño de letra debe estar entre 10 y 12 pt., y la fuente puede ser Verdana, Arial o similar. Por lo general la letra Verdana de 12 pt. suele ser la más utilizada. Nunca hay que usar tipos raros o poco legibles y nunca tamaños inferiores a 10 pt. (10 pt. ya es realmente pequeño) ni superiores a 12 pt.
- Los listados de los programas se hacen con una fuente que tenga espaciado fijo (Courier por ejemplo). Estas fuentes suelen ser más grandes que las normales por lo que puede ser interesante cambiar también el tamaño de letra y dejarlo entre 8 y 10 pt.
- El interlineado debe ser simple y nunca menor de una línea. En algunos casos se puede agrandar algo pero sin sobrepasar la mitad del interlineado simple (1.5 líneas).
- En todos los trabajos resulta interesante incluir un índice con los capítulos, secciones y subsecciones que contenga, así como la página en la que se encuentra.
- La inclusión de imágenes ilustrativas y figuras -sin abusar- ayudan a la claridad y mejoran la presentación de un trabajo.

## Directrices particulares para el proyecto

- La presentación es muy importante precisamente porque hace brillar su contenido cuando es valioso.
- En la portada añadir el título del proyecto y los integrantes.
- No dejar hojas completas en blanco como separación, más allá de las que puedan aparecer en este documento.
- Abrir una nueva página al comenzar una nueva sección, salvo para evitar que un título quede suelto. Hacer un salto de página para empezar una nueva sección.
- La extensión del proyecto debe ser **entre 40 y 60 páginas** incluyendo todo el trabajo realizado, las conclusiones, etc., sin extenderse en detalles superfluos.

## Directrices particulares para los listados de programas

En los proyectos es bastante común la realización de memorias sobre la creación de algún programa, da igual el lenguaje, que haga alguna cosa. En estos casos las directrices dadas en los puntos anteriores son de obligada aplicación, aunque resulta conveniente resaltar los siguientes aspectos:

- El listado del programa debe estar comentado siempre. Si un programa no tiene comentarios casi es mejor no incluirlo.
- Un listado del programa, por muy bien comentado que esté, no es por sí solo la memoria del proyecto, de hecho debería ser una de las partes menos importantes.
- Si se incluye el listado completo del programa se pondrá en los anexos al final de la memoria. Sólo en el caso de que el programa sea muy pequeño (apenas una rutina) se puede poner en la parte de desarrollo o resultados.
- Si el listado es muy grande (más de 3 ó 4 páginas) es mejor no incluirlo o ponerlo aparte en un cd. También se puede intentar reducir el tamaño de letra, los espaciados, etc.
- En la parte de desarrollo o resultados se describe el funcionamiento del programa, por qué se ha hecho así, se explicará si funciona bien o no, bajo qué casos, etc. En estos apartados se pueden incluir trozos del listado del programa, pero sólo aquellas partes significativas para la explicación. Si el trozo que se incluye ocupa una página o más entonces no es aconsejable ponerlo.
- Los listados, tanto si son fragmentos como si es el listado global, se deben hacer con un tipo de letra de tamaño fijo (por ejemplo Consolas). Por otro lado, el tamaño debe ser lo menor posible para que ocupe poco espacio.
- Es recomendables insertar los listados de código en una tabla e insertarse un título (Referencias -> Insertar Título), si hay muchos listados puede ser interesante añadir un Índice de listados.

## Algunas reglas mecanográficas

Estas reglas son casi tan importantes como las ortográficas y su incumplimiento produce una mala impresión del trabajo que se está realizando. Se listan a continuación algunas de las más importantes o utilizadas:

- Los signos de puntuación como los puntos, comas, dos puntos, etc., van siempre unidos a la palabra que preceda; es decir, nunca hay que poner un espacio delante de cualquiera de estos signos.
- En cuanto a los signos que se abren y cierran como las comillas, los paréntesis, las llaves, etc., el signo que abre debe ir unido a la palabra que sigue, y el signo que cierra debe ir unido a la palabra precedente; es decir, detrás de un signo que abre nunca va un espacio al igual que delante de un símbolo de cierre donde tampoco se pone espacio.
- Los títulos de las secciones o subsecciones no deben ir sueltos. Si no hay espacio en la página para empezar una sección se debe empezar en página nueva.