

Technical Project Report - Android Module

# Tour Album

**Course:** Introdução à Computação Móvel

**Date:** Aveiro, 28<sup>th</sup> of November 2020 (28/11/2020)

**Authors:** 93127: Diogo Emanuel de Oliveira Moreira  
93406: Fábio Alexandre Andrade Carmelino

**Project abstract:** Our application is about every user having an **album** with photos of places they visited while having their journeys around the world (**tour**).  
The main achievements of our application is that you can have a gallery with all the photos you took while travelling and having a map to guide you while you are searching for new places to visit.

**Table of contents:**

- 1 Application Concept
- 2 Implemented solution
  - Architectural overview
  - Implemented interactions
  - Limitations
- 3 References and resources
  - Key project resources
  - Reference materials

# 1 Application Concept

In the making of our application we tried to approach it from the perspective of someone who loves to travel and wants an application to **save some memories** of their travels like some photos of the places they have visited.

Our application gives this type of user what they want by giving them a **map** so people can see where they are and **places close to them** while also giving them a **gallery** where they can see all the **pictures** they took while travelling.

There are no requirements to where you should use our application, whether it is while **travelling to some place near** you or while **travelling to another country**. We just want you to have an easier time saving photos of places you liked to visit.

## 2 Implemented solution

### Architectural overview

To save the users information such as his personal information and photos taken, we solely used **Firestore**.

When a user creates an account on the **Sign-up** fragment, we use Firestore's authentication to save his **email** and **password** and also save his **hashed email** (with md5) and **username** on RealTimeDB on the reference "/users", using the **email as the 'Primary key'**.

When a user Logs in, he is **validated** using authentication.

The **information of the user** is loaded from RealtimeDB's "/users/{hashed\_email}" path and the **profile picture** is loaded from the storage, in case it exists, on the path "/images/users/{username}/profilePic".

To **save large files** of information, in our case, the pictures that the user takes, we use Firestore's Storage and RealTimeDB together. When a photo is taken with our **built-in camera**, we **save the picture's information** (Description / Date / Location / Name) in the RealtimeDB "users{hashed\_email}/Album" path (where Album is an array of GalleryPic objects) and **save the image** itself on Storage's "/images/users/{username}/gallery/filename" path.

The pictures the user as taken are loaded by going through his Album saved on RealTimeDB. Knowing the name attribute of the each pic we just have to retrieve them one by one from his gallery folder created on Storage.

The **location** uses the phone's current location and it was implemented with the help of an application found on the web, tracking the user's current location and giving them a map provided by google maps.

To implement the **camera**, we followed a tutorial video about how to implement camera on android and then changed it a bit, so it works for our application correctly.

## Implemented interactions

### Login and Register

The image displays two mobile application screens for 'TourAlbum'.

**Left Screen (Login/Registration Options):**

- Header: TourAlbum
- Title: Tour Album
- Form fields:
  - Email
  - Password
- Buttons: LOGIN, REGISTER (highlighted with a blue box and arrow)

**Right Screen (Registration Form):**

- Header: TourAlbum
- Text: This application is for your own personal use as it is supposed to become like an album for your travels. Hope you like it!
- Form fields:
  - Email
  - Username
  - Name
  - Password
  - Confirm Password
- Button: REGISTER

### Profile

The image displays a mobile application screen titled 'fragment\_\_profile'.

**Profile Section:**

- Profile Picture: A placeholder image featuring a meme of a dog with the text 'I WANT TO BARK' and 'WOOF WOOF!!'.
- Button: CHANGE PROFILE PICTURE

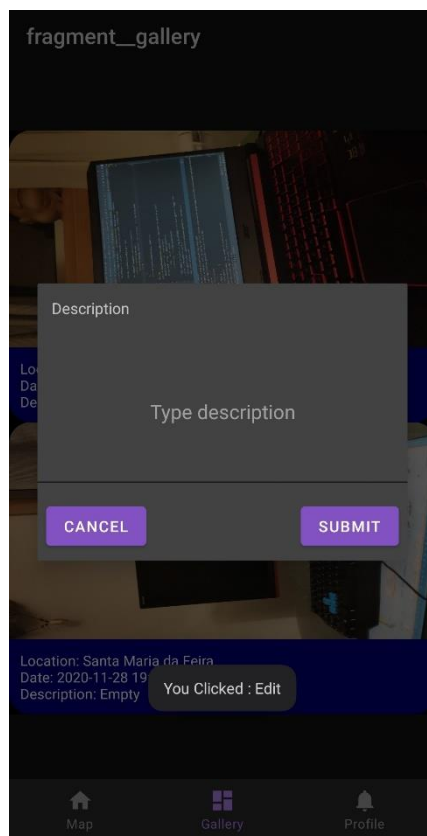
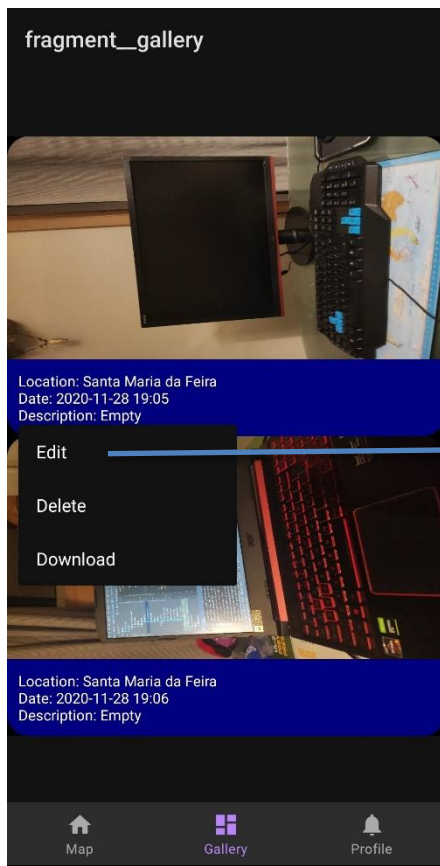
**User Information:**

- Username: didi
- Locations Visited: Loading..
- Achievements:

**Bottom Navigation Bar:**

- Map
- Gallery
- Profile

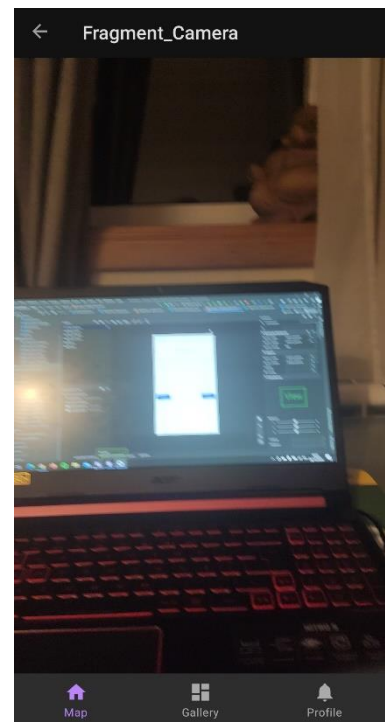
## Gallery with OnHold popup



## Map



## Camera



The **hardest problems** faced in this project were:

- **The implementation of the camera and GPS**

Both had too much deprecated material on the internet and unfinished implementations of other users making it hard to find good material about how to implement them.

- **Using Glide**

The use of Glide was an idea at the start but later on we found out that all the information about Glide on the internet was deprecated and, at the end, we decided to use Picasso.

- **Load images from Firebase to the gallery**

We were trying to use a method that returned a list with all the images retrieved from the database and then use the adapter to load them on the RecyclerView which was not working as expected making it way harder to implement since we were getting an empty list every time. Only after a long time of research we found out that we had to load each image to the RecyclerView with the adapter once at a time.

## **Limitations**

Unfortunately, some features we **wanted to see implemented could not make it** to this delivery either due of lack of time (due to some unforeseen problems arising) or due to the lack of material while searching for them.

Some of those features were not implemented such as:

- Points of interest marked in the map
- Have the option to transfer the photo
- See the photo in big picture
- Achievements and Number of different locations visited

And we have a bug while asking for the permission to use the map and camera for the first time where you need to open the map or camera again after accepting the permission to use it.

Gallery does not refresh after a change you need to refresh it manually.

## **Implementations after presentation**

After the presentation we fixed some minor bugs with the gallery and made some improvements such as **location** and **date working** correctly on images **in the gallery** and a **popup menu (on hold** on an image) making it possible to **edit the description** or **delete the image** (download is not yet implemented).

## 3 References and resources

### Key project resources

- Code repository: <https://github.com/diogoemoreira/ICMproject-TourAlbum>
- Ready-to-deploy APK: Inside git repository

### Reference materials

- <https://stackoverflow.com/>
- <https://firebase.google.com/docs/guides>
- [https://gist.github.com/milon87/9c02159da06ce4a7a8487b928edc4b82?fbclid=IwAR1aI1BLKrYIApcesYTUgEznM8eABgRyPhWvClaQ19xwEI6T8\\_uYclwNEQU](https://gist.github.com/milon87/9c02159da06ce4a7a8487b928edc4b82?fbclid=IwAR1aI1BLKrYIApcesYTUgEznM8eABgRyPhWvClaQ19xwEI6T8_uYclwNEQU)
- <https://javapapers.com/android/get-current-location-in-android/>
- [https://www.youtube.com/watch?v=8ZD6\\_SDsKWI&ab\\_channel=CleverSection](https://www.youtube.com/watch?v=8ZD6_SDsKWI&ab_channel=CleverSection)
- Some more like tutorialspoint which helped solve some other problems