



Computação Móvel

ParkBuddy - Flutter Project

Work made by:

Afonso Teixeira 93170

Manuel Couto 93285

PARKBUDDY 

1 Authors



Manuel Couto 93285



Afonso Teixeira 93170

You can find the repository of our work here <https://github.com/coutooo/ParkBuddy>

2 Motivation

Who has never scratched his head thinking where he did his parking?

The motivation for this project came through experiencing this problem ourselves. So basically the problem is when you park your car and then you can't seem to remember where you parked it. So we thought of an idea that we would like to use, in order to never forget where we parked it, since there's not a well known app trying to solve this issue. And the ones that exist, seem to have some sort of feature missing that we thought was crucial.

3 Project Description

This project consists of creating an app that allows the user to store the GPS location of several cars. So when you go out and park your car somewhere you can take a picture of the parking spot and also save the GPS location of your car. The app tries to resolve a problem that is quite common, but it can be used for different purposes, for example when you want to give your car location to a friend of yours. You can share the car location through scanning a simple QR Code.

4 Features

These are the features that are implemented in the app

- Biometric sensor to do authentication;
- History of past parkings;
- QR Code to share car location;
- QR Code Scanner;
- Camera to take photos of your parking;
- List of your parked cars;
- Removing cars from that list;
- Information about your parked car;
- Map so you can get an idea where you are and where you need to go;
- Markers that give your location and the location of your car;
- Walking Path between the user and the car;
- Proximity sensor that adjusts the brightness of the screen.

5 Architecture/Technical options

No blocs or providers were used in this project and instead of using firebase, neither we used MySQL or a similar type of database, we opted to use Hive as our database. Hive is a very easy to use and straightforward database, instead of having tables it creates a box and stores the information within it. A box is basically a dict, every value you insert in the box as a key associated with that value, so you can access each value by having the key to that value, in our case each key is the index within the box itself.

In our case we used the box to save information about each car in the list of parked cars or the list of history. That information was saved as a car class with the path to the photo taken by the user, name, latitude, longitude, street, address and license plate of each car.

We also used shared preferences which is very similar to Hive, shared_preferences is a Flutter plugin that allows you to save data in a key-value format so you can easily retrieve it later.

6 Overall assessment

All but one objective of ours were achieved. Apart from the pedometer we managed to implement every objective of ours and even add more features that were not thought of at the start of the project. We were able to use 5 technical features in total:

- Biometric sensor, for authentication purposes;
- Camera, to take pictures of your parked car;
- QR Code Scanner, to share the location of your car;
- GPS, to save the location of your car and to know where you are;
- Proximity sensor, to increase and decrease the brightness of the screen saving battery if you forget to lock the phone in your pocket.

We didn't manage to implement a pedometer. Our goal was to count the user's footsteps from the moment he left the car, so the user had an idea of the distance they were from the car. We also want to give out some stats, for the user to know how many steps they usually take once they park, but we had a problem with the plugins at our disposable and couldn't figure out a solution in order to implement it.

So, implementing this feature is an idea for future work.

7 Contribution assessment

The work was done by Afonso Teixeira and Manuel Couto, both did the same amount of work. Both gathered a lot of information and in order not to have irregularities, we had meetings where we grouped up and decided what we wanted to implement. Then, we separated the functions each one had to do, so no one would end up doing more work than the other. So the percentage of work each one has done is 50/50

Afonso's part:

- Profile page;
- Hive's part of the work;
- Biometric;
- Info page.

Manuel's part:

- QR pages;
- Map;
- GPS and path;
- Proximity/brightness;
- Park list page.

The UI was distributed evenly.

8 User Manual

The usage of the app is pretty intuitive and easy to use, nonetheless, we are going to do a manual for better understanding of how to perform each feature.

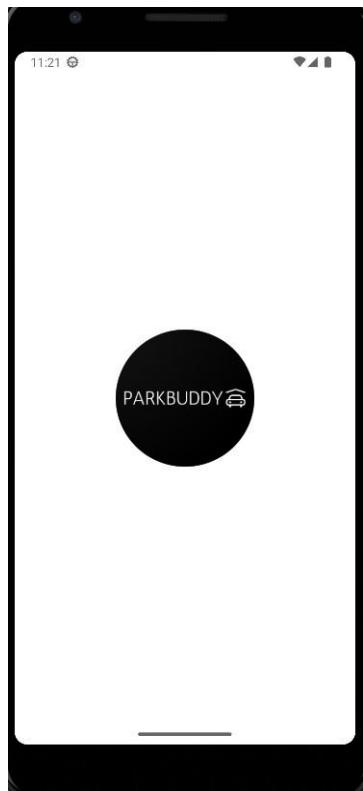


Fig1 - App launch.

We did a personalized launch screen and personalized icons/logos for our app.

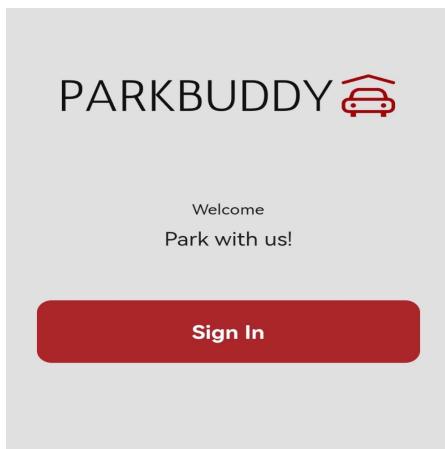


Fig2 - sign In

- This is the Login page, here you can authenticate your account by pressing sign in.
Once pressed the sign in , you will have to authenticate your device through the fingerprint (biometric sensor)
- This is the main page, where you can navigate to the several screens just by clicking the button of the screen you want to access

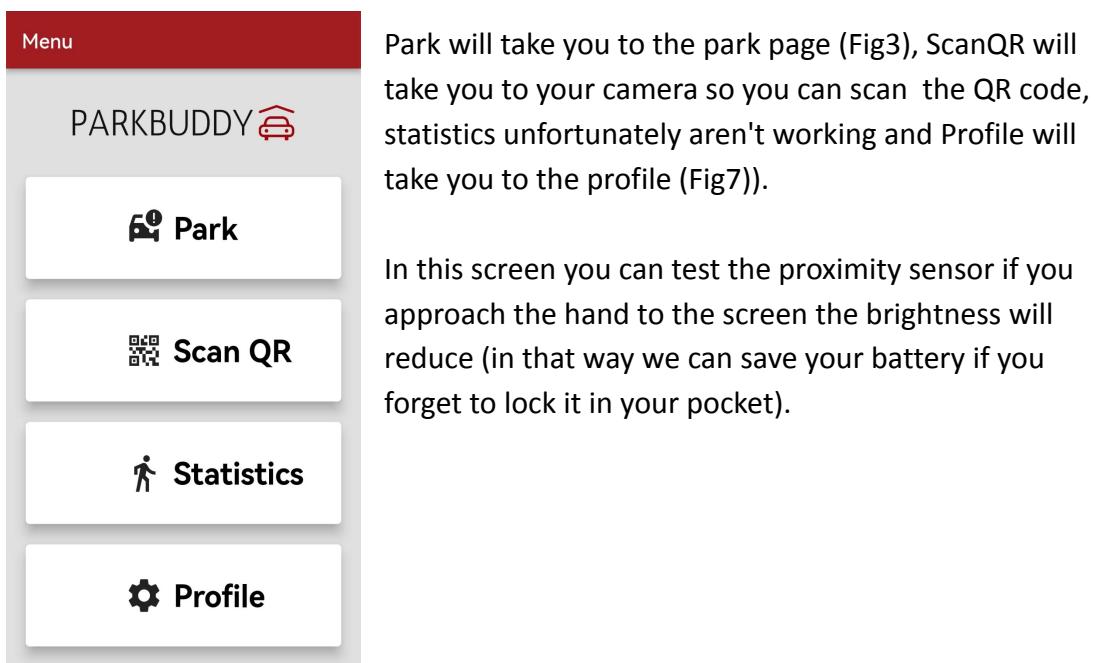


Fig2 - Main page

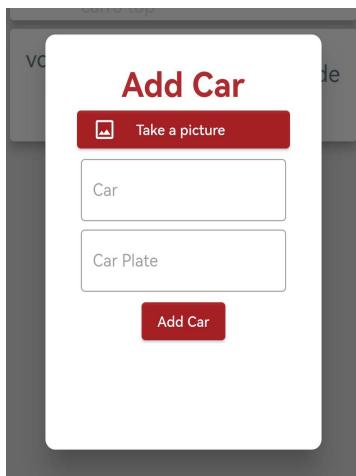
- This is the park page, where you can add parked cars to your list, by clicking the “+” button on the bottom right corner and remove from the list by simply sliding the car you want to remove.



You can go to the screen of each parked car just by pressing the card of the car you want to open.

Fig3 - Park page

- This is the popup you get when you press the “+” button on the Park page, here you can add a car to your list.



Your localization is saved automatically, you just need to provide the car name and the car plate that you want to associate and of course take a picture of the local.

Fig 4- Add car

- This is the info page of each parked car. It contains the photo you took, you can share the QR code so other people can know where your car is, you can check the map and even get more info about your parked car.

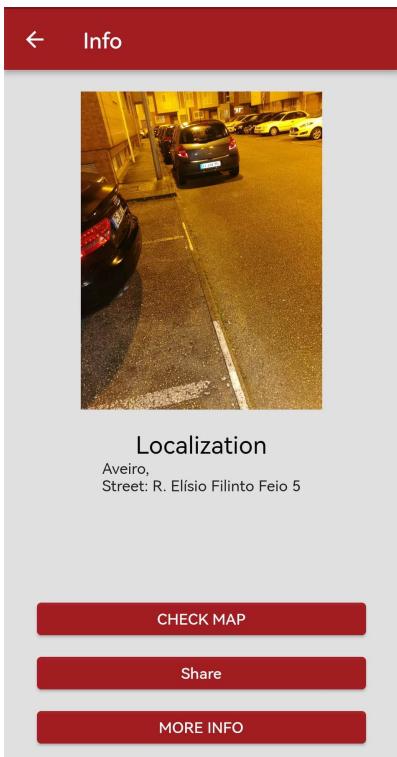


Fig 5- Info page

- This is the map where you have two markers the red one being the location of your car and the blue one being your location, when you press the get path you will get the walking path between the two markers, if you press the other button which have a icon of a marker it will change the map camera between your position and the car position to make it easier to find.

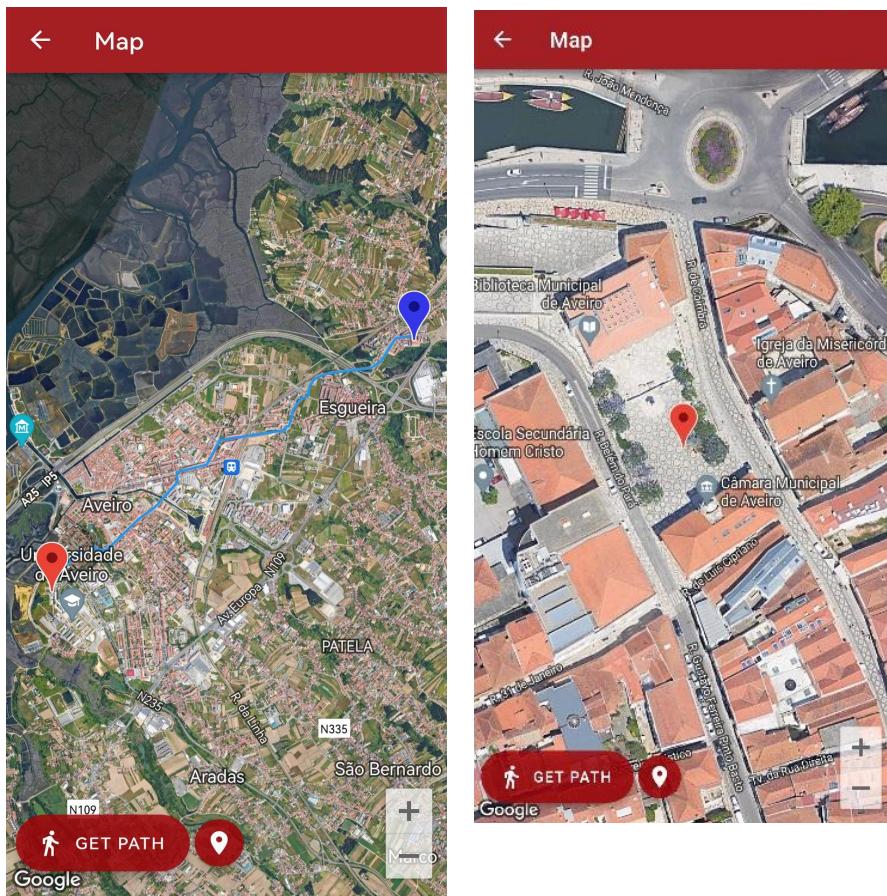


Fig 6 and 7 - Map

- This is the profile page, where you have a historial of your last 4 parked cars.

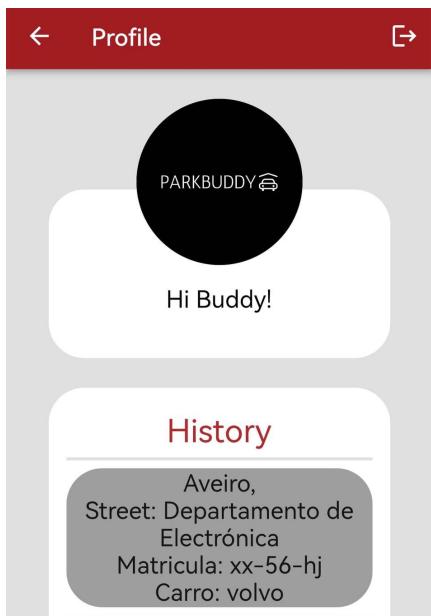


Fig 8 - Profile page

9 Bibliography

<https://pub.dev/>