Flutter Project: TrackMyRun

Bernardo Pinto – 105926 (50%)
João Santos – 11055 (50%)

Sumário

Introduction

Overview of the work

The TrackMyRun app is an ideal tool for monitoring runs, tracking the route, distance covered, allowing association of photos with the runs, and moreover, it offers an integrated feature that enables checking the weather conditions at the location where the run is planned to start, all without needing to exit the application.

Product specification

Functional scope and supported interactions

The TrackMyRun app caters to runners of all levels who want to track and improve their running performance. It allows users to record and monitor their running activities, including route tracking, distance covered, and associated photos. The main actors using the app are runners looking to track their progress and performance.

The primary use case scenario involves a runner using the TrackMyRun app to plan, execute, and analyze their running sessions. Upon opening the app, the user can, before starting a run, access the "weather" tab and check the weather conditions in their location, and then return to the main page and select the option to start a new run.

During the run, the user can directly track real-time information from the main page of their run, such as the distance already covered and pace. Additionally, they don't need to worry if they forget to finish the run, as TrackMyRun features a system that detects when the user is stationary and prompts them to either continue or finish their run.

After completing the run, the user can associate a title of their choice with the run and take a photo and associate it with that run to make the experience more memorable. Then, in the run history, the user can view a summary of their performance, including total distance and the start and end locations. They can also review the route taken on a map and view the associated photo.

Overall, the TrackMyRun app serves as a comprehensive tool for runners to track, analyze, and improve their running activities, enhancing their overall running experience.

1.1 Features

Tracking - Allows users to start a running session and record their activity in real-time. This includes continuous tracking of the user's location, pace, distance covered, and other relevant data throughout the entire run.

Map - During the run, users can view their route on an interactive map directly within the application. This includes a visual representation of the path taken and a route line that shows the path covered up to that moment.

Capture and associate images - Enables users to capture and associate a photo at the end of their run, enhancing the overall running experience.

History - Allows users to revisit and analyze their previous running sessions. It provides a comprehensive summary of the user's performance, including details such as distance covered, time, start and end locations displayed on the map, along with the associated photo. This functionality enables runners to track their progress over time and be inspired by their past achievements.

Weather - Provides users with information about current weather conditions in their running area. This information helps runners plan their runs more safely and effectively by considering real-time weather conditions.

External services

Google Maps API – The Google Maps API is a library that allows to use interactive maps from google maps. This API offers a lot of powerful features to create personalized maps experiences for their users such as different markers and icons, map drawing, actual location, geocoding and reverse geocoding, routes and directions and integration with flutter widgets.

Open Weather Map API – The Open Weather Map API is a service that gives access to meteorologic information in real time to basically anywhere in the world. You can integrate it with maps, and it has a lot of documentation.

Geolocation Service – Geolocation Service is a service that allows you to get information about your geographic location such as longitude latitude and altitude.

1.2 Architecture

We have chosen an architectural approach that promotes maximum flexibility and dynamism in our application. To achieve this, we opted to develop a tab-based interface, with each tab representing a key functionality of the application. The first tab allows the user to start their run, while the second tab displays the history of past runs. The third tab provides relevant weather information to the user, and the last tab is reserved for application settings.

For the implementation of these functionalities, we relied on the powerful widgets provided by Flutter, which enabled us to create an intuitive and responsive user interface.

Furthermore, we integrated several external APIs to ensure the operation of the main features of our application. These APIs provide real-time data, such as weather and location information, enriching the user experience.

Regarding data persistence, we opted to use SQLite. This choice provides us with a robust and efficient solution for storing and retrieving data locally, ensuring a consistent experience even in conditions of limited or absent connectivity.

Environment

SDK - ">=3.0.0 < 4.0.0"

Flutter - ">=3.10.0"

References and resources

Project resources

Resource:	URL/location:
Git repository	https://github.com/JotaCLS/ICM-Flutter-Project
Video demo	https://github.com/JotaCLS/ICM-Flutter-Project/blob/main/demo/TrackMyRunDemo.mp4

Reference materials

- https://flutter.dev/
- https://pub.dev/
- https://pub.dev/packages
- https://docs.flutter.dev/cookbook/persistence/sqlite