

Health Tracker

Subject: Introdução à Computação Móvel

Date: Aveiro, 19/06/2024

Students: 108712: Diogo Falcão
108840: José Gameiro

Project abstract: This report presents the development of a comprehensive health application designed to enhance users' well-being through two integrated platforms: a mobile app and a WearOS app. The mobile application offers extensive health monitoring features, allowing users to track various health metrics and manage their overall health more effectively. The WearOS counterpart complements this by enabling real-time monitoring of health values during physical activities, providing users with immediate feedback and insights while exercising.

Report contents:

[1 Application concept](#)

[2 Implemented solution](#)

[Architecture overview \(technical design\)](#)

[Implemented interactions](#)

[Project Limitations](#)

[New features & changes after the project presentation](#)

[3 Conclusions and supporting resources](#)

[Lessons learned](#)

[Work distribution within the team](#)

[Project resources](#)

[Reference materials](#)

1 Application concept

→ what is this app for? Who are the target users? How would they benefit from using the app in their journeys?

HealthTracker is a health companion app both for Android and WearOS that measures the health data through the HealthConnect API. There are 5 screens in the phone app:

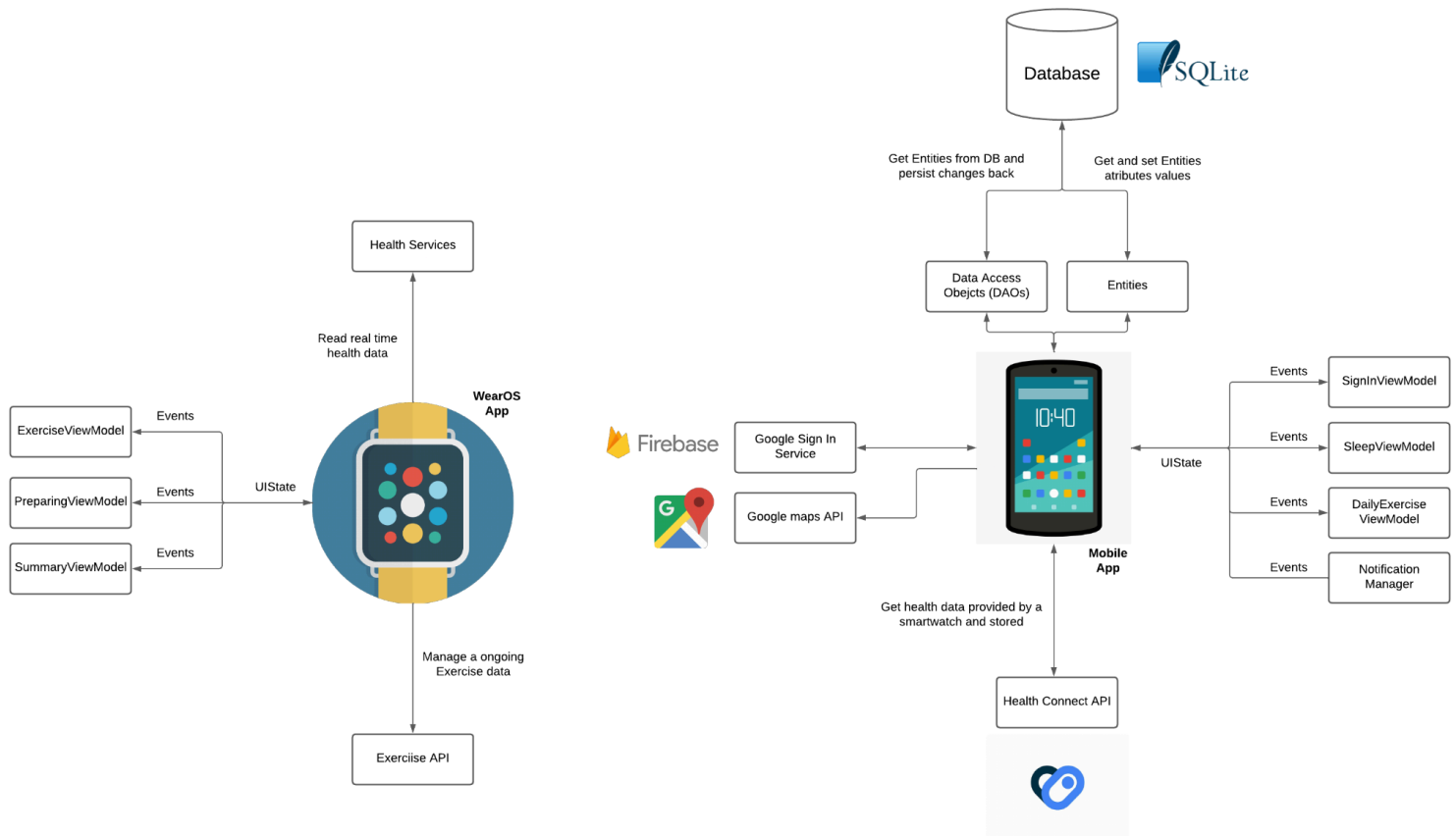
- SignInScreen is a simple sign in with google login to our app.
- HomeScreen is constituted by a ring containing steps made in a day, other useful information such as calories and kilometers, and a map containing POIs for exercise in Aveiro.
- PlannedExercises is a list type screen where we have divided each day of the week into lists for the user to write the exercises we want to do. When a user starts an exercise, he can move to the wearOS app to do that exercise. Later, when he completes the exercise, he checks the list, later receiving a notification about his work.
- NotificationsScreen is the app's notifications control center. Here the user can see and dismiss all the app notifications.
- SleepScreen is a screen with sleep info such as a sleep graph containing all the phases of the sleep and the duration of the sleep.
- Finally there is a setting screen where the user can change its steps goals. When that change is performed, the user can automatically see the change in the ring on the HomeScreen.

In the WearOS app there are 4 screens:

- Welcome screen with a sweet animation welcoming the user.
- A wait by screen waiting for the user to start the exercise.
- An exercise screen containing the time elapsed, the heartbeats read directly using the HealthServices, kms and laps.
- Finally, there is a workout complete screen containing a summary of the activities performed.

2 Implemented solution

Architecture overview (technical design)



Smartwatch

- **Wear OS App:** This is the application running on the smartwatch. It gathers data from various sources and displays it on the watch face.
- **Health Services:** This block retrieves health data in real-time from the user.
- **Exercise API:** Manage an exercise started by the user, by reading health values such as heart rate, total distance covered, number of calories burned and number of laps.

Mobile App

- **SigninViewModel:** This view model is likely responsible for user sign-in and authentication.
- **ExerciseViewModel** and **SummaryViewModel** : These view models handle data related to exercise and user summaries.
- **Firebase Google Sign In Service:** This service allows users to sign in to the

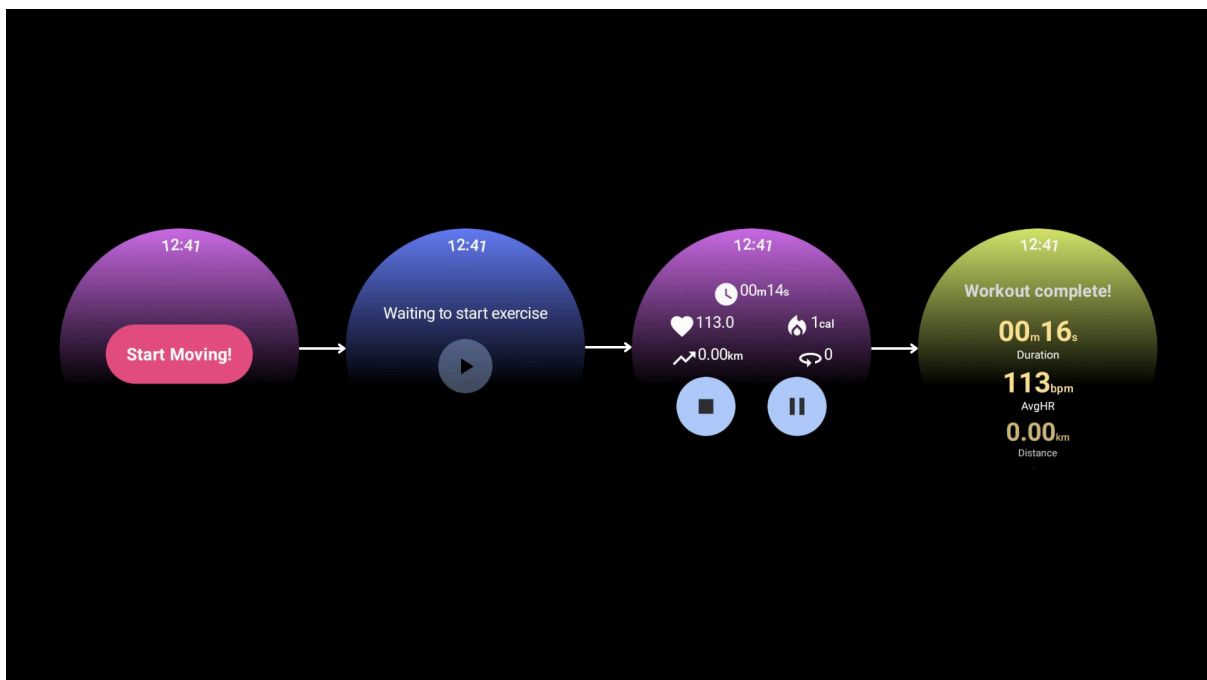
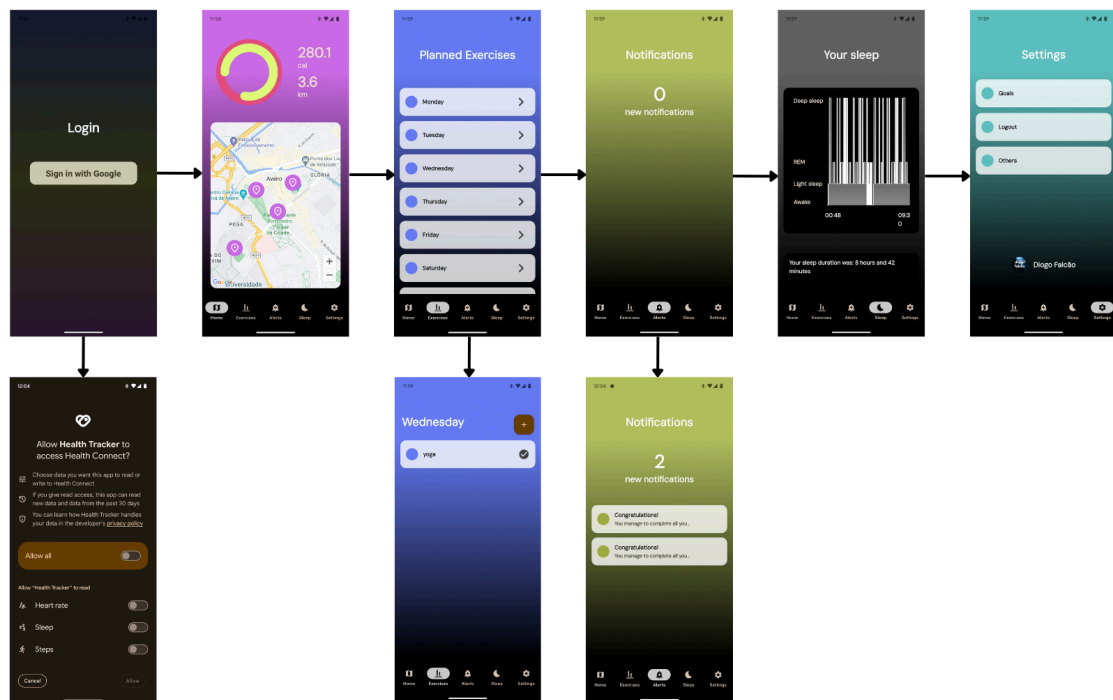
application using their Google account.

- **Google Maps API:** This integration might be used to display maps for location-based activities.
- **Health Connect API:** This API facilitates communication with the Health platform, which provides a unified interface to access health data from various apps.
- **Notification Manager:** This component is responsible for managing notifications on the mobile device.
- **Data Access Objects (DAOs):** These are responsible for getting and setting values of the data attributes.
- **Database (SQLite):** This stores the health data retrieved from the Health Services, User and stepsGoals.

Communication

- The mobile app retrieves data from the Health Connect API and displays it on the user interface.
- WearOS app stores values into the Google Health Connect API using the Google Health Services.
- Notifications are managed by the Notification Manager.

Implemented interactions



As a user, José wants to be able to track his sleep so that we keeps an eye of his sleep routine.

As a user, Diogo wants to see Aveiro's POIs to exercise so that we can track the calories burned later.

As a user, José wants to track his runs on the app so that we can later check the exercise as done on the app.

Project Limitations

Notifications on the watch and some basic unitary tests.

New features & changes after the project presentation

-

3 Conclusions and supporting resources

Lessons learned

Challenges in integrating real-time health data and managing synchronization between mobile and WearOS platforms. Kotlin is quite complex and treating data into the UI is not as easy as it seems.

Work distribution within the team

Taking into consideration the overall development of the project, the contribution of each team member was distributed as follow: José Gameiro did 50% of the work, and Diogo Falcão contributed with 50%.

Project resources

Resource:	Available at:
Code repository:	https://github.com/falcaodiogo/Projeto2-ICM
Ready-to-deploy APK:	https://github.com/falcaodiogo/Projeto2-ICM/tree/main/APK/Phone %20App
App Store page:	-
Demo video:	https://uapt33090-my.sharepoint.com/:f:/g/personal/falcao_diogo_ua_pt/EmnASKVV4oNIhq8FJ2vvrGcBM8a-y6ekquzFXSiyYuHLvQ?e=Fdeev

Reference materials

<https://developer.android.com/jetpack/androidx/releases/health-connect>

<https://developer.android.com/health-and-fitness/guides/health-services>

<https://m3.material.io/styles/motion/overview>

<https://meetpatadia9.medium.com/local-notification-in-android-with-jetpack-compose-437b430710f3>

