

# Application Development for Mobile and Ubiquitous Computing

## **Erasmus Web**

Maria Sofia Nascimento

Mariana Aires



# Idea

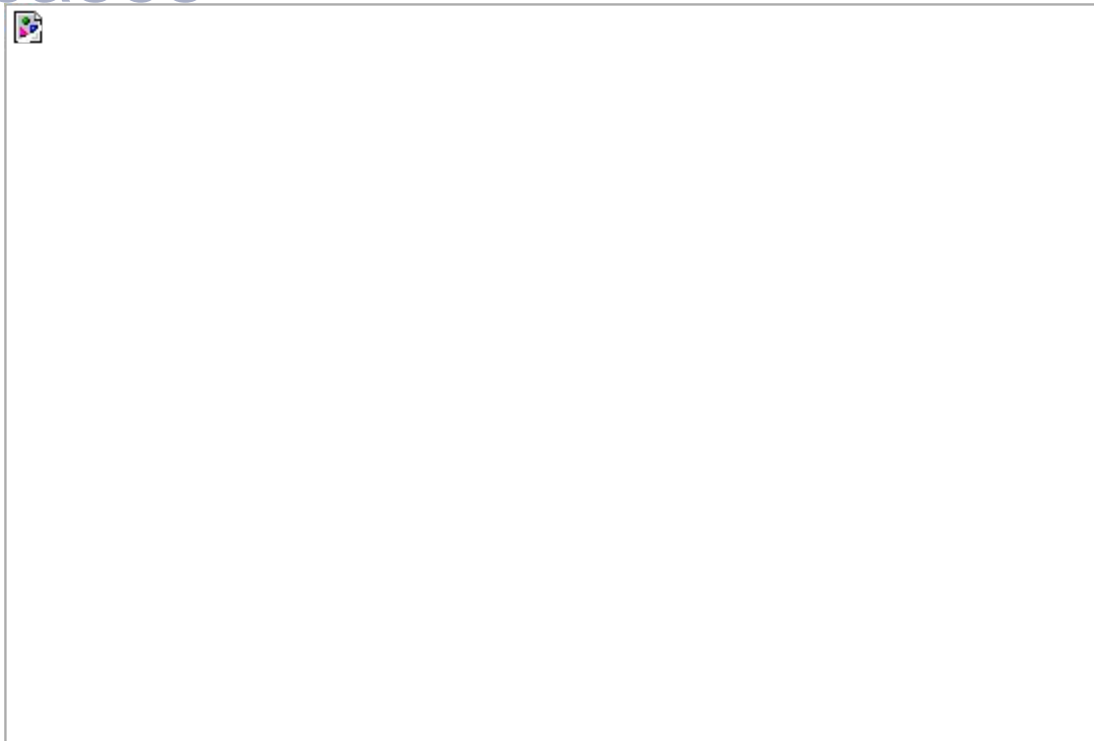
- Location-based event planner;
  - Centralization of events happening in an area
- Matching users with **fellow students** in the same situation;
  - **Stimulate connections** between users via interactions in the application.



erasmus Web



# Use cases

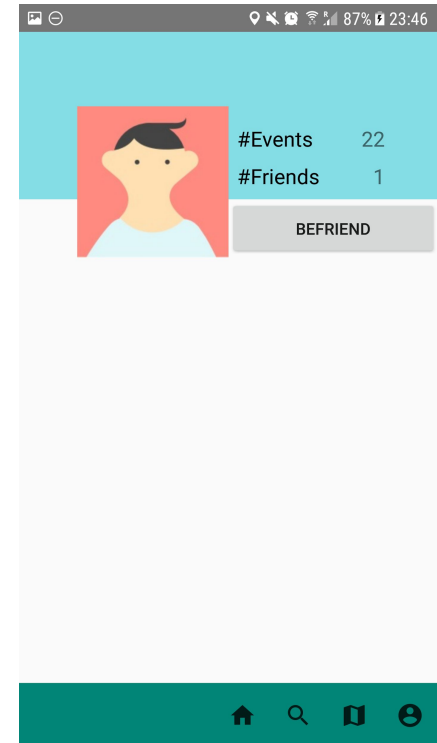
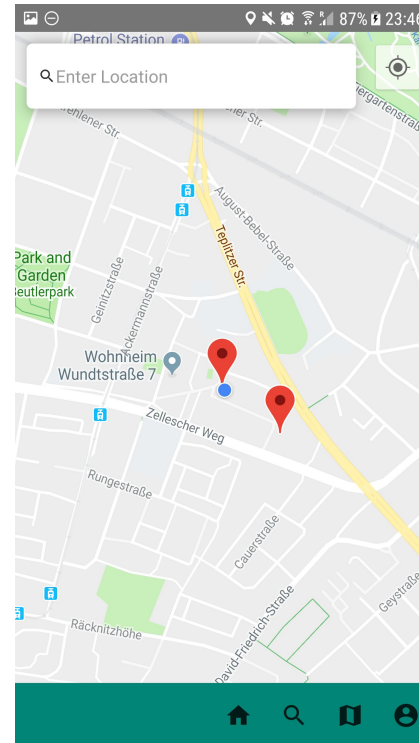
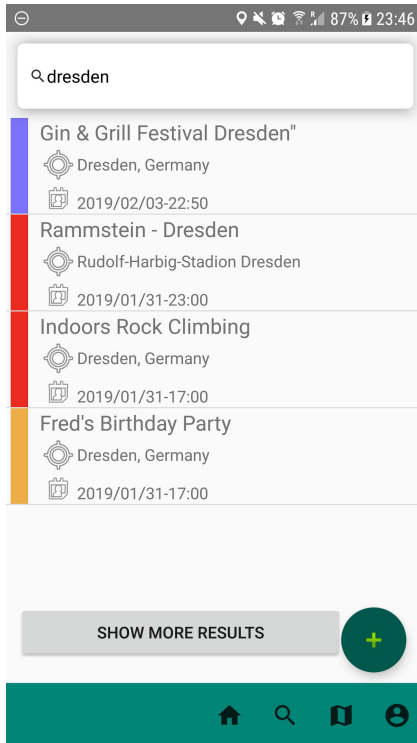


# The app



SIGN IN

REGISTER



# Important feature

- Transmit relevant and opportune information (via notifications);

## How?

- Detect user location with GPS;

```
LocationManager.GPS_PROVIDER && LocationManager.NETWORK_PROVIDER &&  
JobService
```

- Receive updates with events happening nearby;

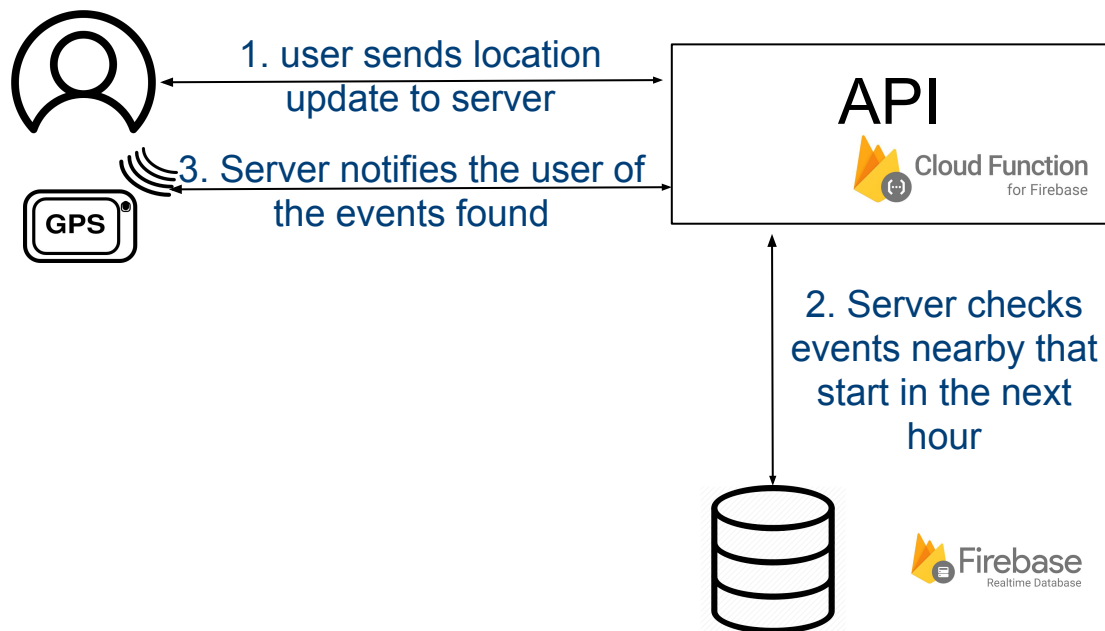
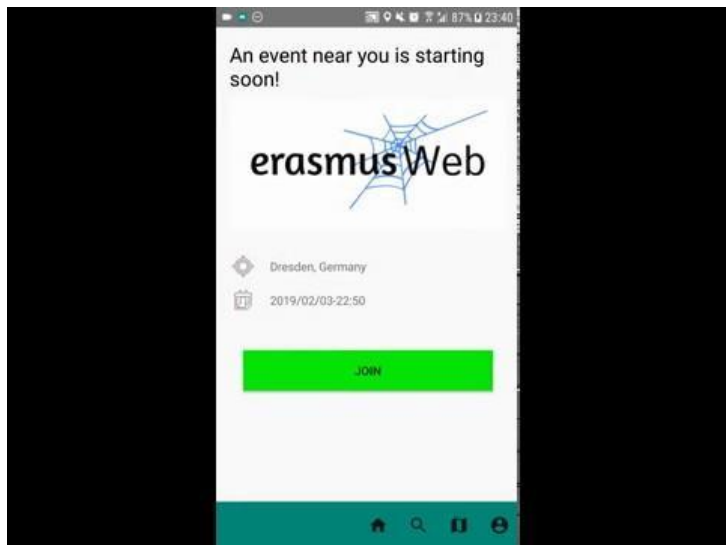
```
Firestore Cloud Functions && Firestore FirebaseMessagingService
```

## What does it do?

- Send push-notifications when the user is within 1 km of a event within the next hour (using GPS tracking), prompting the user with the event page.

# How it works

- Transmit relevant and opportune information (via notifications);



# Challenges

## ➤ Usability

- Minimize user input interaction.

## ➤ Energy

- Minimize the energy consumption caused by GPS usage.

# Energy

- Minimize the energy consumption caused by GPS usage.

## Physical Context:

- Detect the GPS location;
- Two types of GPS tracking: **passive** until the detection of movement; and **active** until the detection of lack of movement;

```
GoogleApiClient && LocationRequest &&  
LocationServices .FusedLocationAPI;
```

- Use of background services to manage the tracking;

```
JobService API && FirebaseDispatcher API;
```

- `Job job = firebaseJobDispatcher.newJobBuilder().build();`
- `firebaseDispatcher.schedule(job);`



# Energy

- Minimize the energy consumption caused by GPS usage.

## Adaptation:

There are two types of GPS tracking:

- **Passive:** detects user location in intervals of 6 minutes (because Android's native battery optimization protocols will stop services that don't "do something" for a max of 10 minutes, from our benchmarks around 7-8 min); If the location is detected to have changed ( $> 500$  m), then the GPS continuous tracking is activated;
- **Active:** enables GPS continuous tracking only when the device is detected to be moving; check every 20 seconds for location changes around 200m and will send location updates to server; will return to passive mode after enough time of lack of movement is shown (from our benchmarks, between 3-5 min);

# Energy

- Minimize the energy consumption caused by GPS usage.

## Adaptation:

- Starting the app schedules background job;
- Background Job starts in passive mode;
- Detection of movement changes the status to active;
- Android's native protocols will stop the job after a few tries of unsuccessful location updates;
- Our app forces the service to restart in passive mode again;

```
public boolean onStartJob(JobParameters job) {
    mLocationRequest = LocationRequest.create();
    mLocationRequest.setSmallestDisplacement(PASSIVE_METERS);
    mLocationRequest.setFastestInterval(PASSIVE_INTERVAL);
    updateLocationUpdates();
}
```

```
public void onLocationChanged(Location location) {
    if (passive && !firstRun) {
        passive = false;
        removePassiveLocation();
        mLocationRequest = LocationRequest.create().
            setSmallestDisplacement(ACTIVE_METERS).
            setFastestInterval(ACTIVE_INTERVAL);
        updateLocationUpdates();
    }
}
```

# Usability

- Minimize user interaction;

## Physical Context:

- Detect user location with GPS;

```
LocationManager.GPS_PROVIDER && LocationManager.NETWORK_PROVIDER
```

- Query last location from the database;

Firestore Realtime Database

# Usability

- Minimize user interaction

## **Adaptation:**

- Database maintains user's last searched location; when the map is opened, the last searched location will be centered; this search has a timeout of two weeks; if that timeout is reached, or no last searched location is found, the map will show the user's current location;

Usa

-

# Usability

- Minimize user interaction

## **Adaptation:**

- (In the map) Display a initial list of events using the bounds of the screen as the coordinates' bounds, limiting the amount of events loaded; if the user zooms out or moves the camera, this amount will be increased and more events will be loaded (based on the new bounds of the screen in the current position);
- (In the list) Displays a initial list of events and when the user presses "Show more results", the search radius is increased by two kilometers, making the results appear ordered by distance.

## Gin & Grill Festival Dresden"



 Dresden, Germany

 2019/02/03-22:50

An mehr als 50 Ständen könnt Ihr Bewährtes, Außergewöhnliches und Leckeres vom Grill genießen. Auch Vegetarier werden hier auf ihre Kosten kommen. Dazu präsentieren wir Euch über 30 verschiedene Ginsorten. Natürlich gibt es auch ein attraktives Rahmenprogramm mit Musik & Kunst.

JOIN

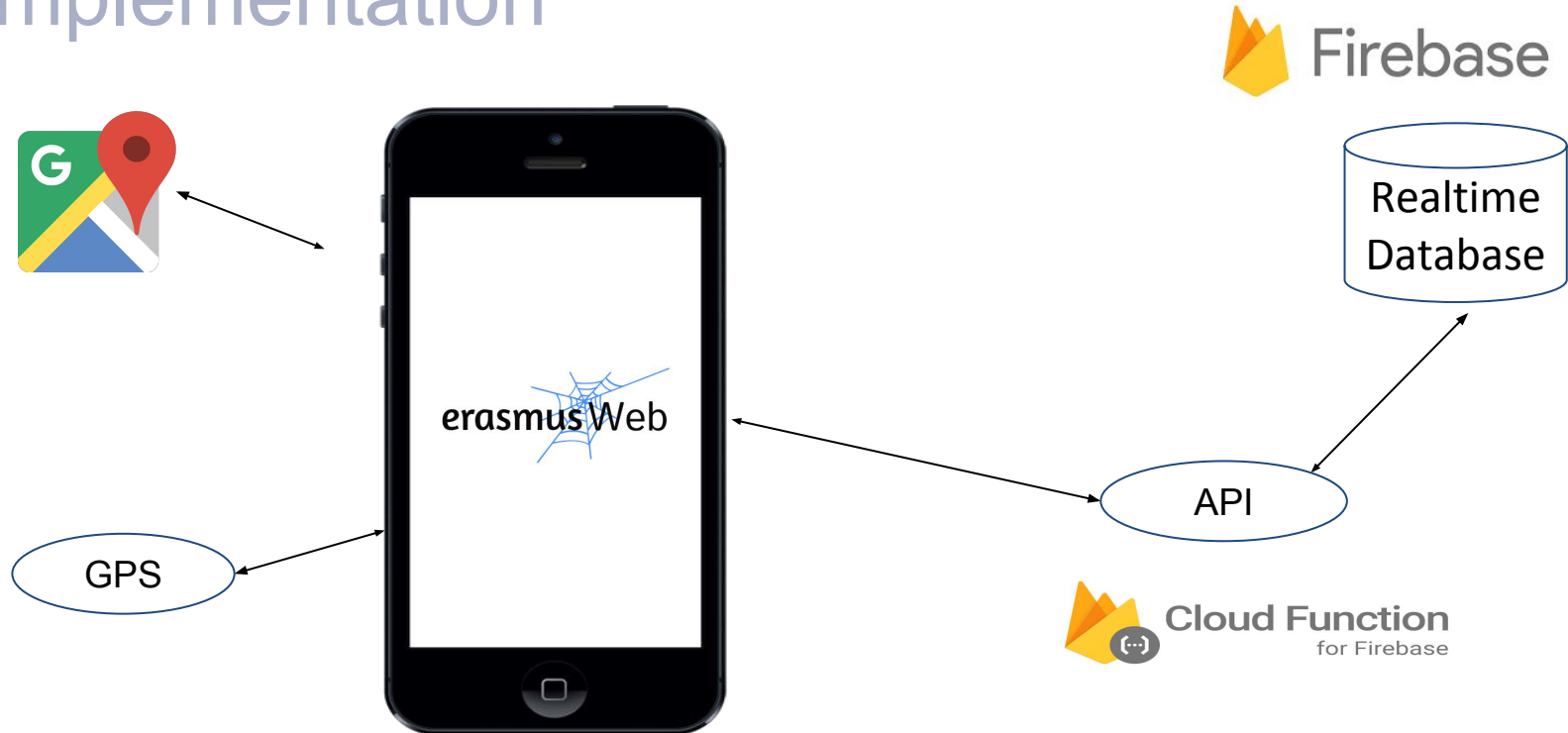


# Technologies

- Android OS/Android Studio;
- Firebase Realtime Database;
- Firebase Cloud Functions, that was used to mimic a server that both interacts with the app and maintains consistency of the database through triggers;
- GPS;
- Google Maps API, for the implementation of the map;
- Firebase functionalities and google APIs for the implementation of challenges;



# Implementation



# Work Plan

- **01.11.2018:** ~~First presentation~~
- ~~November~~
  - ~~Begin of implementation~~
  - ~~Back-end development~~
- ~~December~~
  - ~~First prototype~~
  - ~~Front-end development~~
- **14.12.2018:** ~~Adaption Concepts Presentation~~
- ~~January~~
  - ~~UI-Design~~
  - ~~Bugfixing/Testing~~
- **01.02.2019:** Final Presentation

# Open issues and lessons learned

- UI improvement, since the focus was into functionality;
- More functionalities we didn't have time to implement;
- Some efficient improvements could be done, especially regarding querying from the database;
  
- A lot of knowledge on android since we hadn't worked with it before;
- We were too optimistic regarding how much could be done in the amount of time we had;

END