Mobile Application Development with Android

Developing a basic fitness tracker

Member 1 & Member 2

25. July 2019

Outline

- Introduction & Motivation

- App Structure
  - Frontend Structure
  - Location-Service
  - Database Structure

- Encountered Problems

- Live-Demo

- Further plans

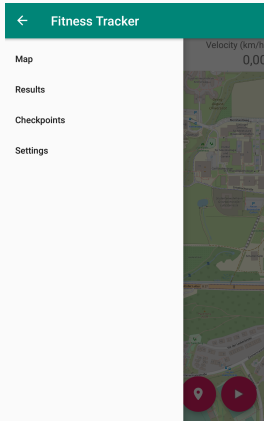## Introduction & Motivation

- We do not trust fitness tracker apps
    - closed source
    - cloud based
- Idea: Create own fitness tracking app
    - open source
    - save data locally
- Collected data:
    - Route
    - Distance
    - Velocity
    - Time

App Structure

- Supports Android 7.0+ (SDK Version 24)

- Project contains only one module

- Divided into multiple packages
  - Activities/Fragments
  - Database packages
  - Services
  - Adapters

- Permissions:
  - Location (GPS)
  - Internet/Network
  - External Storage
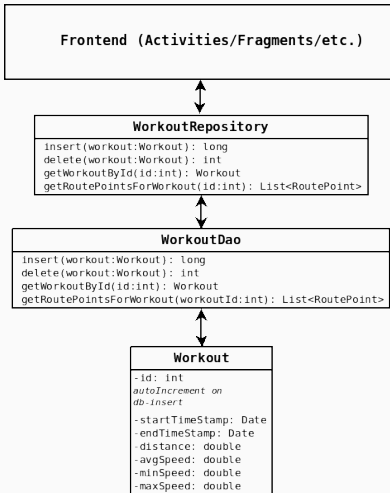 - Foreground Service

- OpenStreetMap using osmdroid

Frontend Structure

- One Activity and a NavigationDrawer
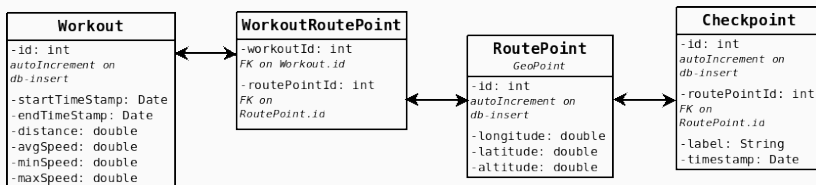- Different screens loaded as Fragments

- Tracks data
- Foreground, background and bound service
- Lifecycle is independent from activity
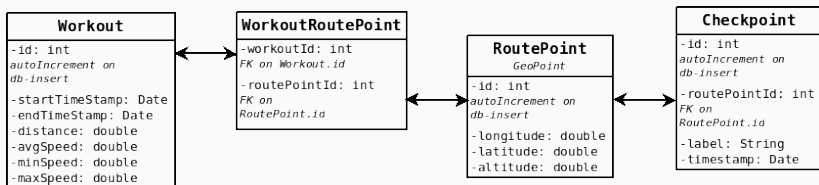
Database Structure - Communication


```
┌─────────────────────────────────────────────┐
│     Frontend (Activities/Fragments/etc.)     │
└─────────────────────────────────────────────┘
                      ↕
┌─────────────────────────────────────────────┐
│              WorkoutRepository               │
├─────────────────────────────────────────────┤
│ insert(workout:Workout): long                │
│ delete(workout:Workout): int                 │
│ getWorkoutById(id:int): Workout              │
│ getRoutePointsForWorkout(id:int): List<RoutePoint> │
└─────────────────────────────────────────────┘
                      ↕
┌─────────────────────────────────────────────┐
│                 WorkoutDao                   │
├─────────────────────────────────────────────┤
│ insert(workout:Workout): long                │
│ delete(workout:Workout): int                 │
│ getWorkoutById(id:int): Workout              │
│ getRoutePointsForWorkout(workoutId:int): List<RoutePoint> │
└─────────────────────────────────────────────┘
                      ↕
       ┌─────────────────────────┐
       │         Workout         │
       ├─────────────────────────┤
       │ -id: int                │
       │ autoIncrement on        │
       │ db-insert               │
       │                         │
       │ -startTimeStamp: Date   │
       │ -endTimeStamp: Date     │
       │ -distance: double       │
       │ -avgSpeed: double       │
       │ -minSpeed: double       │
       │ -maxSpeed: double       │
       └─────────────────────────┘
```

- Frontend communicates with repository

- Repository communicates with DAO

- DAO communicates with database entity

Workout

- -id: int
- *autoIncrement on*
- *db-insert*
- -startTimeStamp: Date
- -endTimeStamp: Date
- -distance: double
- -avgSpeed: double
- -minSpeed: double
- -maxSpeed: double

WorkoutRoutePoint

- -workoutId: int
- *FK on Workout.id*
- -routePointId: int
- *FK on*
- *RoutePoint.ia*

RoutePoint
GeoPoint

- -id: int
- *autoIncrement on*
- *db-insert*
- -longitude: double
- -latitude: double
- -altitude: double

Checkpoint

- -id: int
- *autoIncrement on*
- *db-insert*
- -routePointId: int
- *FK on*
- *RoutePoint.ia*
- -label: String
- -timestamp: Date

Database Structure - Entity Relationship

SQL-Query to get all points for a workout:

```
SELECT id, longitude, latitude, altitude
FROM workout_route_point_table
JOIN route_point_table
ON routePointId = route_point_table.id
WHERE workoutId = :workoutId
```

Encountered Problems

Bugs

- Workout resets itself
- Workout gets killed when in background
- Map loads very slow on first start

Organisational problems

- time management
- too many features

Live-Demo

Further plans

In the next weeks:

- Speed up the initial map loading
- Release the App open-source on GitLab.com
- Maybe release the App to F-Droid

After the exams (september):

- Implement the nice to have features
    - Achievements
    - Social Sharing
    - Usage of weather information
- Code reviews & refactoring

Question time!